# IMAGE DENOISING USING WAVELET TRANSFORMS

DONGWOOK CHO

A THESIS

IN

THE DEPARTMENT

OF

COMPUTER SCIENCE

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE

CONCORDIA UNIVERSITY

MONTRÉAL, QUÉBEC, CANADA

SEPTEMBER 2004

*Your file Votre référence*
*ISBN: 0-612-94737-8*
*Our file Notre référence*
*ISBN: 0-612-94737-8*

# Abstract

## Image Denoising using Wavelet Transforms

Dongwook Cho

Image denoising is a fundamental process in image processing, pattern recognition, and computer vision fields. The main goal of image denoising is to enhance or restore a noisy image and help the other system (or human) to understand it better. In this thesis, we discuss some efficient approaches for image denoising using wavelet transforms. Since Donoho proposed a simple thresholding method, many different approaches have been suggested for a decade. They have shown that denoising using wavelet transforms produces superb results. This is because wavelet transform has the compaction property of having only a small number of large coefficients and a large number of small coefficients. In the first part of the thesis, some important wavelet transforms for image denoising and a literature review on the existing methods are described. In the latter part, we propose two different approaches for image denoising. The first approach is to take advantage of the higher order statistical coupling between neighbouring wavelet coefficients and their corresponding coefficients in the parent level with effective translation-invariant wavelet transforms. The other is based on multivariate statistical modeling and the clean coefficients are estimated in a general rule using Bayesian approach. Various estimation expressions can be obtained by *a priori* probability distribution, called multivariate generalized Gaussian distribution (MGGD). The method can take into account various related information. The experimental results show that both of our methods give comparatively higher PSNR and less visual artifact than other methods.

# Acknowledgments

I have been indebted in the preparation of this thesis to my supervisor, Dr. T. D. Bui, whose passionate academic advice and kindness, as well as support during this work, have been invaluable to me. His extensive knowledge and insight about wavelet transforms and image processing guided me to complete this thesis in addition to all the discussions and careful reading for all my words.

I would also like to thank Dr. C. Y. Suen, Dr. A. Krzyzak and Dr. T. Fevens for their excellent lectures. I have learned many priceless things including knowledge, curiosity for the new fields and attitude as a researcher from them. The informal encouragement and help of many friends has been indispensable, and I would like particularly to acknowledge the contribution of Shuo Li, Guangyi Chen, Song Gao, Chao Jin and Israat Tanzeena Haque.

My wife, Hyojung Kim, has been always my sensible counselor, great supporter and dearest friend. I specially thank her. I am also grateful to my sons, Daniel and Gabriel, who are the joy and everything of my life.

# Contents

# Notations and Abbreviations

## Notations

| | |
|---|---|
| $\mathbb{R}$ | Set of real numbers |
| $\mathbb{C}$ | Set of complex numbers |
| $\infty$ | Infinity |
| $\langle f, g \rangle$ | Inner product |
| $\|f\|$ | Norm |
| $\nabla F$ | Gradient of $F$ |
| $N(\mu, \sigma^2)$ | Gaussian (or normal) distribution |
| $LShift(\cdot)$ | Circular shift function to the left |
| $RShift(\cdot)$ | Circular shift function to the right |
| $\text{diag}(a_1, \cdots, a_n)$ | Diagonal matrix |
| $O$ | Big "O" notation |
| $\text{median}(\cdot)$ | Median function |
| $\hat{x}$ | Estimator of $x$ |
| $sgn(\cdot)$ | sign function |
| $(x)_+$ | if $x < 0$, 0; otherwise $x$; |

# Abbreviations

| | |
|---|---|
| 1D | One dimensional |
| 2D | Two dimensional |
| AWGN | Additive white Gaussian noise |
| CV | Cross validation |
| CWT | Complex wavelet transform |
| DT CWT | Dual-tree complex wavelet transform |
| DWT | Discrete wavelet transform |
| FWT | Forward wavelet transform |
| GCV | Generalized cross validation |
| GGD | Generalized Gaussian distribution |
| GSM | Gaussian scale mixture |
| *iid* | independently and identically distributed |
| IWT | Inverse wavelet transform |
| MAP | Maximum *a posteriori* |
| MGGD | Multivariate generalized Gaussian distribution |
| ML | Maximum likelihood |
| MMSE | Minimum mean squared error |
| MRA | Multiresolution analysis |
| MSE | Mean squared error |
| pdf | Probability density function |
| PSNR | Peak signal to noise ratio |
| QMF | Quadrature mirror filter |
| RMSE | Root mean squared error |
| SI | Shift-invariant |
| SNR | Signal to noise ratio |
| SURE | Stein's unbiased risk estimation |
| TI | Translation-invariant |
| UGGD | Univariate generalized Gaussian distribution |
| WT | Wavelet transform |

# List of Figures

# List of Tables

# Chapter 1

# Introduction

*Image restoration* and *enhancement* are fundamental problems in image processing. In [47], Gonzales and Woods describe that the principal purpose of restoration is to improve a corrupted image by using a priori knowledge of degradation rule while the goal of enhancement is to obtain a better image than the original one for a $specific$ purpose. In brief, restoration is an objective process whereas enhancement is a subjective process. In both cases, image *denoising*, sometimes called *noise removal* or *noise reduction*, plays an important role to achieve those purposes. Denoising is a procedure which removes the existing noise in an image and minimizes the loss of information in a (supposed) clean image.

Wavelet transform has been a popular research subject in many science and engineering areas as well as pure mathematics for the last two decades. In signal and image processing field, wavelet transform tends to replace the role of Fourier transform. Since Mallat introduces *multiresolution analysis* (MRA) theory [66, 67, 68], wavelet has been used in many kinds of image processing applications such as compression, image analysis, computer graphics and watermarking besides denoising. Those algorithms are served as low-level processing units in computer vision and pattern recognition fields.

In a discrete domain, wavelet theory is combined with a filtering theory of signal processing. The coefficients in the wavelet domain have the property that a large number of small coefficients express less important details in an image and a small number of large coefficients keep the information of significance. Therefore, one might suppose that the

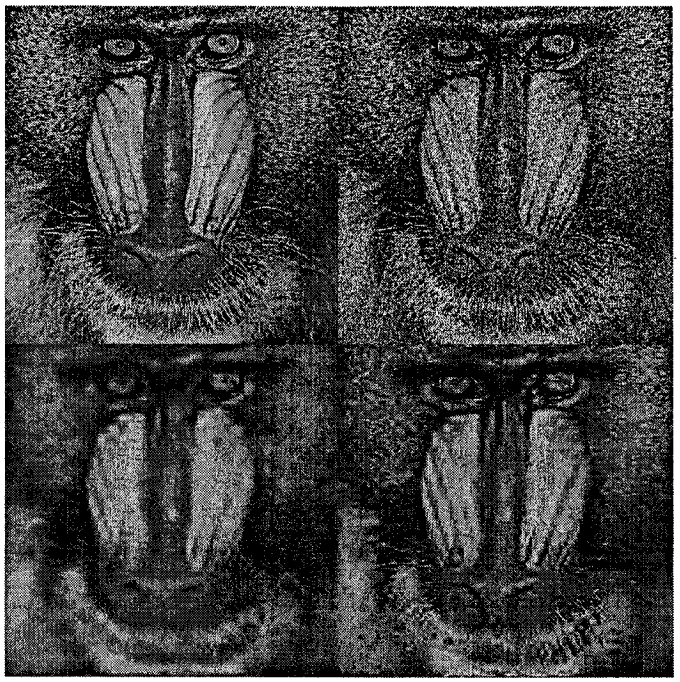


Figure 1: DENOISING EXAMPLE OF *Baboon* IMAGE APPLIED BY DONOHO'S UNIVERSAL THRESHOLD, Original image (top-left), noisy image with $\sigma = 30$ (18.62 dB; top-right), denoised image using soft-threshold (19.80dB; bottom-left), denoised image using hard-threshold (20.32dB; bottom-right).

denoising in the wavelet domain could be achieved by killing the small coefficients which represent the details as well as the noise.

Based on this simple idea, Donoho proposed a neat procedure for denoising in the wavelet domain, called *wavelet thresholding* [35, 33]. If we assume that $A$ is a clean data (signal or image) with size $n$ and $B$ is its noisy data with additive white Gaussian noise (AWGN) $N(0, \sigma^2)$, then the denoising scheme is summarized in three phases as follows:

1. Decompose the noisy data $B$ into an orthogonal wavelet domain.
2. Apply a specific thresholding rule to the coefficients in the wavelet domain by using a threshold like $\lambda = \sigma\sqrt{2\log n}$.
3. Rconstruct the denoised data using inverse discrete wavelet transform from the thresholded coefficients.

Figure 1 shows an example of the denoised images when we use the *universal threshold* $\lambda = \sigma\sqrt{2\log n}$. This shows that thresholding approach is helpful for desnoising and

smoothing, but, at the same time, it makes many of the details lost. Even though the denoised images produce better peak signal to noise ratio (PSNR), we can see that they lose a lot of details.

For image denoising, the definition of noise is critical. In Figure 1, the original clean image has a lot of details and so noise is not so clearly distinguishable from details after adding noise, especially shown in the monkey's hair. For example, it could be noise or detail for the same spot by a subjective point of view.

Another definition about noise is its *type* if we assume that a desired clean image is defined. There could exist numerous types of noise models including AWGN, salt and pepper noise, correlated noise, and uniform noise. In this thesis, we mainly focus on Gaussian additive white noise even though some other types of noise are also considered.

In this thesis, we measure the efficiency and capability of wavelet transforms as a promising mathematical image processing tool in terms of image denoising. For the last decade, there have been proposed many different methods based on the framework of the wavelet thresholding approach described above. However, all the methods have the same goal, that is to obtain an improved image which can satisfy the human visual system or the sensory system of machine to get better features in an image. We introduce various existing and new denoising approaches and evaluate them both visually and numerically.

The organization of this thesis is as follows. In the following chapter, the introductory theory of wavelet transform is briefly described and various kinds of wavelet transforms which are effective for denoising are presented. We categorize and review the existing denoising algorithms in Chapter 3. Several new wavelet shrinkage algorithms are proposed in Chapter 4. These algorithms utilize the information of adjacent coefficients such as neighbours or parent in the wavelet domain. Also the new estimating rule for wavelet coefficients is presented based on multivariate statistical modeling using Bayesian estimator in Chapter 5. In Chapter 6 we present experimental results and performance evaluation. And finally we give the conclusion and future work to be done in Chapter 7.

# Chapter 2

# Wavelet Transforms

Wavelet transform is a great tool in image and signal processing. Many desnoising approaches using wavelet in the literature show that the wavelet is very efficient for image denoising. Unlike the Fourier transforms, the wavelet transform decomposes input data in terms of *time* and *scale* by basis wavelet function, called *mother wavelet*. So various types of wavelets with different properties could be designed. In addition, researchers have developed different types of approaches to reinforce the theory by filling up some possible deficiencies. For example, MRA theory has been combined with discrete wavelet transform (DWT) to play a critical role in image processing. Also filtering theory from signal processing makes the theory more fruitful. As a result, there exist numerous kinds of wavelet algorithms.

In this chapter, we briefly review the DWT which can be explained by *filter banks* and is used much more than the *continuous wavelet transform* in image processing field. More fundamental and broad theory about the wavelet transforms is well discussed in many books [95, 24, 30, 68]. We also give short introduction to several types of wavelet transforms which are effective and promising for image denoising. They include *translation-invariant wavelet*, *multiwavelet*, *complex wavelet* and *directional wavelet*.

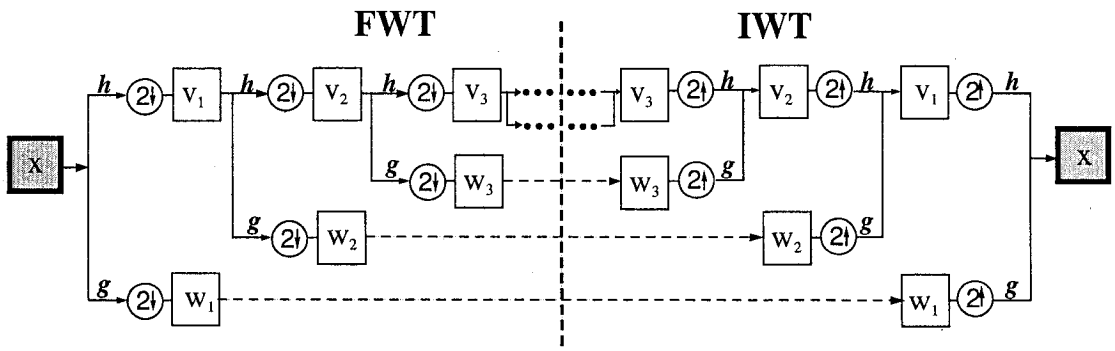



Figure 2: 1D multiresolution wavelet transform.

## 2.1 Scalar Wavelet Transforms

Let $\mathbf{x} = (x_1, x_2, \cdots, x_N)^t$ be a $N$-dimensional vector, which is one-dimensional (1D) descrete signal with length $N$. Also suppose that $\mathbf{h} = (h_1, \cdots, h_p)^t$ and $\mathbf{g} = (g_1, \cdots, g_p)^t$ are low-pass (scaling) and high-pass (wavelet) filters respectively, whose length is $p$. By filtering theory, note that $\mathbf{h}$ and $\mathbf{g}$ are equal length filters and have the following relation : $g_{p-i+1} = (-1)^{i+1} h_i$. This kind of filter bank is called *quadrature mirror filter* (QMF). In this case, the resulting transform can be performed by the conlvolution of the filters $\mathbf{h}$, $\mathbf{g}$ and input data $\mathbf{x}$.

### 2.1.1 Multiresolution analysis

The idea of a multiresolution decomposition is based on the pyramid coding. In descrete data domain, the *resolution* of the data is the level of detail. Since the wavelet transform can be analyzed at differenet scales, the different resolution of wavelet domains can be obtained and a shift-invariant interpretation can be performed. This idea draws the multiresolution analysis. Figure 2 shows the whole process of MRA decomposition and reconstruction. This representation enables us to interpret the input data by a simple hierarchical framework and *coarse-to-fine* analysis.

In brief, forward wavelet transforms (FWT) for input signal $\mathbf{x}$ in $j$-th level by MRA and

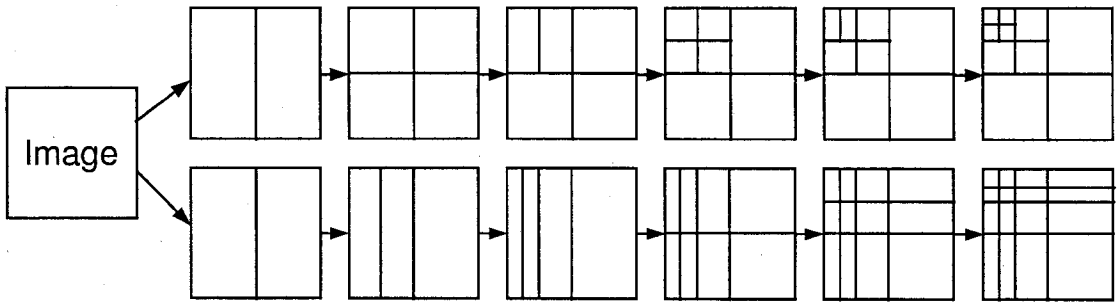



Figure 3: Two different 2D separable WT approaches (upper: square-shaped approach, lower: rectangular approach).

its inverse wavelet transforms (IWT) can be simply expressed using convolution as follows:

$$\mathbf{w}_{j+1} = \mathbf{g} * \mathbf{v}_j;\ \mathbf{v}_{j+1} = \mathbf{h} * \mathbf{v}_j \tag{1}$$

$$\mathbf{v}_j = \mathbf{h} * \mathbf{v}_{\mathbf{j+1}} + \mathbf{g} * \mathbf{w}_{\mathbf{j+1}}, \tag{2}$$

where $\mathbf{w}_j$ and $\mathbf{v}_j$ are vectors for wavelet and scaling coefficients in $j$-th level. $\mathbf{w}_j$ and $\mathbf{v}_j$ can be down-sampled for FWT and up-sampled for IWT. As proven by Daubechies [30], the input data $\mathbf{x}$ decomposed by downsampling can be reconstructed without any data loss.

### 2.1.2 Two-dimensional wavelet transforms

There are several ways to perform two-dimensional (2D) wavelet transform such as *non-separable* transforms [90], *rectangular* separable transform [50], and *square-shaped* separable approach. Among them, the most popular and useful way is square-shaped separable approach by applying two 1D operations for all rows and then all columns for each decomposition level. On the other hand, the rectangular separable transforms apply 1D WT to all the rows of the scaling coefficients until the operations reach the last decomposition level. Then it performs the same process for the column directions. The difference between these two approaches can be found in Figure 3. We assume that the usual 2D wavelet transforms mean square-shaped separable approach in this thesis.

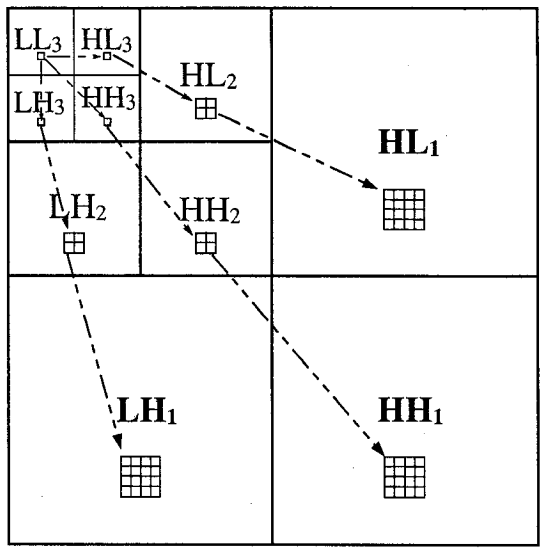



Figure 4: 2D separable wavelet transform.

2D FWT for $j$-th level can be obtained in the similar way to 1D FWT of equation (1):

$$\mathbf{v}_{j+1} = \mathbf{h} * \mathbf{h} * \mathbf{v}_j; \mathbf{w}_{j+1}^{\mathrm{LH}} = \mathbf{g} * \mathbf{h} * \mathbf{v}_j$$
$$\mathbf{w}_{j+1}^{\mathrm{HL}} = \mathbf{h} * \mathbf{g} * \mathbf{v}_j; \mathbf{w}_{j+1}^{\mathrm{HH}} = \mathbf{g} * \mathbf{g} * \mathbf{v}_j, \tag{3}$$

where the superscripts LH, HL and HH denote 2D combinations of low- and high-pass filters.

Similarly, 2D IWT can be expressed as follows:

$$\mathbf{v}_j = \mathbf{v}_{j+1} * \mathbf{h} * \mathbf{h} + \mathbf{w}_{j+1}^{\mathrm{LH}} * \mathbf{g} * \mathbf{h} + \mathbf{w}_{j+1}^{\mathrm{HL}} * \mathbf{h} * \mathbf{g} + \mathbf{w}_{j+1}^{\mathrm{HH}} * \mathbf{g} * \mathbf{g}. \tag{4}$$

In 2D wavelet domain depicted in Figure 4, each square-shaped set of wavelet coefficients is called a *subband* (each separate packet in Figure 2 is also a subband). In 2D image, each decomposition level can have four filtered subbands, LL, LH, HL and HH. The subbands labeled $\mathrm{LH}_1$, $\mathrm{HL}_1$ and $\mathrm{HH}_1$ are sets of wavelet coefficients in the finest level. The following coarser level of subbands, $\mathrm{LH}_2$, $\mathrm{HL}_2$ and $\mathrm{HH}_2$, are obtained from the scaling coefficents in the finer level $\mathrm{LL}_1$. We say that a subband and its adjacent coarser level of subband generated by the same filters have the *parent-child* relationship. For example, $\mathrm{HH}_3$

is a parent subband of $HH_2$ and the current subband $HH_2$ is a child. We also call a subband in the finer level of the current one *offspring* subband. In the same example, $HH_1$ is an offspring subband of $HH_2$ accordingly. This type of multiresolution representation is called *subband coding*. We use this scheme for 2D wavelet transforms of the image throughout the thesis.

## 2.2 Translation-Invariant Wavelet Transforms

Translation invariant (TI) wavelet transforms perform MRA by filtering shifted coefficients as well as the original ones for each decomposition level. TI WT is also known as *shift-invariant* (SI) wavelet transform [15] or *time-invariant* [95]. This approach is *redundant* since additional wavelet coefficients having different properties from the same source are generated in terms of *shifting*. Therefore, more wavelet coefficients are acquired than the usual decomposition, but this redundant analysis is useful for denoising. As a matter of fact, Coifman and Donoho[27] showed that TI WT denoising method produces less mean square errors (MSE) and visual artifacts than denoising method using the usual scalar wavelets. Also more interesting results using TI approach can be found in Chen's theses [16, 17].

### 2.2.1 1D TI wavelet transforms

TI FWT can be extended from FWT by shifting the scaling coefficients for each level. Let $RShift(\cdot)$ and $LShift(\cdot)$ be circular shift functions to the right and the left, respectively. For each level $j$, double size of wavelet coefficients are obtained from both shifted and unshifted scaling coefficients at level $j$+1. Accordingly, 1D TI FWT and TI IWT can be summarized as

$$\mathbf{w}_{j+1}^{2i} = \mathbf{g} * \mathbf{v}_j^i \; ; \; \mathbf{w}_{j+1}^{2i+1} = \mathbf{g} * RShift(\mathbf{v}_j^i), \tag{5}$$

$$\mathbf{v}_j^i = \frac{1}{2}\left(\mathbf{h} * \mathbf{v}_{\mathbf{j+1}}^{\mathbf{2i}} + \mathbf{g} * \mathbf{w}_{\mathbf{j+1}}^{\mathbf{2i}} + LShift(\mathbf{h} * \mathbf{v}_{\mathbf{j+1}}^{\mathbf{2i+1}} + \mathbf{g} * \mathbf{w}_{\mathbf{j+1}}^{\mathbf{2i+1}})\right), \tag{6}$$

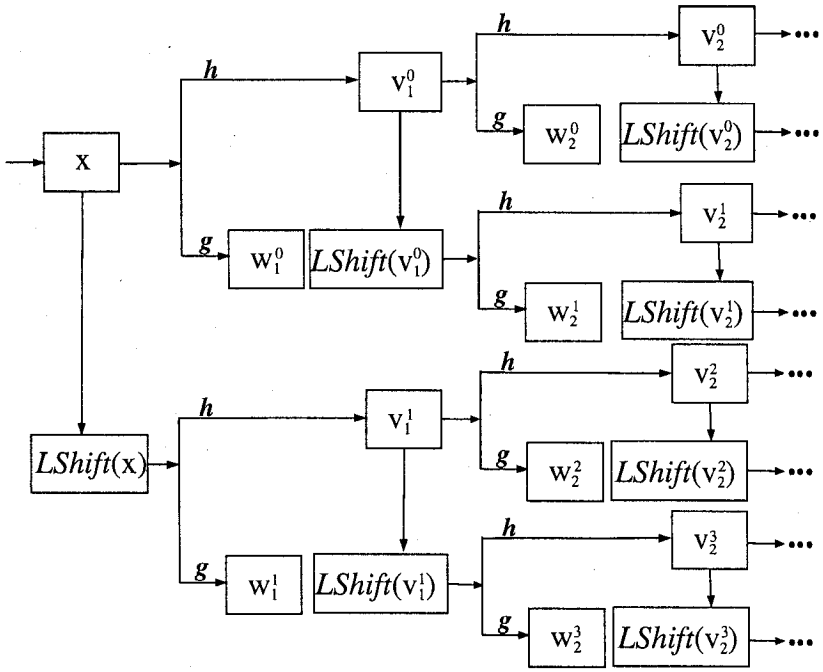



Figure 5: 1D TI wavelet decomposition.

where $\mathbf{w}_j^i$ is $i$-th wavelet packet in $j$-th level. Figure 5 shows multiresolution analysis of TI FWT for 1D signal.

By observation, discrete TI FWT requires $(J - L + 1)N$ length for the wavelet coefficients.

### 2.2.2 2D TI wavelet transforms

2D TI WT can be derived straightforwardly. The main idea of 2D wavelet decomposition is to perform the transforms to an image in two directions, which are $x$ and $y$ direction, in turn. Hence we utilize four different packets, which are *unshifted*, $x$*-shifted* (horizontal), $y$*-shifted* (vertical) and $xy$*-shifted* (diagonal) wavelet packets.

Let $B$ be an $N \times N$ image. We suppose that $RShift_x(\cdot)$ is a circular shifting function to the right on the $x$-axis. Similarly we can define $RShift_y(\cdot)$ and $RShift_{xy}(\cdot)$. $LShift$

is also defined by the same analogy of $RShift$. Then we can get the following equations:

$$\begin{aligned} \mathbf{w}_{j+1}^{2h,2i,\mathrm{HH}} &= \mathbf{g} * \mathbf{g} * \mathbf{v}_j^{h,i}\ ; \quad \mathbf{w}_{j+1}^{2h+1,2i+1,\mathrm{HH}} = \mathbf{g} * \mathbf{g} * RShift_x(\mathbf{v}_j^{h,i}) \\ \mathbf{w}_{j+1}^{2h,2i+1,\mathrm{HH}} &= \mathbf{g} * \mathbf{g} * RShift_y(\mathbf{v}_j^{h,i})\ ; \quad \mathbf{w}_{j+1}^{2h+1,2i+1,\mathrm{HH}} = \mathbf{g} * \mathbf{g} * RShift_{xy}(\mathbf{v}_j^{h,i}), \end{aligned} \tag{7}$$

where $\mathbf{w}_j^{h,i,\mathrm{HH}}$ indicates a square matrix composed of wavelet coefficients in a level $j$ which is produced by high-pass filters for both directions and located in $(h,i)$-th block. We can acquire by analogy $\mathbf{w}_j^{\mathrm{LH}}$, $\mathbf{w}_j^{\mathrm{HL}}$ and $\mathbf{v}_j^{\mathrm{LL}}$ by combining high- and low-pass filters. Then the wavelet coefficients can be obtained like Figure 6. It should be noted that four scaling subbands generated by the shift functions are used in each decomposition level and each scaling subband is decomposed into four different subbands, LL, LH, HL, and HH, just like the usual 2D wavelet transforms. Hence, each decomposition level for 2D TI FWT generates 16 down-sampled subbands while the usual 2D FWT produces 4 down-sampled subbands. This process for each decomposition level $j$ is illustrated in the upper figure of Figure 6. The scaling coefficients $\mathbf{v}$ except the last decomposition level are not stored as shown in the lower figure of Figure 6.

2D TI IWT for reconstruction should be operated inversely using $LShift$. This can be expressed as follows:

$$\begin{aligned} \mathbf{v}_j^{h,i,\mathrm{LL}} = \frac{1}{2}\Big[ & \mathbf{h} * \mathbf{h} * \mathbf{v}_{j+1}^{2h,2i,\mathrm{LL}} + \mathbf{h} * \mathbf{g} * \mathbf{w}_{j+1}^{2h,2i,\mathrm{LH}} \\ & + \mathbf{g} * \mathbf{h} * \mathbf{w}_{j+1}^{2h,2i,\mathrm{HL}} + \mathbf{g} * \mathbf{g} * \mathbf{w}_{j+1}^{2h,2i,\mathrm{LL}} \\ & + LShift_x\Big(\mathbf{h} * \mathbf{h} * \mathbf{v}_{j+1}^{2h+1,2i,\mathrm{LL}} + \mathbf{h} * \mathbf{g} * \mathbf{w}_{j+1}^{2h+1,2i,\mathrm{LH}} \\ & + \mathbf{g} * \mathbf{h} * \mathbf{w}_{j+1}^{2h+1,2i,\mathrm{HL}} + \mathbf{g} * \mathbf{g} * \mathbf{w}_{j+1}^{2h+1,2i,\mathrm{HH}}\Big) \\ & + LShift_y\Big(\mathbf{h} * \mathbf{h} * \mathbf{v}_{j+1}^{2h,2i+1,\mathrm{LL}} + \mathbf{h} * \mathbf{g} * \mathbf{w}_{j+1}^{2h,2i+1,\mathrm{LH}} \\ & + \mathbf{g} * \mathbf{h} * \mathbf{w}_{j+1}^{2h,2i+1,\mathrm{HL}} + \mathbf{g} * \mathbf{g} * \mathbf{w}_{j+1}^{2h,2i+1,\mathrm{HH}}\Big) \\ & + LShift_{xy}\Big(\mathbf{h} * \mathbf{h} * \mathbf{v}_{j+1}^{2h+1,2i+1,\mathrm{LL}} + \mathbf{h} * \mathbf{g} * \mathbf{w}_{j+1}^{2h+1,2i+1,\mathrm{LH}} \\ & + \mathbf{g} * \mathbf{h} * \mathbf{w}_{j+1}^{2h+1,2i+1,\mathrm{HL}} + \mathbf{g} * \mathbf{g} * \mathbf{w}_{j+1}^{2h+1,2i+1,\mathrm{HH}}\Big)\Big] \end{aligned} \tag{8}$$

In 2D, we can realize that shifting occurs in two different axes. Therefore, there exist

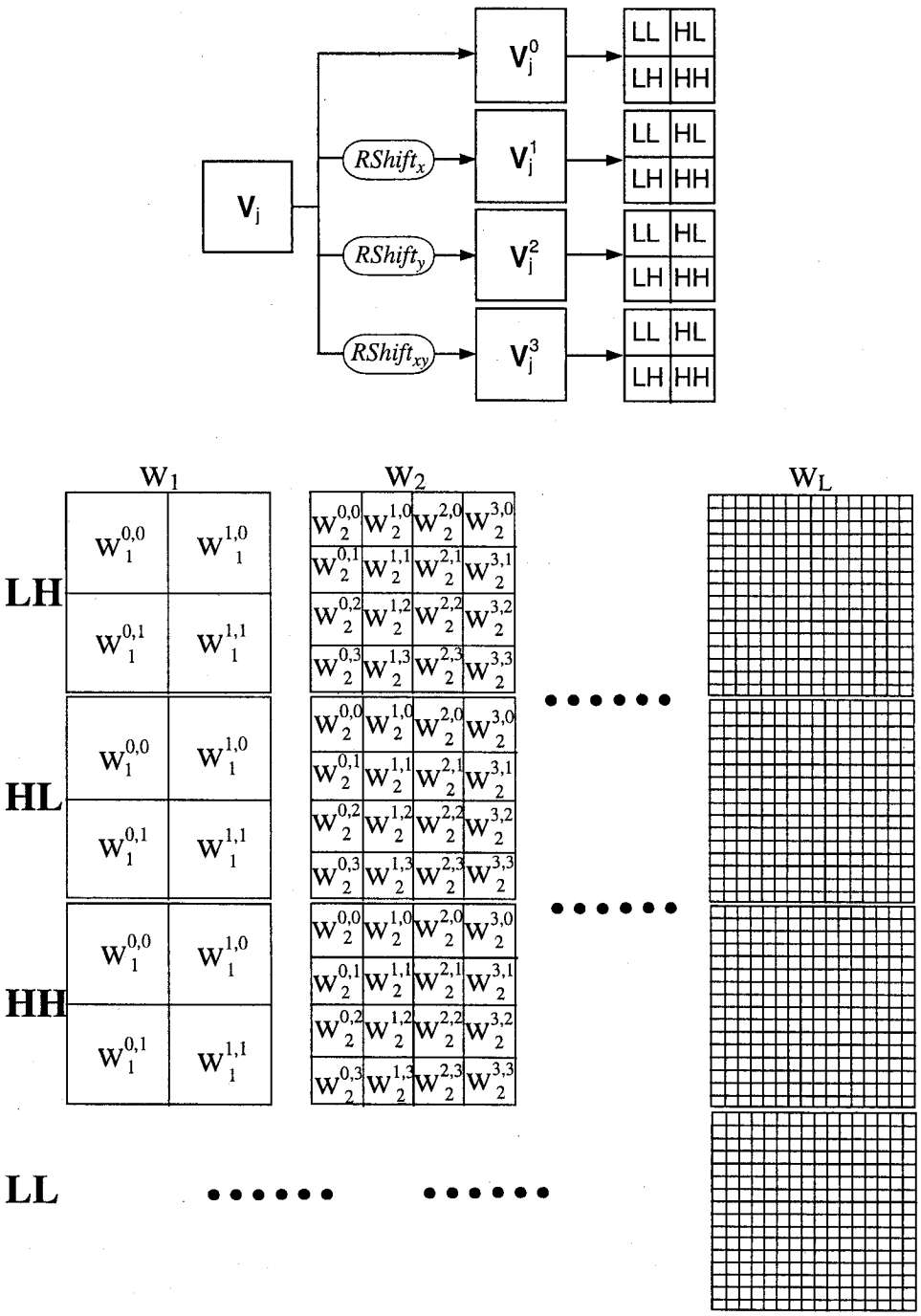



Figure 6: 2D TI wavelet decomposition

four different shifted packets. When we perform $L$ steps of decomposition iteratively, the required storage for TI wavelet coefficients of an image is $(3L+1)N^2$ since wavelet packets are quadrupled for each level but the LL packets only in the coarsest level are required.

## 2.3 Multiwavelets

Multiwavelets are the wavelets generated by more than one scaling function while the scalar wavelet described in the previous section uses only one scaling function. This seems to be simple, but this gives many advantages such as boundary improvement via linear-phase *symmetry*, *orthogonality* which can provide the perfect reconstruction with the same size of input data, and *vanishing moments* as pointed out in [96]. Also multiwavelets can preserve good properties of scalar wavelet such as MRA.

The most popular multiwavelet system is GHM (named after the authors Geronimo, Hardin and Massopust) proposed in [46]. They have used two symmetric scaling functions and their wavelet functions which generate a symmetric/antisymmetric pair. Another important multiwavelet system is CL multiwavelet (named after Chui and Lian like GHM) [25].

We can describe generalized multiwavelets based on the scalar wavelet theory. If we suppose downsampling in equation (1), it can be rewritten varying in time $t$ as $w_{j+1}(t) = \sum_i g_i v_j(2t-i)$ and $v_{j+1}(t) = \sum_i h_i v_j(2t-i)$, where $v(t)$ is a scaling function and $w(t)$ is a wavelet representation. Then the general multiwavelet decomposition can be extended by matrix dilation as follows:

$$\begin{aligned} \mathbf{w}_{j+1}(t) &= \sum_i \mathbf{G}_i \mathbf{v}_j(2t-i)\ ; \\ \mathbf{v}_{j+1}(t) &= \sum_i \mathbf{H}_i \mathbf{v}_j(2t-i), \end{aligned} \tag{9}$$

where $\mathbf{v}_j(t)$ is a vector composed of $m$ scaling functions and $\mathbf{G}_i$ and $\mathbf{H}_i$ are $m \times m$ matrices. In other words, the filter banks $\mathbf{H}$ and $\mathbf{G}$ consist of $p$ m×m matrices each when $p$ is the length of each filter. In this case, the input data $\mathbf{x}$ needs to be changed in the form of a set

of vector $\mathbf{v}_0$. In order to preserve the same length in the wavelet domain, this condition requires *prefilter* and *postfilter*. In fact, the prefilter and postfilter design is an important factor of the multiwavelet performance [103, 102].

Well-studied review for image processing applications of multiwavelets is described in [96]. Also some denoising algorithms using multiwavelets are proposed in [9, 4, 5, 16, 18, 38, 63, 97].

## 2.4 Complex Wavelets

Complex wavelet transforms (CWT) are a comparatively recent addition to the wavelet studies. Complex number includes some properties that can not be represented by real number. The properties of complex number lead CWT to provide better *shift-invariant* feature and *directional selectivity*. However, CWT with perfect reconstruction and good properties of filter are difficult to develop. So some researchers have tried to find an effective way to design complex wavelets. Some of the early works can be found in [58, 62]. Since then, many researchers have paid attention to CWT [54, 43, 44]. In this section, we introduce Kingsbury's dual-tree complex wavelet transform (DT CWT) [54, 55, 56], which can be easily extended using the conventional DWT algorithm and is used in our experiments for denoising algorithms.

### 2.4.1 Dual-tree Complex Wavelets

In 1998, Kingsbury proposed dual-tree complex wavelet transforms (DT CWT) which can overcome some drawbacks of his previous CWT approach in [64]. DT CWT have some good properties such as reduced *shift sensitivity*, good *directionality*, *perfect reconstruction* using linear-phase filters, explicit phase information, fixed redundancy and effective computation in $O(N)$.

Design of CWT filters which can achieve prefect reconstruction with more than one level of decomposition is difficult. Kingsbury proposed dual-tree approach to overcome this problem by applying dual filters as shown in Figure 7.

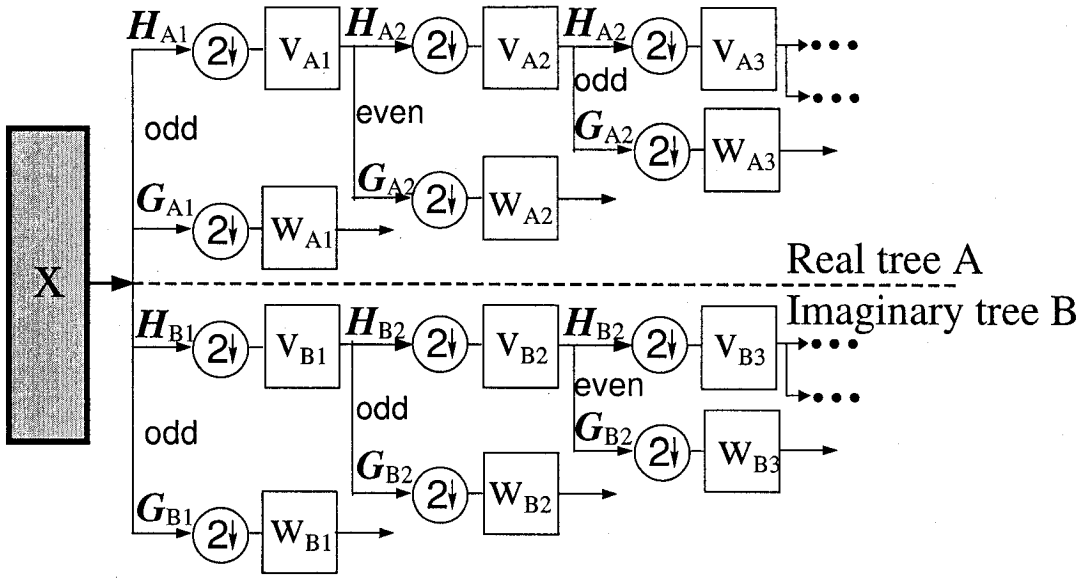



Figure 7: 1D decomposition by DT CWT.

Due to the property of the complex filters which emphasizes only positive (or negative) frequencies, perfect reconstruction is not possible in single tree structure. In addition, it is well-known that *undecimated wavelet* which does not perform down-sampling, can achieve shift invariance.

We can achieve near shift-invariant when the sampling rate is doubled at each level of the tree A in Figure 7 by eliminating the down-sampling after the level 1 filter, And this procedure is the same as the case we have dual *decimated* trees. In this case, we consider uniform intervals between samples after the 1st level by changing filter lengths. This is key idea of DT CWT. Figure 8 shows a simple example about effective sampling location for each level when the 1D input signal $\mathbf{x}$ has 16 sample points. As shown in the example, uniform interval between samples can be obtained by applying odd-length and even-length filters for each tree alternatively.

2D DT CWT can be extended by applying the filters to column and row one by one like ordinary DWT. Therefore, this produces four times of redundancy since 2:1 redundancy is required for each as depicted in Figure 7. In general DT CWT requires $2^m : 1$ redundancy for $m$-dimensional transforms. This is performed by complex conjugates and achieves great symmetry of CWT. For instance, if we assume that we get four-elements of complex

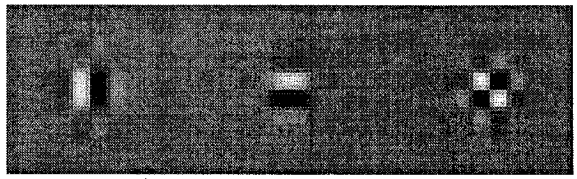
| Input x | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | : | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1st Level** | (odd) $\mathbf{v}_{A1}$ | A | | A | | A | | A | | : | A | | A | | A | | A | |
| | (odd) $\mathbf{w}_{A1}$ | | A | | A | | A | | A | : | | A | | A | | A | | A |
| | (odd) $\mathbf{v}_{B1}$ | | B | | B | | B | | B | : | | B | | B | | B | | B |
| | (odd) $\mathbf{w}_{B1}$ | B | | B | | B | | B | | : | B | | B | | B | | B | |
| **2nd Level** | (even) $\mathbf{v}_{A2}$ | | A | | | | A | | | : | | A | | | | A | | |
| | (odd) $\mathbf{v}_{B2}$ | | | | B | | | | B | : | | | | B | | | | B |
| | $\mathbf{w}_{A2}, \mathbf{w}_{B2}$ | | * | | | | * | | | : | | * | | | | * | | |
| **3rd Level** | (odd) $\mathbf{v}_{A3}$ | | A | | | | | | | : | | A | | | | | | |
| | (even) $\mathbf{v}_{B3}$ | | | | | | B | | | : | | | | | | B | | |
| | $\mathbf{w}_{A3}, \mathbf{w}_{B3}$ | | | | | | * | | | : | | | | | | * | | |
| **4th Level** | (even) $\mathbf{v}_{A4}$ | | | | | | A | | | : | | | | | | | | |
| | (odd) $\mathbf{v}_{B4}$ | | | | | | | | | : | | | | | | B | | |
| | $\mathbf{w}_{A4}, \mathbf{w}_{B4}$ | | | | | | * | | | : | | | | | | | | |

Figure 8: Example of sampling points of odd and even filters for length 16 signal.

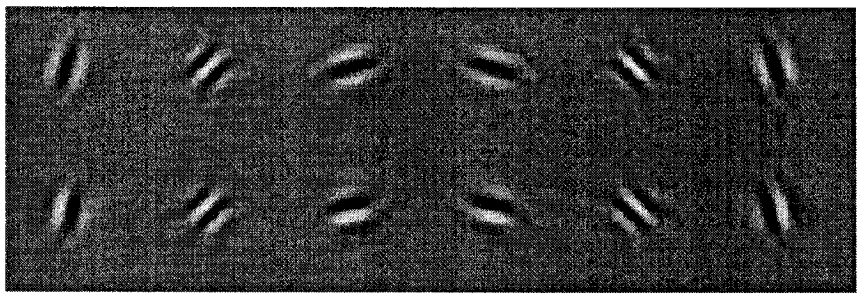
vector $(a, b, c, d)^t$ after a series of row and column filtering, then this can be expressed as $a + bi_1 + ci_2 + di_1i_2$. This produces $(a - d) + (b + c)i$ and $(a + d) + (-b + c)i$ by complex operations supposing $i_1 = i_2 = i$ and its conjugate pair $i_1 = -i_2 = -i$. These properties of complex filters yield good directional selectivity in multi-dimensional space. Six different directions for real and imaginary parts are emphasized in 2D DT CWT while DWT has only three directions as shown in Figure 9. Each directional element including angles $\pm\pi/12$, $\pm\pi/4$ and $\pm5\pi/12$ is from each subband of real and imaginary parts.

## 2.5 Other effective wavelets for image denoising

The wavelet transforms that generate more wavelet coefficients than the size of input data are called *redundant* or *overcomplete*. For the scalar wavelet transforms, downsampling with subband coding and pyramid representation makes it possible that the size of coefficients in the wavelet domain is the same as in the spatial domain. On the other hand, TI WT and CWT are overcomplete. Multiwavelet is not overcomplete except the case that *repeated signal* prefilter is applied. The redundancy of the wavelet transforms is helpful for the shift invariance property in the wavelet domain.



(a) 2D DWT



(b) 2D DT CWT

Figure 9: Impulse response of 2D wavelets.

When we described the properties of CWT in the previous section, the directional selectivity is also important for image representation. For 2D separable implementation of DWT, only three directions can be considered as seen in Figure 9(a). But more directions can be considered in DT CWT. These types of wavelets are called *directional* wavelets. Recently many approaches including [12, 13, 31, 69, 91, 99] have been developed by emphasizing directionality. Since directional wavelet transforms have more elemental directions, it is possible to emphasize features along lines or curves which are important in processing natural images. These features are also useful in building an effective thresholding rule for image denoising. Figure 10 illustrates one of the directional wavelet approaches developed by Candes et al. [12, 13]. This is called *curvelet*, which gives good property to extract lines (or curves) by combining wavelet transform and *ridgelet* transform based on *Radon* transform. The denoising application using curvelet transforms can be found in [92, 93].

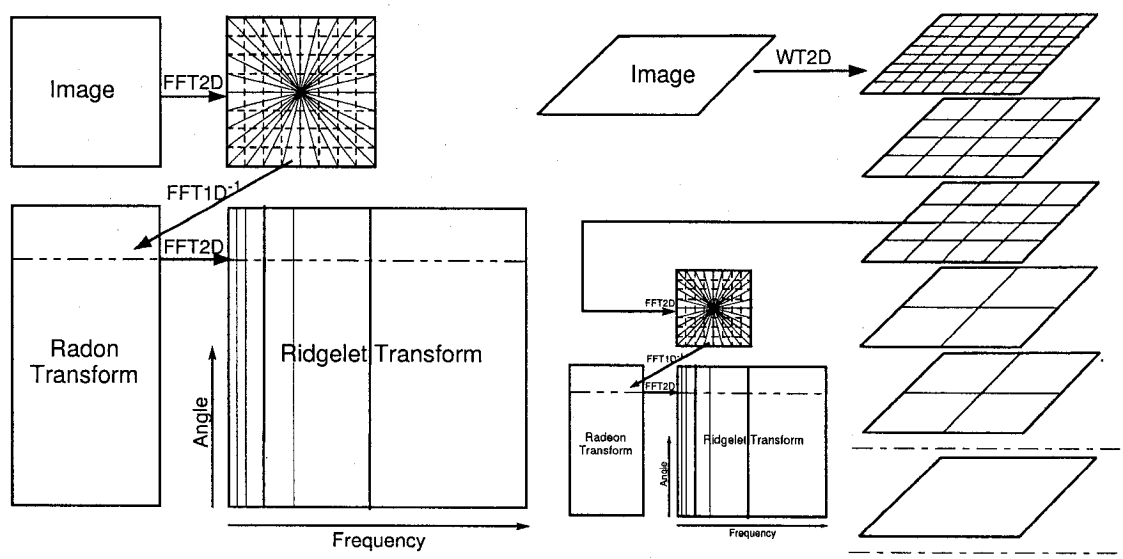



Figure 10: Ridgelet (left) and curvelet (right) transforms (Courtesy of J. Starck et al. [93]).

# Chapter 3

# Review on Image Denoising using Wavelet Transforms

## 3.1 Introduction

Various research works on image denoising using wavelet transforms have shown that wavelet transform is an efficient tool for the enhancement of noisy images. This is because wavelet transform has the compaction property [67] of having only a small number of large coefficients. All the rest wavelet coefficients are very small. These small coefficients keep detail information of the image. Therefore, we deal with the detail coefficients in the wavelet domain for removing the noise which usually considered as the detail information.

As described in the first chapter, Donoho's thresholding algorithm has three phases including forward wavelet transform, coefficient thresholding (or estimation in general) in the wavelet domain, and reconstruction of the estimated wavelet coefficients. His idea can be generalized and illustrated as shown in Figure 11.

This three-phase algorithm is very simple, but there exist numerous different approaches to perform this simple procedure. In this chapter, we first categorize the existing approaches and describe some representative works briefly. An extensive survey on image denoising using wavelet transforms, which summarizes the existing approaches for a decade and give

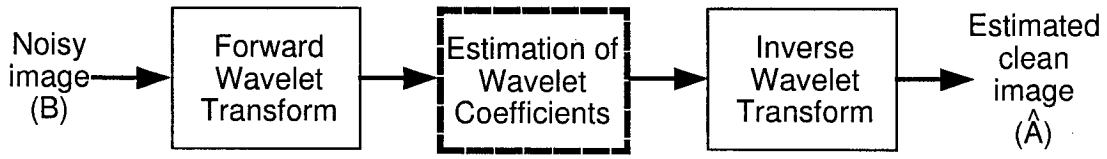



Figure 11: Image denoising procedure using wavelet transforms.

a critical review with wide and deep insight, to our knowledge, is not found yet, but there are a few well-studied articles on wavelet thresholding [50] or simulation study [3]. Mallat also describes some fundamental issues on denoising and wavelet estimation theory nicely in terms of Bayes and minimax estimators in [68].

There are several ways to categorize the existing denoising methods. First of all, a method can be classified by the wavelet transform as explained in Chapter 2. Some related works for special wavelet transforms for denoising are listed in Table 1. Another way to categorize the methods is by noise type. In most of works, they suppose zero-mean AWGN as a noise type since Gaussian noise has some good properties such as symmetry, continuity and smoothness of the density distirbution. However, there exist many other types of noise in practice. For example, noise can be correlated even though they still keep Gaussian dsitribution. Or noise can be distributed in different ways like Poisson, Laplacian distribution, or non-additive salt-pepper noise. If we say more practically, scratches in an image can be also noise. So it is not easy to define the noise itself. In addition, it may be possible that a denoising approach depends on the noise type. The last categorization method we can consider depends on the estimation approaches of the clean coefficients in the wavelet domain. In Figure 11, the second phase can have different approaches and there are many works to focus on this part. In this chapter, we also focus on this categorization scheme.

For convenience's sake, we divide the methods into three categories including *threshold*, *shrinkage*, and other *etimation* approaches. Threshold approach is the way to get estimated clean wavelet coefficients by comparing a noisy wavelet coefficient with a certain threshold and deciding the estimated value based on the threshold. Shrinkage approach is to *shrink* noisy coefficients by certain shrinkage rule and consider it as the estimated values

of clean coefficients. One problem for this categorization is that threshold and shrinkage are ambiguous to divide them precisely. In fact, one can say that shrinkage is one type of thresholding approaches, and vice versa in some sense. So we define them in the following ways. Suppose that $x$ is a clean wavelet coefficient and $w$ is a given noisy coefficient. Then our goal is to get an estimated coefficient $\hat{x}$. The methods in the threshold approaches use a threshold $\lambda_t$ and they are classified by the way to obtain $\lambda_t$. For example, if we use a *soft thresholding* approach (see the following section for detail), the threshold approaches have the common form as follows:

$$\hat{x} = sign(y)\left(|y| - \lambda_t\right)_+ ,$$

where $sign(\cdot)$ is a function to decide that the value is +1 if positive or -1 if negative and $(\cdot)_+$ is a function to set the value to zero if the value is less than zero, otherwise to keep the original value. On the other hand, the methods in the shrinkage category have the following form:

$$\hat{x} = y\left(1 - \lambda_s\right)_+ , \tag{10}$$

where a shrinkage constant $\lambda_s$ is between 0 and 1 ($0 \leq \lambda_s \leq 1$).

We call both $\lambda_t$ and $\lambda_s$ *threshold* since equation (10) can be easily changed in the form of thresholding approach. Most of the chapter is devoted to estimate the threshold. In the next section, we define the problem set for denoising in terms of risk minimization in the wavelet domain. Then various existing approaches shown in Table 1 are described by abovementioned categorization scheme.

## 3.2 Estimation and Denoising

Denoising is a procedure to calculate the estimators. The problem set for general denoising can be expressed in the following. In the wavelet domain, suppose that $y_{j,k}$ is a wavelet coefficient of a noisy signal $B$ with length $N$ and $x_{j,k}$ is the corresponding wavelet coefficient

| Category | Approach | Related works |
|---|---|---|
| Thresholding | Universal thresholding | [35, 37, 33] |
| | Stein's unbiased risk estimation (SURE) | [94, 36] |
| | Cross validation | [72, 101, 52, 51] |
| | BayesShrink | [14] |
| Shrinkage | MMSE | [70, 60] |
| | Bivariate shrinkage using level dependency | [84, 85] |
| | Neighbour dependency | [11, 18] |
| | Adaptive Bayesian wavelet shrinkage (ABWS) | [19] |
| | Markov Random Field | [65, 74] |
| | Hidden Markov Tree | [29, 23, 41, 82] |
| Other | Gaussian scale mixture | [77, 78, 76] |

| Wavelet Type | | Related works |
|---|---|---|
| Orthogonal separable wavelet | | Most of works |
| Translation-invariant wavelet | | [27, 26, 9, 18, 15, 60, 6] |
| Multiwavelet | | [38, 9, 18, 97, 63, 4, 5, 49, 40] |
| Complex wavelet | | [23, 55, 61, 83, 84, 85] |
| Others | Curvelet | [92, 93, 32] |
| | Steerable pyramid | [89, 78] |
| | Brushlet | [57] |

Table 1: Categorization of image denoising methods by estimation approach of wavelet coefficient (above) and wavelet type (below).

of the signal without noise, then

$$y_{j,k} = x_{j,k} + z_{j,k}, \tag{11}$$

where $z_{j,k}$ is the noise with $N(0, \sigma^2)$. Our goal for denoising is to obtain the estimate $\hat{x}_{j,k}$ of $x_{j,k}$ for every level and position. When we suppose that $\mathbf{x}$ is a set of $x_{j,k}$, our goal can be achieved by minimizing the difference between $\mathbf{x}$ and $\hat{\mathbf{x}}$, which is often called *risk*.

To measure the risk, we employ a loss function $\epsilon$ defined by Euclidean distance or $L^2$ norm: $\epsilon = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$. Then the risk of the estimator $\hat{\mathbf{x}}$ of $\mathbf{x}$ is the average loss : $R = E(\epsilon)$. Therefore, our problem set for denoising is to minimize the risk $R$. If we assume that $\hat{\mathbf{x}}$ is estimated by the transformation of the given noisy signal $\mathbf{y}$ using a decision operator $D$

like $\hat{\mathbf{x}} = D\mathbf{y}$, we need to estimate the optimal operator $D^*$ as follows:

$$D^* = \arg\min_D R(D) = \arg\min_D E[\|\hat{\mathbf{x}} - \mathbf{x}\|^2]. \tag{12}$$

There are two main approaches to expand the risk $R$ and to estimate the operator $D$ by using *Bayes risk* and *minimax risk*. Bayes risk assumes that we know a prior pdf. Minimax approach uses prior set of the signals instead of using prior probability distribution.

Most of denoising approaches begin from this point of view. However, it is not possible to calculate the general estimator which minimizes the risk $R$ using Bayes or minimax approach due to the complexity and non-linearity of the equation. We review some important image denoising algorithms in the rest of the chapter.

## 3.3 Threshold Approaches

Before we survey some representative works to calculate the threshold $\lambda_t$, it should be noted that there exist some different thresholding approaches to use $\lambda_t$. In his early works like [35] and [33], Donoho defines two thresholding rules, called *hard-thresholding* and *soft-thresholding*. *Hard-thresholding kills* all the coefficients smaller than the threshold $\lambda_t$ and leaves the others without changes while *soft-thresholding* shrinks the rest of the coefficients by the threshold $\lambda_t$. Mathematically these thresholding rules can be re-stated respectively as follows:

$$\textbf{Hard-thresholding}: \hat{x} = \Theta_H(y, \lambda_t) = \begin{cases} 0 & \text{if } |y| < \lambda_t \\ y & \text{otherwise} \end{cases}, \tag{13}$$

and

$$\textbf{Soft-thresholding}: \hat{x} = \Theta_S(y, \lambda_t) = sign(y)\,(|y| - \lambda_t)_+ . \tag{14}$$

Under these thresholding rules, our goal is to calculate the optimal threeshold $\lambda_t$. We can choose the same value for $\lambda_t$ and apply it to all the wavelet coefficients regardless of subband. This way is called *global* thresholding. Another way to obtain $\lambda_t$ is by the

statistical characteristics in each wavelet subband. In this case, we can compute different $\lambda_t$ for each subband, which is called *subband* thresholding. At last, the threshold could be computed for each wavelet coefficient using adaptive approaches. Then a threshold for each location will be different according to the wavelet coefficient. We call this approach *adaptive* thresholding.

### 3.3.1 Universal threshold

Universal threshold proposed and proven in [35, 37, 33] is most widely used and well-known in the wavelet denoising literature. It is probably the most popular global thresholding approach. The threshold can be formulated as follows:

$$\lambda_t = \sigma\sqrt{2\log N}, \tag{15}$$

where $N$ is the size of signal or image and $\sigma$ is noise variance. When we decide the threshold $\lambda_t$, it should be just above the maximum level of noise but not be too large. For instance, one can imagine that too many large coefficients should not be killed, e.g. $|x_k| \geq \sigma$, which considers noise level. Also, the threshold $\lambda_t$ increases with the length $N$ due to the tail of the Gaussian distribution.

More formally, it is possible to prove that the maximum amplitude of the noise has a very high probability of being just below the universal threshold $\lambda_t$:

$$\lim_{N\to\infty} P\left(\lambda_t - \frac{\sigma\log\log N}{\log N} \leq \max_{1\geq k\geq N} |z_k| \leq \lambda_t\right) = 1.$$

Universal threshold is nearly minimax for general function since it does not require prior information like Bayesian approach. This threshold can be applied to all the smooth data as Donoho and his collaborators have proven various optimal properties for this simple threshold.

The universal threshold is a good choice when the size of input signal $N$ approaches infinity. Also, the statistical smoothness which has asymptotic behavior is better considered

than the minimization of the mean squared error. In practice, this approach is simple and fast. However, when it is applied to an image, it produces a denoised image which loses many of the details.

### 3.3.2 *SureShrink*

We can reduce thresholding risk by choosing a threshold smaller than universal threshold. However, universal threshold can be applied for any input function regardless of the data statistics of the function. It may be possible to calculate an adaptive threshold to the data which can minimize the estimated risk.

An adaptive threshold, called *SureShrink*, was developed by Donoho and Johnstone [36]. *SureShrink* is named from Stein's unbiased risk estimation (*SURE*). Like the universal threshold, the threshold by *SureShrink* achieves asymptotic minimax optimialities over function spaces like Besov spaces. In addition, it is optimally smoothness-adaptive by the statistics of the input data.

The threshold $\lambda_t$ by *SureShrink* can be defined in the following.

$$\begin{aligned} \lambda_t &= \arg\min_{\lambda} SURE(\lambda) \\ &= \arg\min_{\lambda} \left[ N - 2 \cdot \#\{k : |y_{j,k}| \leq \lambda\} + \sum_{k}^{N} \min(|y_{j,k}|, \lambda)^2 \right]. \end{aligned} \tag{16}$$

Since *SureShrink* is an adaptive approach, different threshold can be decided for each subband. This type of threshold is called *subband* (or *level*) threshold. The threshold for image denoising yields much better image quality and lower mean squared error (MSE) than the universal threshold which produces too smooth function spaces by bigger threshold and is not appropriate for the natural images having complicated structures. Also, it produces the numerical results close to the minimum MSE of the optimal thresholding estimator. In spite of the high performance, however, Donoho and Johnstone in [36] pointed out that *SureShrink* in extremely sparse wavelet representations might obtain an inadequate threshold and they suggested hybrid approach as an alternative.

### 3.3.3 *BayesShrink*

In the previous section, we have mentioned two main approaches to minimize the risk. The prior probability should be known to minimize the risk $R$ in equation (12) for Bayesian approach. We introduce *BayesShrink* threshold which minimizes the Bayes risk, unlike the universal threshold or $SURE$ threshold obtained by minimax rule.

From equation (12), Bayes risk can be written as

$$R(\lambda_t) = E(\|\hat{\mathbf{x}} - \mathbf{x}\|^2) = E_{\mathbf{x}} E_{\mathbf{y}|\mathbf{x}}(\|\hat{\mathbf{x}} - \mathbf{x}\|^2),$$

where $\hat{\mathbf{x}} = \Theta_S(\mathbf{y}, \lambda_t)$ and $\mathbf{y}|\mathbf{x} \sim N(0, \sigma^2)$. We also assume that the prior pdf of $\mathbf{x}$ is generalized Gaussian distribution (GGD; see the definition in Appendix). When $\beta = 2$, GGD becomes Gaussian distribution. Then the optimal threshold can be obtained by solving $\lambda_t^* = \arg\min_{\lambda_t} R(\lambda_t)$. Then when $\beta = 2$ in GGD,

$$\begin{aligned} R(\lambda_t) &= E_{\mathbf{x}} E_{\mathbf{y}|\mathbf{x}}(\|\hat{\mathbf{x}} - \mathbf{x}\|^2) \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (\Theta_S(\lambda_t, y) - x)^2 p(y|x) p(x) dy dx \\ &= \sigma^2 w\left(\frac{\sigma_x^2}{\sigma^2}, \frac{\lambda_t}{\sigma}\right), \end{aligned}$$

where

$$w(\sigma_x^2, \lambda_t) = \sigma_x^2 + 2(\lambda_t^2 + 1 - \sigma_x^2)\Phi\left(\frac{\lambda_t}{\sqrt{1+\sigma_x^2}}\right) - 2\lambda_t(1+\sigma_x^2)\phi(\lambda_t, 1+\sigma_x^2)$$

when $\phi(x, \sigma^2)$ is $N(0, \sigma^2)$ and $\Phi(x) = \int_x^{\infty} \phi(t, 1) dt$. By numerical calculation, the approximated threshold close to the optimal threshold $\lambda_t^*$ is

$$\lambda_t = \frac{\sigma^2}{\sigma_x}. \tag{18}$$

*BayesShrink* is simple and effective. The experimental results show that it is comparable to *SureShrink*. *BayesShrink* is also applied in [15] and combined with SI wavelet

transforms and *context modeling* which considers the related coefficients to the coefficient to be thresholded. This adaptive approach, called *AdaptShrink*, improves the denoising results by the threshold in equation (18).

### 3.3.4 Cross-validation

Cross-validation (CV) is a classical yet effecitve method to estimate the smoothing parameters which minimize MSE. CV has been widely used for evaluating the optimality of a smoothing parameter such as linear regression of spline smoothing [28, 48]. It has been employed for thresholding approach in the wavelet domain due to the properties of minimizing MSE and the asymptotic behavior [72, 100, 101, 52].

The main idea is so-called *leaving-out-one* scheme, which means that we measure and predict the value of one point by all the rest of points. For every element in noisy observation $\mathbf{y}$, we predict $\tilde{y}_{\lambda_k}$, the estimate of $y_k$ as a measure for the optimality of the choice of the threshold. In order to express the compromise for all the elements in $\mathbf{y}$, CV function can be defined as follwos:

$$CV = \frac{1}{N}\sum_{k=1}^{N}(y_k - \tilde{y}_{\lambda_k})^2.$$

Then we can get $CV \approx \frac{1}{N}\sum_{k=1}^{N}\alpha_k^2(\lambda)(y_k - \tilde{y}_{\lambda_k})^2$ with $\alpha_k^2(\lambda) = 1/(1 - A_{kk})$, where $y_k - \tilde{y}_{\lambda_k} = \frac{y_k - y_{\lambda_k}}{1-a_k^*}$ and $a_k^* = \frac{y_{\lambda_k} - \tilde{y}_{\lambda_k}}{y_k - \tilde{y}_{\lambda_k}} \approx \frac{\partial y_{\lambda_k}}{\partial y_k} = A_{kk}$.

By defining the cumbersome matrix $A$, we can define generalized cross validation (GCV) function. In [101], Weyrich and Warhola defined the GCV criterion in a simple formula as follows:

$$\begin{aligned}\lambda_t &= \arg\min_{\lambda} GCV(\lambda) \\ &= \arg\min_{\lambda} \frac{\|\mathbf{y} - \mathbf{y}_\lambda\|^2/N}{(\#\{k : |y_k| \le \lambda\}/N)^2},\end{aligned} \tag{19}$$

where $\mathbf{y}_\lambda$ is a modified wavelet subband by a threshold $\lambda$ from $\mathbf{y}$.

GCV threshold, $\lambda_t$ is an asymptotically optimal procedure. In fact, Jansen proved the

asymptotic behaviour

$$\lim_{N\to\infty} \frac{R(\lambda_t)}{R(\lambda^*)} = 1,$$

where $\lambda^* = \arg\min MSE(\lambda)$. Like *SureShrink*, GCV is a threshold approach which estimates the minimum MSE threshold by noisy observation and the threshold is calculated in each subband. In fact, there is a relationship between *SURE* and GCV like $GCV(\lambda) \approx SURE(\lambda) + \sigma^2 \approx MSE(\lambda) + \sigma^2$. Also, it could achieve the fast tresholding by formularizing CV in a general way as shown in equation (19). Computational complexity is less than the orthogonal wavelet transforms. More GCV thresholding applications can be found in [50, 51]

## 3.4 Shrinkage Approaches

As defined before, equation (10) is the form of shrinkage approaches. This can be simply rewritten as

$$\hat{x} = \gamma y, \tag{20}$$

where $0 \leq \gamma \leq 1$ is a shrinkage factor. This means that a denoised coefficient is *shrinked* by a linear operator $\gamma$.

We introduce some shrinkage algorithms having the form of $\hat{x} = \gamma y$. They include linear minimum mean squared error (MMSE) approach with local variance [70], bivariate shrinkage approach [84], *NeighBlock* based on neighbouring dependency [11], and the approaches using geometric priors such as Markov random field (MRF) [65] and hidden Markov models (HMMs) [29].

### 3.4.1 Linear MMSE estimator

Mihcak et al. in [70] proposed linear MMSE estimation procedure using maximum likelihood (ML) and maximum *a posteriori* (MAP) estimates for local variances. Under the assumption of *iid* Gaussian noise and in the sense of MSE, the optimal predictor for clean

wavelet coefficient in $k$-th location $x_k$ is linear and given by

$$\hat{x}_k = y\frac{\hat{\sigma}^2_{x_k}}{\hat{\sigma}^2_{x_k} + \sigma^2} = y\left(1 - \frac{\sigma^2}{\hat{\sigma}^2_{x_k} + \sigma^2}\right)_+ . \tag{21}$$

The estimate of the local variance $\sigma^2_{x_k}$ can be computed from an approximate ML estimator:

$$\hat{\sigma}^2_{x_k} = \arg\max_{\sigma^2_{x_k}} \prod_{j \in N_k} P(y_j|\sigma^2_{x_k}) = \left[\frac{1}{M}\sum_{j \in N_k} y_j^2 - \sigma^2\right]_+$$

and an approximate MAP estimator:

$$\hat{\sigma}^2_{x_k} = \arg\max_{\sigma^2_{x_k}} \left(\prod_{j \in N_k} P(y_j|\sigma^2_{x_k})\right) p(\sigma^2_{x_k}) = \left[\frac{M}{4\lambda}\left(-1 + \sqrt{\frac{8\lambda}{M^2}\sum_{j \in N_k} y_j^2}\right) - \sigma^2\right]_+ ,$$

where the exponential prior distribution function $p(\sigma^2_{x_k}) = \lambda e^{-\lambda\sigma^2}$, which is empirically chosen.

Linear MMSE is in fact equivalent to Wiener filter [59]. Li and Orchard also used MMSE and overcomplete expansion of wavelet transforms to get better performance for image denoising [60]. And there have been many discussions about Wiener filtering in the wavelet domain such as [68, 22, 53, 105].

### 3.4.2 Bivariate shrinkage using level dependency

In [86], Shapiro proposed an image coding approach called *zerotree*. He observed the parent-child dependencies of subbands in the wavelet domain and proposed the scanning order of the subbands for encoding a significance map. This level dependency is useful information for MRA. We can see many wavelet-based image processing applications which utilizes cross-scale dependecies such as image compression [8] and denoising [15, 74, 84].

Sendur and Selesnick in [84] proposed a bivariate shrinkage function using MAP estimator and the statistical dependency between a wavelet coefficient and its parent coefficient. If $x_2$ represents the parent of $x_1$, then $y_1 = x_1 + z_1$ and $y_2 = x_2 + z_2$, where $y_1$

and $y_2$ are noisy coefficients of $x_1$ and $x_2$ and $z_1$ and $z_2$ are noise. So $\mathbf{y} = \mathbf{x} + \mathbf{z}$, where $\mathbf{x} = (x_1, x_2)^t$, $\mathbf{y} = (y_1, y_2)^t$ and $\mathbf{z} = (z_1, z_2)^t$. By MAP estimator of $\mathbf{x}$,

$$\begin{aligned} \hat{\mathbf{x}} &= \arg\max_{\mathbf{x}} p_{\mathbf{x}|\mathbf{y}}(\mathbf{x}|\mathbf{y}) \\ &= \arg\max_{\mathbf{x}} [p_{\mathbf{z}}(\mathbf{y} - \mathbf{x}) \cdot p_{\mathbf{x}}(\mathbf{x})] \end{aligned}$$

In this case, we can assume that $p_{\mathbf{z}}(\mathbf{y} - \mathbf{x}) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{z_1^2+z_2^2}{2\sigma^2}\right)$ since $z_1$ and $z_2$ are *iid* with $N(0, \sigma^2)$. To find a prior distribution $p_{\mathbf{x}}(\mathbf{x})$, they observed the empirical model and proposed four different models for $p_{\mathbf{x}}(\mathbf{x})$. Among them, the models they could derive the shrinkage functions are *Model 1*:

$$p_{\mathbf{x}}(\mathbf{x}) = \frac{3}{2\pi\sigma_x^2} \exp\left(-\frac{\sqrt{3}}{\sigma_x}\sqrt{x_1^2 + x_2^2}\right),$$

which is circularly symmetric and related to the family of spherically invariant random processes (SIRPs) [79, 106] and *Model 2*:

$$p_{\mathbf{x}}(\mathbf{x}) = C \exp\left(-\left[a\sqrt{x_1^2 + x_2^2} + b(|x_1| + |x_2|)\right]\right),$$

which is an extended version of *Model 1* combining independent Laplacian model. $C$ is the normalization constant. By solving the equations, we can get the following shrinkage functions:

$$\hat{x} = \hat{x_1} = y_1 \left(1 - \frac{\sqrt{3}\sigma^2}{\sigma_x\sqrt{y_1^2 + y_2^2}}\right)_+ \tag{22}$$

for *Model 1* and

$$\hat{x} = \hat{x_1} = \text{soft}(y_1, b\sigma^2)\left(1 - \frac{a\sigma^2}{R}\right)_+ \tag{23}$$

for *Model 2*, where $\text{soft}(y, \lambda)$ is a soft thresholding function with a threshold $\lambda$ and $R = \sqrt{\text{soft}(y_1, b\sigma^2)^2 + \text{soft}(y_2, b\sigma^2)^2}$. The bivariate shrinkage function can be given by a simple formula taking into account the interscale dependency. This approach achieves both simplicity and efficiency. Later Sendur and Selesnick achieve better results by using local variance estimation in the bivariate shrinkage function [85].

### 3.4.3 Neighbour dependency

A large wavelet coefficient will probably have large wavelet coefficients at its neighbours [86][88]. This is because even when the coefficients are uncorrelated, there are still significant higher order correlations, like a strong positive covariance in amplitude between neighbour coefficients. Many recent works for wavelet-based denoising such as [29, 70, 11, 87, 18] have paid attention to this point.

Cai and Silverman in [11] proposed a simple and effective approach for a 1D signal by incorporating the neighbouring coefficients. Their *block thresholding* method, called *NeighBlock* and *NeighCoeff*, can be simply described in the following stages.

1. Decompose the noisy signal into the orthogonal wavelet domain.
2. For each decomposition level $j$, we define a small block whose length is $M_0 = [\log N/2]$ for each coefficient in the level $j$.
3. The block is extended to each direction by $M_1 = \max(1, [M_0/2])$ and therefore new block length is $M = M_0 + 2M_1$. The new block $N_k$ with the block length $M$ consists of the thresholded wavelet coefficient and its neighbours.
4. A desired estimate $\hat{x}_k$ of $x_k$ is defined as follows:

$$\hat{x}_k = y_k \left(1 - M\lambda^2/S_k^2\right)_+, \tag{24}$$

   where $S_k^2 = \sum_{s_l \in N_k} s_l^2$, and $\lambda$ is a threshold.
5. Reconstruct the denoised data using inverse discrete wavelet transform from the thresholded coefficients.

While *NeighBlock* uses an appropriate block length defined by the signal size $N$ and $\lambda = 4.50524...$, which is the solution of $\lambda - \log\lambda = 3$, *NeighCoeff* is a specific case of *NeighBlock* when $M_0 = M_1 = 1$, $M = 3$, and $\lambda = \sqrt{\frac{2}{3}\sigma^2 \log N}$. So *NeighCoeff* uses three coefficients including the coefficient being thresholded itself and two adjacent coefficients to both the left and the right. $\lambda$ is obtained by Donoho's universal threshold [27], but it is different due to the block length.

The experimental results showed apparent advantages over the traditional term-by-term wavelet denoising. The 2D extension of *NeighCoeff* is straightforward. In Chapter 4, we do not only extend but also improve the algorithm for image denoising.

### 3.4.4 Markov random field

Malfait and Roose proposed an image denoising algorithm using Markov random field image model as *a priori* in 1997 [65]. Also Pizurica et al. considered a joint inter- and intrascale statistical model in [74] and improved the approach by Malfait and Roose. Some other approaches and application using MRF can be found in [80, 98, 104, 73]. In this section, we explain the basic idea and algorithm by Malfait and Roose [65].

From equation (20), we can assume it as the binary operation when $\gamma = 0$ if $y$ is governed by noise and $\gamma = 1$ if $y$ is clean enough. Under this shrinkage form, we consider two measures : a simple approximation to the local Hölder exponent and *a priori* geometrical knowledge. These can be combined in a Bayesian framework. We assume that $m_k$ is the measure that denotes the noisy level of the $k$-th wavelet coefficient $y_k$ defined by local Hölder exponent approximation and $\mathbf{m}$ is a set (vector) consisting of $m_k$ in a subband. In this case, we can upgrade equation (20) as $\hat{x}_k = P(\gamma_k = 1|\mathbf{m})y_k$ instead of the binary operation. By Bayes' rule, $P(\boldsymbol{\gamma}|\mathbf{m}) = P(\mathbf{m}|\boldsymbol{\gamma})P(\boldsymbol{\gamma})/P(\mathbf{m})$, where $\boldsymbol{\gamma}$ is a set (vector) consisting of $\gamma_k$ in a subband. If we assume that $P(\mathbf{m})$ is uniform, $P(\boldsymbol{\gamma}|\mathbf{m}) \propto P(\mathbf{m}|\boldsymbol{\gamma})P(\boldsymbol{\gamma})$. Based on the relation between MRF's and Gibbs probability functions, *a priori* probability is

$$P(\boldsymbol{\gamma}) = 1/Z \exp\left(-V(\boldsymbol{\gamma})\right)$$

where $N_k$ is the neighbourhood system of $\gamma_k$, $Z = \sum \exp\left(-V(\boldsymbol{\gamma})\right)$, and $V(\boldsymbol{\gamma}) = \sum V_{N_k}(\boldsymbol{\gamma})$ when $V_{N_k}(\boldsymbol{\gamma}) = \sum V_{k,l}(\gamma_k, \gamma_l)$ with $V_{k,l}(\gamma_k, \gamma_l) = \begin{cases} -c, & \text{if} \gamma_k = \gamma_l, \\ +c, & \text{if} \gamma_k \neq \gamma_l. \end{cases}$ $c$ is a parametr of the *a priori* probability.

Based on these ideas, the MRF denoising algorithm can be summarized as follows:

1. Decompose the noisy image into the wavelet domain.
2. For each subband in the decomposition level $j$,

(a) Compute approximations to the local Hölder exponent $m_k$. For example,

$$m_k = \frac{1}{\text{depth}} \sum_{j=J-\text{depth}-1} \left| \frac{y_j}{y_{j+1}} \right|.$$

(b) Generate an initial mask by applying the threshold $\lambda$ to $m_k$ such that $E(\lambda) \simeq \sigma^2$.

(c) Run the stochastic sampling procedure with as probability

$$P(\gamma|\mathbf{m}) \propto \exp\left[-\left(\alpha \sum V(m_k|\gamma_k) + \beta \sum V_{N_k}(\gamma)\right)\right]$$

accounting for local Hölder exponents and *a priori* model. Sampling yields approximations to the marginal probabilities $P(\gamma_k = 1|\mathbf{m})$ for all $k$.

(d) Estimate a clean coefficient $x_k$ by the following shrinkage rule :

$$\hat{x}_k = y_k P(\gamma_k = 1|\mathbf{m}).$$

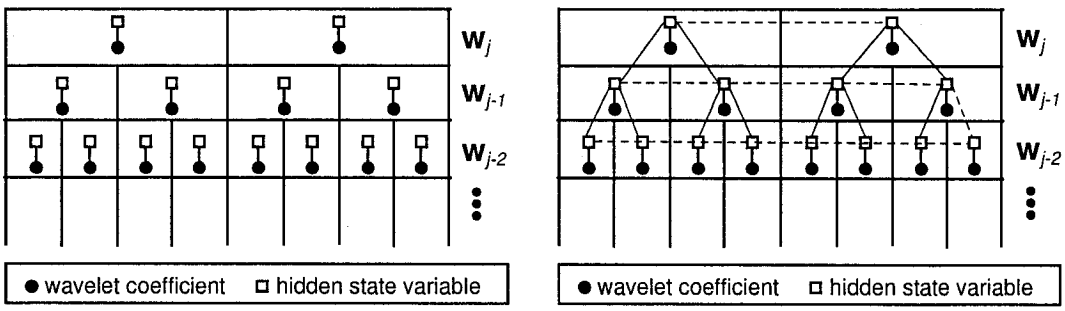
3. Reconstruct the denoised data using inverse discrete wavelet transform from the thresholded coefficients.

Since MRF eyes geometrical dependency, it is also related the method which considers neighbour dependency in the previous section. Also, the method uses Bayesian framework using MRF prior models. In practice, the experimental results for the test images in [65] and [74] show that this dependency helps the image quality to be improved and the image details to be preserved better.

### 3.4.5 Hidden Markov models

There are some works based on wavelet-domain hidden Markov models (HMMs) such as [29, 23, 41, 42, 82]. Based on HMMs theory, Crouse et al. in [29] proposed the hidden Markov tree (HMT) framework which enables us to concisely model the non-Gaussian statistics of individual wavelet coefficients and capture statistical dependencies between coefficients.

The HMT framework has two main features of HMMs for richness and flexibility, called




(a) Independent mixture (IM) model (b) HMT model

Figure 12: Mixture density Models with hidden state variables for 1D signal

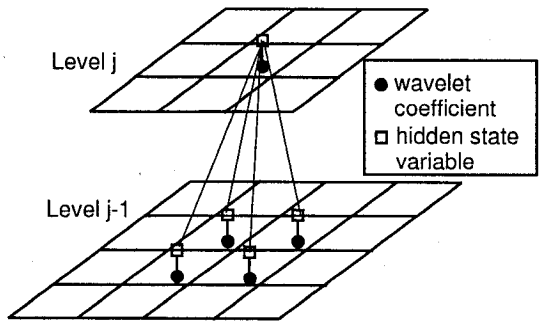



Figure 13: Markovian dependency between levels in HMT model for 2D image (quadtree)

*mixture densities* and *probabilistic graphs*. Mixture densities are obtained in terms of the marginal probability having non-Gaussian nature of wavelet coefficients. The probability density is modeled with a *hidden state variable* for each wavelet coefficient. And Markovian dependencies between the hidden variables are featured by the probabilistic graphs.

Figure 12 shows geometric ideas of HMT model. *independent mixture* (IM) model illustrated in Figure 12(a) considers only the wavelet state variables regardless of Markovian dependencies. HMT model is a tree-structured graph as shown in Figure 12(b), which connects vertically the hidden state variables between the levels while hidden Markov chain (HMC) model connects the state variables horizontally. When 2D image is used for input data, a quadtree can be formed like Figure 13.

Based on the HMM scheme and Bayesian estimation, the estimate of clean wavelet

coefficient can be formulated as folows:

$$\hat{x}_k = E(x_k|\mathbf{y}, \boldsymbol{\theta}) = \sum_m p(S_k = m|\mathbf{y}, \boldsymbol{\theta}) \times \frac{\sigma_{k,m}^2}{\sigma_{k,m}^2 + \sigma^2} y_k, \tag{25}$$

where $\boldsymbol{\theta}$ is a vector of HMM parameters that characterize the wavelet coefficients and $S_k$ is a hidden state of a observed wavelet coefficient $y_k$. So it is important to obtain the parameters $\boldsymbol{\theta}$ and then determine the most likely sequence of hidden states from $\boldsymbol{\theta}$. The parameters can be estimated by ML principles and they can be obtained by a training approach, called *expectation-maximization* (EM) (also called *Baum-Welch*) algorithm.

HMT method originally proposed by Crouse et al. is one of the methods which consdier the interdependency in the wavelet domain. The experimental results of them and the following works show that HMT method gives better results than IM approach. Choi et al applied DT CWT to image and obtained higher PSNR values than usual wavelet transforms in [23].

## 3.5 Other Estimation Approaches

If the denoising approaches are not classified in the thresholding or shrinkage rule, they are included in this category. In this section, we present a recently proposed efficient algorithm based on Gaussian scale mixtures [78].

### 3.5.1 Gaussian scale mixtures

For Bayesian denoising approaches, there have been many suggestions for prior distribution. One of them is Gaussian scale mixtures [2]. Gaussian mixture model has been used for denoising by several researchers [19, 29, 23, 78]. This model is quite similar to the empirical model of natural images in the wavelet domain.

In this section, we introduce the algorithm proposed in 2003 by Portilla et al. [78]. They have used multivariate model using neighbouring coefficients. In this case, a random vector $\mathbf{x}$ of Gaussian scale mixture could be $\mathbf{x} = \sqrt{a}\mathbf{u}$, where $\mathbf{u}$ is a zero-mean Gaussian

vector and $\sqrt{a}$ is an independent positive scalar random variable. So the $d$-dimensional vector form of equation (11) becomes as follows:

$$\mathbf{y} = \mathbf{x} + \mathbf{z} = \sqrt{a}\mathbf{u} + \mathbf{z}. \tag{26}$$

Based on this model, the algorithm obtains the estimate $E(x_c|\mathbf{y})$, where $c$ is the index of the reference coefficient within the neighborhood vector. Their denoising algorithm can be summarized as follows:

1. Decompose the noisy image into the wavelet domain.
2. For each subband in the decomposition level $j$,
   (a) Compute neighbourhood noise covariance, $\mathbf{C_z}$, from the image-domain noise covariance.
   (b) Estimate noisy neighbourhood covariance, $\mathbf{C_y}$.
   (c) Estimate $\mathbf{C_u}$ from $\mathbf{C_u} = \mathbf{C_y} - \mathbf{C_z}$ if $E(a) = 1$.
   (d) Compute $\mathbf{\Lambda}$ and $\mathbf{M}$ by the following ways: $\mathbf{\Lambda}$ and $\mathbf{Q}$ are eigenvalue and eigenvector expansion of $\mathbf{S}^{-1}\mathbf{C_u}\mathbf{S}^{-t}$, where $\mathbf{C_z} = \mathbf{S}\mathbf{S}^t$, and $\mathbf{M} = \mathbf{S}\mathbf{Q}$.
   (e) For each neighbourhood,
      i. For each value $a$ in the integration range, compute $E(x|\mathbf{y}, a)$ and $p(\mathbf{y}|a)$ as follows:
      $$E(x|\mathbf{y}, a) = \sum_{i=1}^{d} \frac{a m_{cn} \lambda_i v_i}{\lambda_i + 1}$$
      and
      $$p(\mathbf{y}|a) = \frac{\exp\left[-\frac{1}{2}\sum_{i=1}^{d} \frac{v_i^2}{a\lambda_i+1}\right]}{\sqrt{(2\pi)^d |\mathbf{C_z}| \prod_{i=1}^{d} (a\lambda_i + 1)}},$$
      where $m_{ij}$ represents an $i$-th row and $j$-th column element of $\mathbf{M}$, $\lambda_i$ is a diagonal element of $\mathbf{\Lambda}$, and $v_i$ is an element of a vector $\mathbf{v} = \mathbf{M}^{-1}\mathbf{y}$.
      ii. Compute $p(a|\mathbf{y}) = \frac{p(\mathbf{y}|a)p_a(a)}{\int_0^\infty p(\mathbf{y}|\alpha)p_a(\alpha)d\alpha}$ using $p_a(a) \propto \frac{1}{a}$ by Jeffrey's prior [7].
      iii. Compute $E(x|\mathbf{y}) = \int_0^\infty p(a|\mathbf{y})E(x|\mathbf{y}, a)da$ numerically.
3. Reconstruct the denoised data using inverse discrete wavelet transform from the thresholded coefficients.

To our knowledge, this approach gives the best image denoising results to date, especially in terms of MSE, when the steerable pyramid with 8 orientations is employed as

the wavelet transform. However, the algorithm is not easy to implement and this requires much more time and storage complexity than the simple thresholding approaches. Also when other wavelet transforms such as Daubechies family are used, the results are not as good as the steerable pyramid as illustrated in the later chapter.

### 3.5.2 Multiwavelet thresholding

As explained in the last chapter, multiwavelet has good properties the scalar wavelet does not have. Multiwavelet transform generates multiple wavelet coefficients in the same location. For example, if we apply GHM to 2D image, it produces four wavelet coefficients having close dependency in the same location by the multiple filters (two in this case) when the usual scalar WT generates one wavelet coefficient in a subband. So the approach using these multiple coefficients can be considered only for multiwavelet. Since Downie and Silverman paid attention to this idea [38], there have been several approaches proposed for both signal and image [9, 16, 18, 17, 97, 63, 4, 5, 49, 40]. We describe this method as an extra approach because this approach is only for multiwavelet family.

The first and easy approach we can think is to apply the term-by-term threshold as we described in the previous sections, e.g. universal threshold, *SURE*, etc. However, we need to consider the correlated multiple coefficeients in order to take advantage of the filtering dependency in the multiple wavelet domain. Downie and Silverman in [38] proposed a *vector* thresholding approach which considers the multiple coefficients in the same location as the elements of a vector. As mentioned in the previous chapter, the multiwavelet transform with an appropriate prefilter generates the correlated multiple wavelet coefficients. So equation (11) can be rewritten in a vector form as $\mathbf{y}_k = \mathbf{x}_k + \mathbf{z}_k$ and $\mathbf{z}_k$ has multivariate normal distribution $N(\mathbf{0}, \Sigma)$, where $\mathbf{0}$ is a zero-mean vector and $\Sigma$ is a covariance matrix. Then the quantity $\theta_k^2$ can be defined as follows:

$$\theta_k^2 = \mathbf{y}_k^t V^{-1} \mathbf{y}_k. \tag{27}$$

$\theta_k^2$ has $\chi_L^2$ distribution. And $\theta_k^2$ is compared with the threshold $\lambda_t$ such as a universal threshold or *multivariate universal threshold* defined as $\lambda_t^2 = \sigma^2(2\log N + (L-2)\log\log N)$, where $L$ is a number of filters in multiwavelet. In this case, the hard thresholding rule is $\hat{\mathbf{x}}_k = \begin{cases} \mathbf{x}_k \cdot I, & \text{if } \theta_k \geq \lambda_t, \\ \mathbf{0}, & \text{otherwise,} \end{cases}$ and the soft thresholding rule is $\hat{\mathbf{x}}_k = \mathbf{x}_k \left(1 - \frac{\lambda_t^2}{\theta_k^2}\right)_+$.

Bui and Chen in [9] extended this idea and performed various experiments. They showed that multiwavelet thresholding approach is promising and better than the scalar wavelet. They also improved the denoising performance by employing TI scheme to multiwavelet. In [18], they tried to use neighbouring coefficients by substituting the quantity $\theta_k^2$ with the sum of the $k$-th and its adjacent quantities $S_k^2 = \theta_{k-1}^2 + \theta_k^2 + \theta_{k+1}^2$. And this approach produces lower MSE values.

For 2D image, multiwavelet transform generates $L^2$ correlated wavelet coefficients in the same location. Some extended approaches have been used in [97, 4]. Multiwavelet thresholding approaches can maximize the merits of multiwavelet properties such as symmetry, orthogonality, and vanishing moments as described in the previous chapter.

# Chapter 4

# Wavelet Shrinkage using Level and Neighbour Dependency

In the following three chapters, we first propose two image denoising approaches and then present various experimental results and performance evalutions. These approaches have been submitted to journals in [21, 20].

In this chapter, we suggest simple but very efficient wavelet shrinkage rules based on the correlations between the related neighbouring and parent wavelet coefficients and the coefficient to be thresholded.

## 4.1 Shrinkage approach based on interdependency

In the spatial domain, it is well-known that an adaptive Wiener method based on estimation from local information is very efficient for digital image enhancement [59]. In the wavelet domain, despite the weak correlation properties of the wavelet transform, as pointed out in the introduction, there still exist significant residual statistical dependencies between neighbour wavelet coefficients. Our goal is to exploit this dependency to improve the estimation of a coefficient given its noisy observation and a *context* (spatial and scale neighbours).

Recently, there have been several works which try to use the context in the wavelet

domain [11][15][65][70][85]. Among them, Cai and Silverman in [11] proposed a simple and effective approach for a 1D signal by incorporating the neighbouring coefficients. Their *block thresholding* method in [11], called *NeighBlock* and *NeighCoeff*, can be simply described in the following.

Suppose that $w_{j,k}$ is a wavelet coefficient of a noisy signal $B$ with size $N$ and $x_{j,k}$ is the wavelet coefficient of the signal without noise, then $w_{j,k} = x_{j,k} + \sigma z_{j,k}$, where $\sigma^2$ is the variance and $z_{j,k}$ is the noise with $N(0,1)$. To get a desired estimate $\hat{x}_{j,k}$ of $x_{j,k}$, their shrinkage rule is defined as follows:

$$\hat{x}_{j,k} = w_{j,k}\left(1 - M\lambda^2/S_{j,k}^2\right)_+, \tag{28}$$

where

$$S_{j,k}^2 = \sum_{s_l \in N_{j,k}} s_l^2,$$

$N_{j,k}$ is a block consisting of the thresholded wavelet coefficient and its neighbours, $M$ is the block length and $\lambda$ is a threshold. While *NeighBlock* uses an appropriate block length defined by the signal size $N$ and $\lambda = 4.50524...$, which is the solution of $\lambda - \log\lambda = 3$, *NeighCoeff* is a specific case of *NeighBlock* when $M = 3$ and $\lambda = \sqrt{\frac{2}{3}\sigma^2 \log N}$. So *NeighCoeff* uses three coefficients including the coefficient being thresholded itself and two adjacent coefficients to both the left and the right. $\lambda$ is obtained by Donoho's universal threshold [27], but it is different due to the block length. As shown by Cai and Silverman, both *NeighBlock* and *NeighCoeff* have excellent asymptotic properties and attain the optimal rate of convergence in the Besov sequence space. This means that the least upper bound of the expected denoised error is close to zero when the length of the signal tends to infinity just like Donoho's universal threshold.

In the next three sections, three different wavelet shrinkage approaches are presented for a 2D image. We call them *NeighShrink*, *NeighLevel* and *NeighSure*. They are based on Cai and Silverman's shrinkage equation (28) combined with more enhanced factors such as scale neighbours.

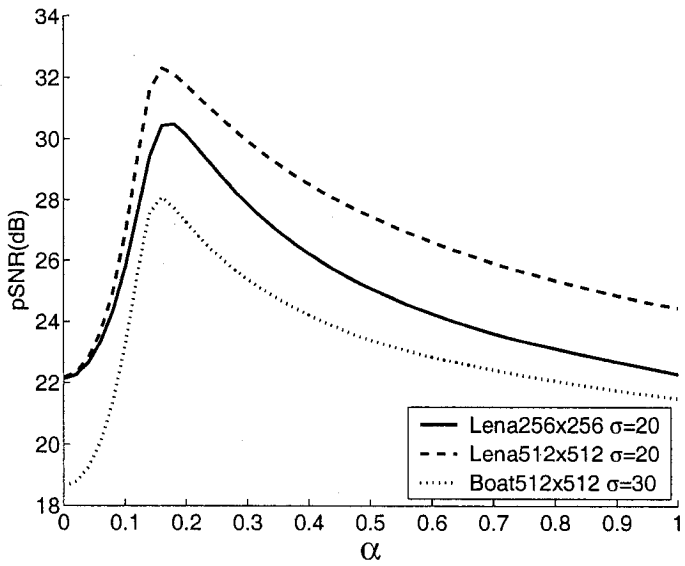



Figure 14: Peformance change by a threshold $\lambda^* = \alpha\lambda$ for different kinds and sizes of images when $NeighShrink$ ($M = 5$) is applied using dual tree complex wavelet (DT CWT) proposed in [54].

## 4.2 Optimal Threshold and *NeighShrink*

One of the simplest wavelet shrinkage rules for an $N \times N$ image is the universal threshold $\lambda = \sqrt{2\sigma^2 \log N^2}$ suggested by Donoho in [33]. The universal threshold grows asymptotically and removes more noisy coefficients as $N$ tends to infinity. The universal threshold is designed for smoothness rather than for minimizing the errors. So $\lambda$ is more meaningful when the signal is sufficiently smooth or the length of the signal is close to infinity. Natural image, however, is usually neither sufficiently smooth nor composed of infinite number of pixels. In fact, if we suppose that an optimal threshold which minimizes MSE (or maximizes PSNR), $\lambda^*$, is $\alpha\lambda$, $\alpha$ is always much less than 1.0 for natural image as shown in Figure 14. Especially we got very similar $\alpha$ value for different kinds and sizes of images when we applied soft thresholding rule.

We can extend *NeighCoeff* to 2D image in a straightforward way.

$$\hat{x}_{j,k} = w_{j,k} \left( 1 - M^2 \lambda^{*2} \Big/ \sum_{s_l \in N_{j,k}} s_l^2 \right)_+ , \tag{29}$$

where $\lambda^* = \alpha\lambda$ and $N_{j,k}$ is an $M \times M$ window which consists of the thresholded coefficient and all its neighbours. and we call it *NeighShrink*. In the experiments, we investigate the optimal threshold by varying $\alpha$ and $M$.

## 4.3 *NeighSure*

Although *NeighShrink* yields very good performance, a constant $\alpha$ must be chosen empirically. If we substitute $\alpha\lambda$ in equation (29) by an optimal threshold with minimum risk, we may not need to use the universal threshold with a constant $\alpha$ to build a robust shrinkage method. One of the existing optimal thresholding methods is *SureShrink*, which uses an adaptive threshold by minimizing *SURE* for each wavelet decompsition level [36]. When $\mathbf{w}_j$ is an $n \times n$ wavelet subband in level $j$, *SURE* threshold $\lambda^*$ is

$$\begin{aligned} \lambda^* &= \arg\min_{\lambda} SURE(\mathbf{w}_j, \lambda) \\ &= \arg\min_{\lambda} \left[ (n^2 - 2 \cdot \#\{k : |w_{j,k}| \leq \lambda\} + \sum_{k}^{n^2} \min(|w_{j,k}|, \lambda)^2 \right]. \end{aligned} \tag{30}$$

Then a new shrinkage rule, called $NeighSure$, can be obtained from equation (29) by substituting nonparametric threshold $\lambda^*$ in equation (30) instead of the universal threshold.

## 4.4 *NeighLevel*

Here we suggest one more method, which is called *NeighLevel* so as to distinguish it from the other two methods.

For previous two methods, we have considered neighbour dependency. There is another possible correlation between the wavelet coefficients lying in different decomposition levels. The statistical correlation between parent and child coefficients have been widely recognized in image coding and denoising [15][29][78][84][85][86] since *zerotrees* were introduced by Shapiro [86]. Parent and child have interdependency similar to neighbours. Therefore, if we can utilize neighbours spreaded both vertically and horizontally as shown

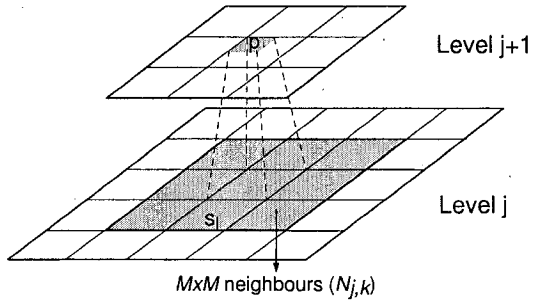



Figure 15: *Context*

in Figure 15, then a better performance can be expected.

According to this idea, we propose the following criterion by applying the coefficients in the coarser level to equation (29):

$$\hat{x}_{j,k} = w_{j,k}\left[1 - (M^2+1)\lambda^{*2} \Big/ \Big(\sum_{s_l \in N_{j,k}} s_l^2 + p^2\Big)\right]_+, \tag{31}$$

where $\lambda^* = \alpha\lambda$ like *NeighShrink*, $s_l$ denotes the coefficient to be thresholded and its neighbours in an $M \times M$ window, and $p$ is a parent of the coefficient to be thresholded, which is the coefficient matched in the coarser level (see Figure 15). In equation (31), it should be noted that a normalized factor, $M^2 + 1$, is used which is the number of correlated elements in the context. By this rule, the effect of the local variance from the parent level is considered as well as from the current level.

## 4.5 Analysis of shrinkage functions

The three proposed shrinkage rules have a common feature: Local sample coefficients are taken from an $M \times M$ window $N_{j,k}$ for *NeighShrink* and *NeighSure* and $N_{j,k}$ and a parent coefficient $p$ for *NeighLevel* as shown in Figure 15. If we assume that the mean of the sample coefficients is zero, the local variance of $w_{j,k}$ can be defined as $\sigma^2_{w_{j,k}} = \sum a_i^2/m$, where $a_i$ is sample coefficient and $m$ is the number of local samples. In this case, we can notice that both equations (29) and (31) become $\hat{x}_{j,k} = w_{j,k}\big(1 - \lambda^{*2}/\sigma^2_{w_{j,k}}\big)_+$.

In [70], linear MMSE-like estimator of $x_{j,k}$ is given by $\hat{x}_{j,k} = w_{j,k}\frac{\hat{\sigma}^2_{x_{j,k}}}{\hat{\sigma}^2_{x_{j,k}}+\sigma^2}$, where $\hat{\sigma}^2_{x_{j,k}}$ is an estimated variance of $x_{j,k}$. They have computed an approximate ML estimator for $\hat{\sigma}^2_{x_{j,k}}$ as $\left(\frac{1}{M^2}\sum_{s_l \in N_{j,k}} s_l^2 - \sigma^2\right)_+$. In other words, $\hat{\sigma}^2_{x_{j,k}} = \left(\sigma^2_{w_{j,k}} - \sigma^2\right)_+$. Therefore, linear MMSE estimator can be rewritten as $\hat{x}_{j,k} = w_{j,k}\left(1 - \sigma^2/\sigma^2_{w_{j,k}}\right)_+$. Since $\lambda^* \propto \sigma$ for the universal threshold used in *NeighShrink* and *NeighLevel*, the shrinkage rule of linear MMSE estimator is the same as that of *NeighShrink* and *NeighLevel* if we assume that $\alpha\sqrt{2\log N^2} = 1$. In this sense, it could be said that *NeighLevel* uses the local variance considering both neighbour and level dependency. And our thresholds have strong connection with the estimator based on the probability density of wavelet coefficients and *prior* like linear MMSE estimator.

# Chapter 5

# Multivariate Statistical Modeling for Image Denoising

## 5.1 Introduction

In this chapter, we propose another new approach based on multivariate statistical modeling in the wavelet domain of natural images unlike the approach in the previous chapter. Different statistical models can be applied according to image characteristics since the model is from the probality density of natural images. Therefore, there could be many possible denoising rules. We first generalize our shrinkage rule by multivariate modeling and try to adapt the model to natural images. In addition, we present several rules as examples which can be derived from our approach.

Our fundamental concerns for wavelet image denoising lie in finding a general way to estimate the denoised coefficients. The classical yet powerful approach to estimate the denoised coefficients is based on Bayesian statistics. Chipman et al.[19] proposed *adaptive Bayesian wavelet shrinkage* (ABWS) method by fitting a wavelet-based model into a mixture of two Gaussian distributions. An adaptive wavelet thresholding method, called $BayesShrink$, was derived in a Bayesian framework using GGD by Chang et al. [14]. Sendur and Selesnick estimated an optimal threshold using MAP estimator (see Appendix)

by maximizing bivariate probability density function (pdf) whose elements include the parent coefficients [84]. MAP estimator was also used by Mihcak et al. for estimating the locally adaptive variance [70]. Very recently Portilla et al. presented an image denoising algorithm which is based on a Gaussian scale mixture (GSM) model for zero-mean Gaussian vector using an overcomplete multiscale oriented basis [78].

We describe here a general wavelet denoising approach which can estimate the optimal wavelet coefficients using information based on the multivariate statistical theory. We begin our derivation from MAP estimator used in [84]. Unlike their approach, we generalize our estimator in an arbitrary-dimensional space for any multivariate distribution model. The multivariate distributions of the original image can be estimated empirically from the sample image set. Then we define a specific multivariate pdf, called multivariate generalized Gaussian distribution (MGGD), which can closely fit into the sample distribution by its parameters. Multivariate model makes it possible that the estimated wavelet coefficients can be made correlated with related information such as their neighbours or coefficients in a different subband. Also it can be shown that some of the existing methods based on statistical modeling are subsets of our multivariate approach by changing the related elements and varying the distribution parameters. Experimental results show that our approach could achieve high quality image denoising by proper setting of the parameters. Among the existing image denoising methods which use Daubechies 8 wavelet filter, our results produce the highest PSNR values.

## 5.2 Bayesian Estimation for Multivariate Statistical Model

Let $A$ be a clean natural image with size $N \times N$, $B$ be a noisy image which can be expressed as $B = A + \sigma C$, and $C$ be zero-mean Gaussian white noise with variance $\sigma^2$, where $C \sim N(0, 1)$.

After performing multiresolution wavelet decomposition on $B$, we get the wavelet coefficient $y_{j,k}$, which is the $k$-th wavelet coefficient in $j$-th level for $B$. Then,

$$y_{j,k} = x_{j,k} + \sigma z_{j,k}, \tag{32}$$

where $x_{j,k}$ is the wavelet coefficient of $A$ in the same location as $y_{j,k}$.

Let $\mathbf{x}$ be a $d$-dimensional wavelet coefficient vector, $\mathbf{x} = (x_1, x_2, \cdots, x_d)^t$, where $x_1$ is the wavelet coefficient under consideration and $x_i$ $(i = 2, \cdots, d)$ are related coefficient, e.g. neighbours, parent and offsprings. Similarly a vector $\mathbf{y}$ can be defined for noisy image $B$ and we assume that $x_i$ and $y_i$ correspond to each other in both decomposition level and location. So we can rewrite equation (32) in vector form:

$$\mathbf{y}_{j,k} = \mathbf{x}_{j,k} + \sigma \mathbf{z}_{j,k}. \tag{33}$$

For the sake of simplicity, we omit subscripts $j, k$ in equation (33) in the rest of this chapter.

Our concern lies mainly in estimating the unknown wavelet coefficient vector $\hat{\mathbf{x}}$, and $\hat{\mathbf{x}}$ should be obtained only from $\mathbf{y}$ of the noisy image $B$. One of the ways to estimate $\hat{\mathbf{x}}$ is to use MAP estimator to maximize $p(\mathbf{x}|\mathbf{y})$. MAP estimator for $\hat{\mathbf{x}}$ can be obtained as follows:

$$\begin{aligned}
\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}\in\mathbb{R}^d} \ln p(\mathbf{x}|\mathbf{y}) \\
&= \arg\max_{\mathbf{x}\in\mathbb{R}^d} \ln \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})} \\
&= \arg\max_{\mathbf{x}\in\mathbb{R}^d} [\ln p(\mathbf{y}|\mathbf{x}) + \ln p(\mathbf{x}) - \ln p(\mathbf{y})] \\
&= \arg\max_{\mathbf{x}\in\mathbb{R}^d} [\ln p(\mathbf{y}|\mathbf{x}) + \ln p(\mathbf{x})]
\end{aligned} \tag{34}$$

In equation (34), $p(\mathbf{y})$ does not affect the result since it is only a constant. Therefore equation (34) shows that the optimal value $\hat{\mathbf{x}}$ with minimum probability error can be estimated by $p(\mathbf{y}|\mathbf{x})$ and $p(\mathbf{x})$,

First, from equation (33), $p(\mathbf{y}|\mathbf{x})$ is the multivariate Gaussian distribution with $N(\mathbf{0}, \Sigma_z = \sigma^2\mathbf{I})$ since Gaussian noise is independently and identically distributed for each element of

the vector. Hence the logarithm of the conditional pdf, $p(\mathbf{y}|\mathbf{x})$, is

$$\begin{aligned}
\ln p(\mathbf{y}|\mathbf{x}) &= \ln p(\mathbf{z}) \\
&= \ln \frac{1}{(2\pi)^{d/2}|\Sigma_z|^{1/2}} \exp\left\{-\frac{(\mathbf{y}-\mathbf{x})^t \Sigma_z^{-1}(\mathbf{y}-\mathbf{x})}{2}\right\} \\
&= \ln \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left\{-\frac{(\mathbf{y}-\mathbf{x})^t(\mathbf{y}-\mathbf{x})}{2\sigma^2}\right\} \\
&= -\frac{d}{2}\ln(2\pi\sigma^2) - \frac{(\mathbf{y}-\mathbf{x})^t(\mathbf{y}-\mathbf{x})}{2\sigma^2} \qquad (35)
\end{aligned}$$

Second, we need to find an appropriate statistical model for $p(\mathbf{x})$. This can be obtained empirically from the sample coefficients which are from the natural images in our case. If we inspect the wavelet coefficients of the sample images, their distribution looks close to Gaussian distribution. However, there exist better models which have closer approximation. In fact, many researchers suggested univariate probability model and even bivariate model for natural images so as to get the closely approximated statistical model [67][19][14][84]. $p(\mathbf{x})$ might be varied depending on the type of sample images. We will discuss more how to choose and measure the approximated model for $p(\mathbf{x})$ later.

We consider the second term $\ln p(\mathbf{x})$ in equation (34) as an unknown function $g(\mathbf{x})$. Then, from equations (34) and (35),

$$\begin{aligned}
\hat{\mathbf{x}} &= \arg\max_{\mathbf{x}\in\mathbb{R}^d} F(\mathbf{x}) \\
&= \arg\max_{\mathbf{x}\in\mathbb{R}^d} \left[-\frac{d}{2}\ln(2\pi\sigma^2) - \frac{(\mathbf{y}-\mathbf{x})^t(\mathbf{y}-\mathbf{x})}{2\sigma^2} + g(\mathbf{x})\right], \qquad (36)
\end{aligned}$$

where $F(\mathbf{x})$ represents the term inside arg max.

In order to maximize $F(\mathbf{x})$, we suppose that $F(\mathbf{x})$ is continuous, differentiable in $\mathbb{R}^d$ and convex (we have to choose $g(\mathbf{x})$ that can satisfy the conditions of $F(\mathbf{x})$). If there exists $\hat{\mathbf{x}}$ that satisfies $F(\hat{\mathbf{x}}) > \lim_{x_i\to\pm\infty} F(\mathbf{x})$, equation (36) is equivalent to the solution of the follwing equation:

$$\nabla F(\hat{\mathbf{x}}) = \frac{\partial F(\mathbf{x})}{\partial \mathbf{x}} = 0 \qquad (37)$$

when there exists one solution.

Finally, we can simplify equation (36) by using equation (37) as follows:

$$\nabla F(\hat{\mathbf{x}}) = -\frac{\hat{\mathbf{x}} - \mathbf{y}}{\sigma^2} + \nabla g(\hat{\mathbf{x}}) = 0$$

$$\Leftrightarrow \hat{\mathbf{x}} = \mathbf{y} + \sigma^2 \nabla g(\hat{\mathbf{x}}). \tag{38}$$

## 5.3 Empirical Multivariate Models of Wavelet Coefficients for Natural Images

In [67], Mallat proposed that the digital image histogram in the wavelet domain can be modeled with GGD and Chang et al. [14] used it with specific parameters to derive their Bayesian threshold. Another close statistical model for the original wavelet coefficients is Gaussian mixture model which has been applied in [19][29][78]. Also Sendur applied a circularly symmetric pdf for their bivariate threshold [85].

The existing models for wavelet denosing, however, are usually based on univariate statistical model whereas $p(\mathbf{x})$ is a multivariate pdf in our model. There are several multivariate functions which are symmetric spherically like multivariate Gaussian model. For example, multivariate Cauchy distribution, Gaussian distribution, Laplacian distribution or Gaussian mixture can be considered. Especially Gaussian mixture model shows various possibility to construct the optimal model in the literature [81][19][29][78].

In our paper, we use extended GGD model for its simple form and to achieve good fitting errors. We call this model multivariate generalized Gaussian distribution (MGGD):

$$p(\mathbf{x}) = C \exp\left\{ -\left( \frac{(\mathbf{x}-\mu)^t \Sigma_{\mathbf{x}}^{-1} (\mathbf{x}-\mu)}{\alpha} \right)^{\beta} \right\}, \tag{39}$$

where $\alpha$ and $\beta$ are parameters which can represent the spherical shape of the model and $C$ indicates a normalized constant defined by $\alpha$, $\beta$ and the covariance matrix $\Sigma_{\mathbf{x}}$.

When the dimension of $\mathbf{x}$ is one (scalar), the MGGD is still applicable and is denoted by UGGD (univariate generalized Gaussian distribution). MGGD is a particular case of the

$v$-spherical distribution defined by Fernández [45].

Using MGGD model, we can derive more specific forms of equation (38).

Since we can assume that $\mu = \mathbf{0}$,

$$\nabla g(\mathbf{x}) = -\frac{2\beta}{\alpha^\beta}(\mathbf{x}^t\Sigma_{\mathbf{x}}^{-1}\mathbf{x})^{\beta-1}\Sigma_{\mathbf{x}}^{-1}\mathbf{x}. \tag{40}$$

From equations (38) and (40),

$$\begin{aligned}
\hat{\mathbf{x}} &= \mathbf{y} - \frac{2\sigma^2\beta}{\alpha^\beta}(\hat{\mathbf{x}}^t\Sigma_{\hat{\mathbf{x}}}^{-1}\hat{\mathbf{x}})^{\beta-1}\Sigma_{\hat{\mathbf{x}}}^{-1}\hat{\mathbf{x}} \\
\hat{\mathbf{x}} &= \left(I + \frac{2\sigma^2\beta}{\alpha^\beta}(\hat{\mathbf{x}}^t\Sigma_{\hat{\mathbf{x}}}^{-1}\hat{\mathbf{x}})^{\beta-1}\Sigma_{\hat{\mathbf{x}}}^{-1}\right)^{-1}\mathbf{y} \\
&= \left(\Sigma_{\hat{\mathbf{x}}} + \frac{2\sigma^2\beta}{\alpha^\beta}(\hat{\mathbf{x}}^t\Sigma_{\hat{\mathbf{x}}}^{-1}\hat{\mathbf{x}})^{\beta-1}I\right)^{-1}\Sigma_{\hat{\mathbf{x}}}\mathbf{y}.
\end{aligned} \tag{41}$$

To simplify equation (41) in terms of scalar variable, we define

$$q(\hat{\mathbf{x}}) = \hat{\mathbf{x}}^t\Sigma_{\hat{\mathbf{x}}}^{-1}\hat{\mathbf{x}}. \tag{42}$$

If we substitute (42) into (41), we can get the following expression regarding $q(\hat{\mathbf{x}})$,

$$q(\hat{\mathbf{x}}) = \mathbf{y}^t\left(\Sigma_{\hat{\mathbf{x}}} + \frac{2\sigma^2\beta\{q(\hat{\mathbf{x}})\}^{\beta-1}}{\alpha^\beta}I\right)^{-2}\Sigma_{\hat{\mathbf{x}}}\mathbf{y}. \tag{43}$$

Hence, if we solve equation (43) and get the value of $q(\hat{\mathbf{x}})$, we can get the estimated vector $\hat{\mathbf{x}}$ using the equation (41).

However, there is no general solution for equation (43). To overcome this problem, we can define a particular condition for $\alpha$, $\beta$ and $\Sigma_{\hat{\mathbf{x}}}$ or use a numerical method. We simply use classical Newton's method in our experiments [39].

## 5.4 Model Selection

In the previous two sections, we have derived the general solution for the estimated original coefficients based on MGGD model. But, we do not know which parameters are optimal yet. In this section, we analyze the distribution of the wavelet coefficients for natural images and get the optimal values for the parameters $\alpha$ and $\beta$.

### 5.4.1 Distribution in wavelet domain and approximation

It is well-known that the distribution of detailed wavelet coefficients from natural images looks Gaussian-like with zero mean such as GGD [67]. We have tried to find the closest MGGD model for each subband analytically. 20 test images[1] with 512×512 size have been inspected to extract enough sample coefficients. In this analysis, Daubechies 8 filter is applied to the image set for the wavelet decomposition. The distribution parameters could be slightly different depending on the mother wavelet.

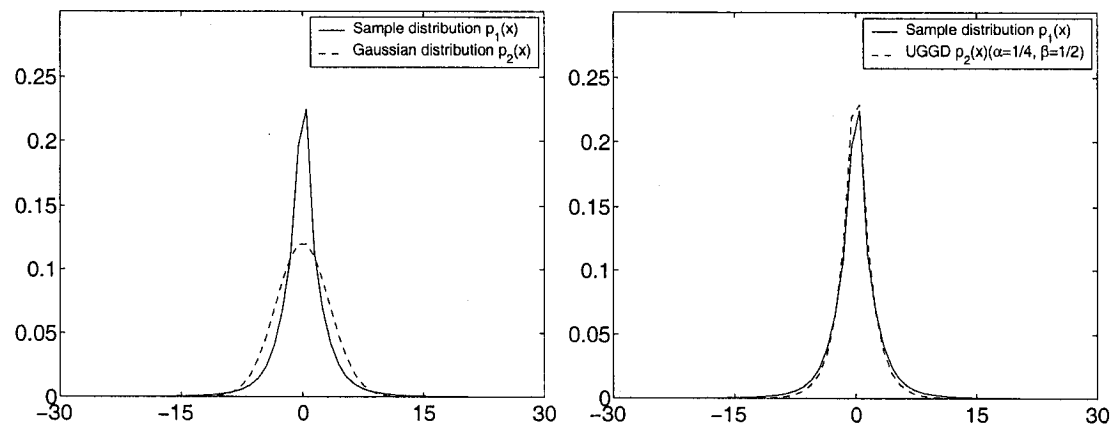



Figure 16: Distribution of sample coefficients and the estimated UGGD function (using sample coefficients in HH subband of the 1st decomposition stage).

Figure 16 shows the difference between the sample distribution and its estimated pdfs. As mentioned before, Gaussian distribution in the left figure does not fit the sample distribution closely but UGGD model with particular parameters is better adapted.

[1]Free images collected and offered by Computer Vision Group in University of Granada, Spain: `http://decsai.ugr.es/cvg`

Our issue that cacluates optimal parameters can be restated in terms of data-fitting problem. We apply *nonlinear least squares* fitting technique for our analysis.

If we consider mean squares between two distribution functions, the squared $L^2$-norm of the residual can be defined as follows:

$$\begin{aligned} R &= \left\| p_2(\mathbf{x}|\alpha,\beta) - p_1(\mathbf{x}) \right\|_{L^2}^2 \\ &= \sum_i \left( p_2(\mathbf{x}_i|\alpha,\beta) - p_1(\mathbf{x}_i) \right)^2 \end{aligned} \tag{44}$$

In this case, the closest $p_2(\mathbf{x})$ to $p_1(\mathbf{x})$ and its parameters $\alpha$, $\beta$ can be obtained by minimizing $R$. In our analysis, this process is carried out by using `lsqcurvefit()` function in Matlab optimization toolbox. Table 2 includes a list of the optimal parameters for each subband in UGGD model. The optimal parmeters vary by decomposition level and subband.

| **Subband** | | **Closest parameters** | | $R$ | **Sample Statistics** | |
|---|---|---|---|---|---|---|
| **Decomposition Level** | Detail | $\alpha$ | $\beta$ | ($\times 10^{-4}$) | mean ($\mu_x$) | Stddev ($\sigma_x$) |
| 1 | LH | 0.0368 | 0.3359 | 1.1661 | -0.0115 | 7.9642 |
| | HL | 0.0276 | 0.3152 | 0.7654 | -0.0333 | 7.7792 |
| | HH | 0.1834 | 0.4131 | 2.1585 | 0.0206 | 3.2922 |
| 2 | LH | 0.0098 | 0.2788 | 1.3089 | -0.0953 | 21.9288 |
| | HL | 0.0078 | 0.2632 | 1.4366 | -0.1183 | 21.3319 |
| | HH | 0.0675 | 0.3558 | 0.7916 | -0.0042 | 9.5786 |
| 3 | LH | 0.0074 | 0.2650 | 1.5015 | 0.1610 | 54.9322 |
| | HL | 0.0055 | 0.2441 | 0.9591 | -0.0497 | 51.1904 |
| | HH | 0.0160 | 0.2821 | 0.9977 | 0.0089 | 26.2107 |
| 4 | LH | 0.0057 | 0.2382 | 0.7746 | 0.3035 | 126.9218 |
| | HL | 0.0118 | 0.2533 | 0.5633 | 0.5073 | 121.9800 |
| | HH | 0.0154 | 0.2713 | 0.7513 | 0.0924 | 65.2276 |

Table 2: Appropriate parameters for MGGD model and $L^2$ norm of its resdual decided by nonlinear curve-fitting of least-squares algorithm.

Practically, it is hard to find the optimal parameters for all the cases. Each image has a different values although they are close to each other. Since we assume that the original image is not offered, the parameters cannot be measured each time.

### 5.4.2 pdf for multivariate distributions

It is not always easy to get multivariate distribution of wavelet coefficients from sample images since the $d$-dimensional vector requires the $d$-th power of samples for the same resolution. If sample points are sparsely distributed in multi-dimensional space, the estimated fitting function may not be accurate. Also, it is not easy to manage covariance matrix in general case. Hence, instead of analyzing the multivariate model in our experiments which uses ten elements, a bivariate model is analyzed as a specific example of multivariate model.

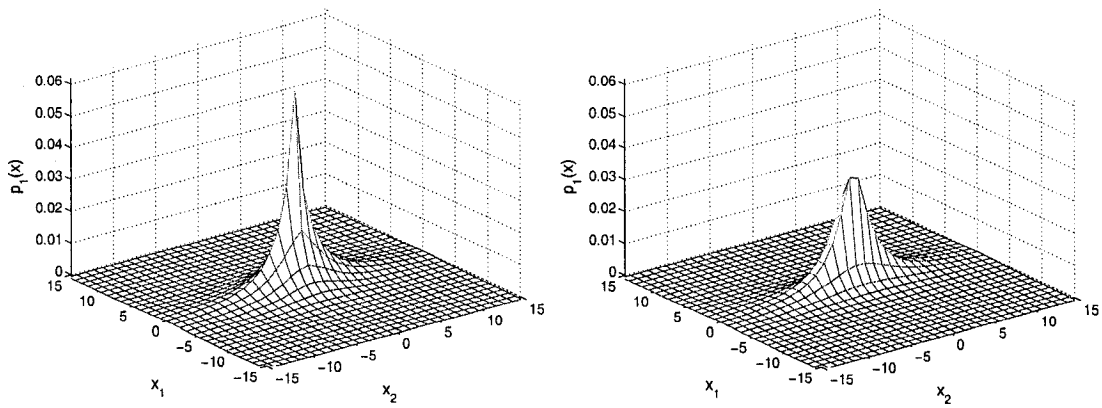



Figure 17: Sample distribution of bivariate model (left) and its fitting MGGD model (right) when $\mathbf{x} = (x_1, x_2)^t$, where $x_1$ is the coefficient from HH1 and $x_2$ is its parent from HH2.

In Figure 17, we can see one example of bivariate distribution for the wavelet coefficients and its MGGD model fitted by nonlinear least-squares optimization. The cutway view of left figure by the plane which is perpendicular to $x_1x_2$ plane and passes through the origin is close to the shape of univariate model. As a matter of fact, the multivariate model of wavelet coefficient samples is spherically symmetric.

The optimal parameters for bivariate case are in a similar range, where mostly $\alpha \in (0, 1]$ and $\beta \in [0.2, 0.5]$, but they are slightly different case by case. The optimal parameters of MGGD model vary as the number of elements as well as the way to choose the elements. Since we do not have the statistical properties of the clean image, we choose particular parameters in a certain closed range to simplify our experiments. These parameter ranges are based on the experiments using the sample images we have tested.

### 5.4.3 Specific Examples

In this part, we inspect the specific cases of our multivariate approach. Since the statistical model of the original wavelet coefficients is the main factor to decide the estimated coefficient for denoising, the parameters of MGGD model strongly affect the denoising performance of our approach. As shown in the last parts, optimal parameters could be decided by certain data fitting rule. However, we can define the specific parameters for lower computation because the multivariate estimator does not have general solution as shown in equation (43) and this requires high computational complexity. In fact, there have been used several models having specific solution and they can be included in UGGD or MGGD model. We look through some of those special cases and obtain the solution of each case.

**UNIVARIATE GAUSSIAN MODEL** : MGGD model includes Gaussian distribution. More specificially MGGD is multivariate Gaussian when $\alpha = 2$ and $\beta = 1$. For univariate Gaussian distribution, the model is denoted by UGGD and we can estimate the wavelet coefficient from equation (41) as follows:

$$\hat{x} = \frac{\sigma_x^2}{\sigma_x^2 + \sigma^2} y. \tag{45}$$

Equation (45) is the same form as MMSE estimator used in [70].

**UGGD MODEL WITH $\beta = 1/2$** : UGGD model with $\beta = 1/2$ is related to Laplacian distribution. When $\beta = 1/2$, the solution for the estimated coefficient is the form of soft thresholding as follows:

$$\hat{x} = sgn(y) \left( |y| - \frac{\sigma^2}{\sqrt{\alpha}\sigma_x} \right)_+ . \tag{46}$$

We can notice that when $\alpha = 1$, the threshold in equation (46) is equivalent to $BayesShrink$ proposed by Chang et al. [14].

**MGGD WITH $\beta = 1/2$ AND COVARIANCE MATRIX $\Sigma_{\mathbf{x}} = \sigma_{x_1}^2 I$** : If the covariance matrix is assumed as a simple form especially when the correlation coefficient is close to zero, it may be easier to simplify our matrix formula and obtain the solution for the estimated coefficient. From equations (41) and (43) under the given conditions, the simplified

solution for MGGD model with $\beta = 1/2$ is obtained in a *shrinkage* form as follows:

$$\hat{\mathbf{x}} = \left(1 - \frac{\sigma^2}{\sqrt{\alpha}\sigma_{x_1}} \frac{1}{\|\mathbf{y}\|}\right)_+ \mathbf{y}. \tag{47}$$

Sendur and Selesnick proposed several bivariate models in [84]. Among them, *Model 1* is a specific case of MGGD model when $d = 2$, $\alpha = 1/3$, $\beta = 1/2$ and $\Sigma_{\mathbf{x}} = \sigma_{x_1}^2 I$. And the derived solution can be expressed as equation (47).

In brief, there are numerous possibilities to derive the proper threshold or shrinkage rule by varying the parameters of MGGD. These examples are only a few specific cases among all the possibilities. For different types of sources of data such as astronomial images, medical images or 1D signals, the statistical models which do not include noise could be different. We can apply appropriate parameters for each case.

# Chapter 6

# Experiments and Performance Evaluation

In this chapter, we perform the experiments applying our proposed methods to various images from different sources. We use several types of wavelet transforms. The ways to choose parameters for our methods and practical configurations for our proposed algorithms are discussed. Also, the experimental results are evaluated by comparing our approaches with other existing methods both numerically and visually.

## 6.1 Test images

The test images are 8-bit gray-level with $256 \times 256$ and $512 \times 512$ sizes. The types of images include natural, medical and astronomial images. Among them we do more experiments on three popular $512 \times 512$ images, which are *Lena*, *Boat* and *Barbara*, for comparison purposes. All the images we use in our experiments are obtained from several different sources [1, 34, 75, 10].

## 6.2 Wavelets for denoising

Before applying a shrinkage algorithm, selecting an effective wavelet transform for denoising should be considered since wavelet transforms do affect the quality of a denoised image as mentioned in Chapter 2. The quality of a denoised image, in fact, considerably depends on the wavelet type even for the same kind of shrinkage algorithm as seen from the tables in the last part of this chapter.

There exist many orthogonal wavelet families such as Daubechies, Coiflet and Symmlet. We have chosen Daubechies wavelet D8 which is one of the most popular mother wavelets for denoising and DT CWT used in [23] and [85]. In addition, TI CWT have been used by combining both TI and DT CWT as described in Chapter 2. We have implemented TI CWT based on the `WaveLab` software package [34] used by [27] and Matlab codes used in [85] which is offered by Selesnick [10]. Since DT CWT has the near shift-invariant property, TI CWT gives slightly better results than DT CWT. Wavelet decomposition levels also affect the denoising performance. We have empirically set a satisfactory level $L = 6$ for 512×512 images.

## 6.3 Choosing parameters

### 6.3.1 Optimal threshold for *NeighShrink* and *NeighLevel*

For our proposed shrinkage algorithms *NeighShrink* and *NeighLevel* in Chapter 4, determining an optimal constant $\alpha$ in equations (29) and (31) is required. $\alpha$ can be chosen experimentally as shown in Figure 14. We found that the value is empirically similar to the optimal value for universal soft-thresholding in [33] and located in a particularly narrow range even for diverse types of images with different size and noise level.

We set $\alpha$ to $0.16 \sim 0.19$ for both $NeighShrink$ and $NeighLevel$ depending on the sizes of the neighbouring window and parent when we use DT CWT. From our experiments, we found that $\alpha$ is slightly bigger if the neighbouring window becomes smaller.

In DT CWT, a threshold should be doubled since it is compared with the sum of the

squares of the magnitudes calculated from both real and imaginary parts.

### 6.3.2 Parameters for multivariate model

For the proposed approach in Chapter 5, we need to choose the elements of vector $\mathbf{x}$ which includes the estimated (thresholded) coefficient itself and other related coefficients. In our experiments, we choose 10 elements as depicted in Figure 18. The elements should be closely connected with the estimated coefficient, for example the neighbouring coefficients.

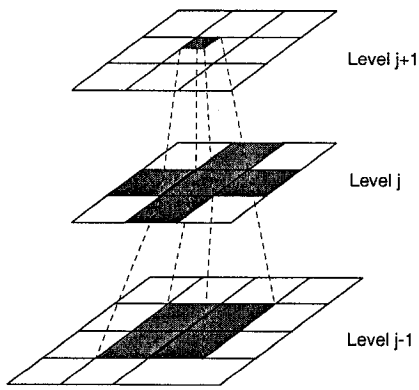



Figure 18: Selected elements of vector $\mathbf{x}$ from wavelet coefficients in our experiments ($d = 10$).

As described in the previous chapter, the proper parameters for the statistical model are necessary for better estimation. This could be difficult since they are decided case by case empirically depending the type of images, the subband in the wavelet domain and the chosen elements of vector $\mathbf{x}$. For this combination of elements, we select the parameters of MGGD model as $\alpha = 1/6$ and $\beta = 1/2$ for simplicity in our experiments.

### 6.3.3 Variances and covarinaces

As mentioned in the introduction, the noise model in this thesis is zero-mean AWGN $N(0, \sigma^2)$. So the variance of the noise model is the only parameter we have. The noise

variance $\sigma^2$ can be estimated by a robust median estimator from the noisy wavelet coefficients in $HH_1$ subband as follows [35]:

$$\hat{\sigma}^2 = \frac{\text{median}(|y_k|)}{0.6745}, \tag{48}$$

where $y_k$ is an element in $HH_1$ and so the median$(|y_k|)$ is a median from all the absolute values of the coefficients in $HH_1$.

The variances in clean image and noisy image are important statistical information to get our denoised wavelet coefficients as shown in the previous chapters. Especially our multivariate approach requires the estimated covariance $\hat{\Sigma}_{\mathbf{x}}$ of multivariate model as well as the noise variance $\sigma^2$. Therefore, we have to choose $\hat{\Sigma}_{\mathbf{x}}$ for a model. In our case, we use the following estimation for $\hat{\Sigma}_{\mathbf{x}}$ since the noise is independently distributed:

$$\widehat{\Sigma}_{\mathbf{x}} = \Sigma_{\mathbf{y}} - \sigma^2 I. \tag{49}$$

There are two ways to build the covariance matrix $\Sigma_{\mathbf{y}}$. One way is that $\Sigma_{\mathbf{y}}$ is composed by the subbands that each element of vector $\mathbf{y}$ belongs to. The other way is to use a $M \times M$ *local window* which surrounds each element of $\mathbf{y}$ instead of each subband. This is on the same line as the local variance which is empirically used in recent works [70][85]. In our experiments, the local covariance produces higher quality of image when a $7 \times 7$ window is applied for each element. Since the estimated correlation between different sources could not be accurate due to the number of samples and correlation between elements are not so big, we ignore them by using a diagonal matrix for the local covariance for the experiments.

## 6.4 Evaluations

To evaluate and analyze our proposed algorithms, we compared them with the existing effective approaches introduced in Chapter 3. Denoised images can be compared both visually and numerically. Representative numerical measures for image quality are MSE and PSNR (see the definitions in Appendix). We use 255 as the maximum value of $A$ since

the 8-bit gray level value ranges $[0, 255]$.

For different noise variances, the measured PSNR values are listed in Tables 3, 4 and 5 for our proposed and other methods. The results are categorized in terms of the type of the wavelet used since denoising results are dependent on the wavelet transforms. We have made use of some functions in `WaveLab` for $VisuShrink$ and $SureShrink$. For other existing methods, we use the experimental results from the original papers and the PSNR table[1] in [85]. GSM results for Daubechies 8 filters are obtained from the software offered by Portilla [78]. A comparison of selected methods is given in Figure 21 and 22 for a $512 \times 512$ size *Lena* and *Barbara* images.

*Wiener* filter[2] [59] is also included since it considers the neighbouring dependency and achieves efficacious performance with simple linear filter even if wavelet transforms are not applied.

As shown in the tables and graphs, both of our proposed approaches present better or comparable results compared to other existing methods.

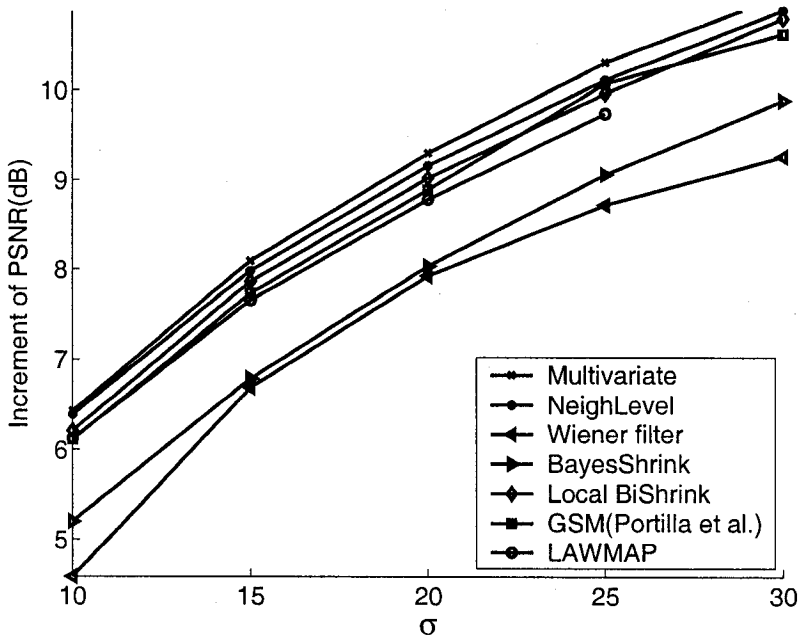
In [78], an image denoising algorithm using Gaussian scale mixtures is proposed. In their experiments, the results are only 0.1dB $\sim$ 0.2dB better than *NeighLevel* as shown in the tables. However, it may not be proper to compare our results with [78] because Portilla et al. use a customized wavelet called *steerable pyramid* [91] while we use the usual orthogonal wavelets and DT CWT. The emphasis of our paper is in the study of the effects of neighbouring and level dependencies on thresholding the wavelet coefficients. Moreover, our Matlab program takes 7 seconds for a 512×512 image with DT CWT on 1GHz Pentium III, whereas their Matlab implementation takes roughly 40 seconds for a 256×256 image on 1.7 GHz Pentium III according to [78]. Also our multivariate modeling approach outperforms all other methods reported in the literature when the scalar wavelet transform is used, in particular Daubechies 8 filter.

[1]Since they assume that the *maximum value* for PSNR is 256, we subtract 0.03 dB, which is the approximate value of $20 \log_{10} \frac{256}{255}$.

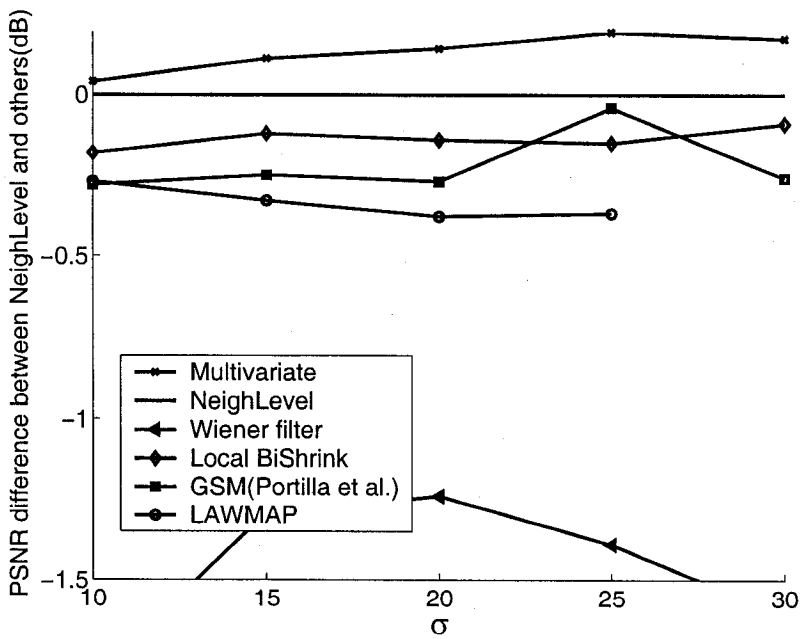
[2]We used `wiener2` function in MATLAB image processing toolbox with a 5×5 neighbouring window and unknown noise level.

Since there is no general solution for equation (43), iterative numerical solution is applied for the multivariate approach. It takes about 30 seconds for a $512 \times 512$ image with Daubechies 8 filter on 2.4 GHz Pentium IV PC when 10 elements are used. However, when $\hat{\mathbf{x}}$ can be calculated explicitly without Newton's method in equation (41) like the examples in Section 5.4.3, it only takes less than 3 seconds under the same condition.

In Appendix, more examples for image denoising are given for different types of images such as astronomical image, satellite image, character image and biomedical image. We believe the denoised images are helpful to recognize certain features in the images.

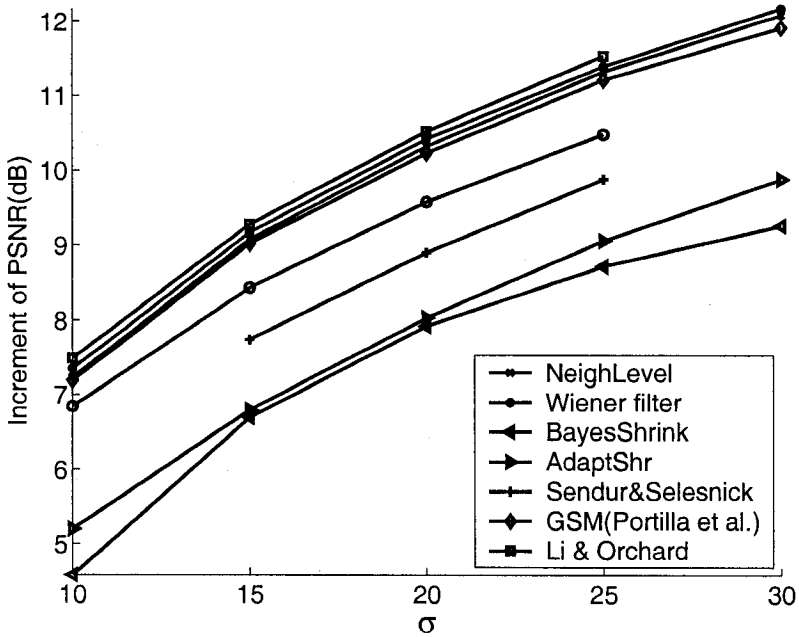



(a) incremental PSNR vs. $\sigma$

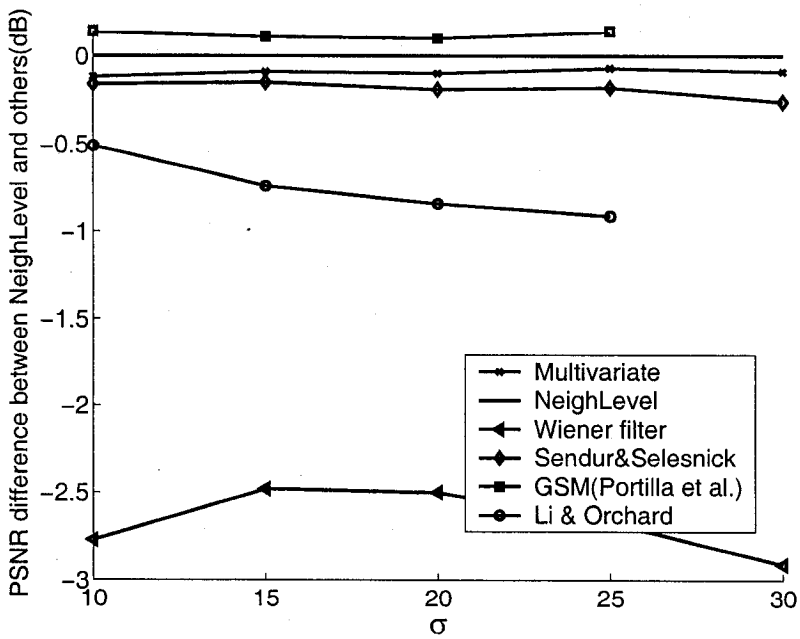



(b) PSNR differences between *NeighLevel* (zero horizontal line) and other approaches

Figure 19: Comparison graphs for some principal approaches from 512×512 *Lena* image using Daubechies 8 filter.




(a) incremental PSNR vs. $\sigma$




(b) PSNR differences between *NeighLevel* (zero horizontal line) and other approaches

Figure 20: Comparison graphs for some principal approaches from 512×512 *Lena* image regardless of the wavelet transforms.

Figure 21: Denoised images using proposed algorithms for 512×512 *Lena* image with $\sigma$=30 : Original (top-left), Noisy (top-center; 18.60dB), $VisuShrink$ soft (top-right; 25.60dB), $VisuShrink$ hard (middle-left; 26.33dB), $Wiener$ filter (middle-center; 27.83dB), $NeighSure$ (middle-right; 30.23dB), $NeighShrink$ (bottom-left; 30.46dB), $NeighLevel$ (bottom-center; 30.76dB) Multivariate (bottom-right; 30.68dB).

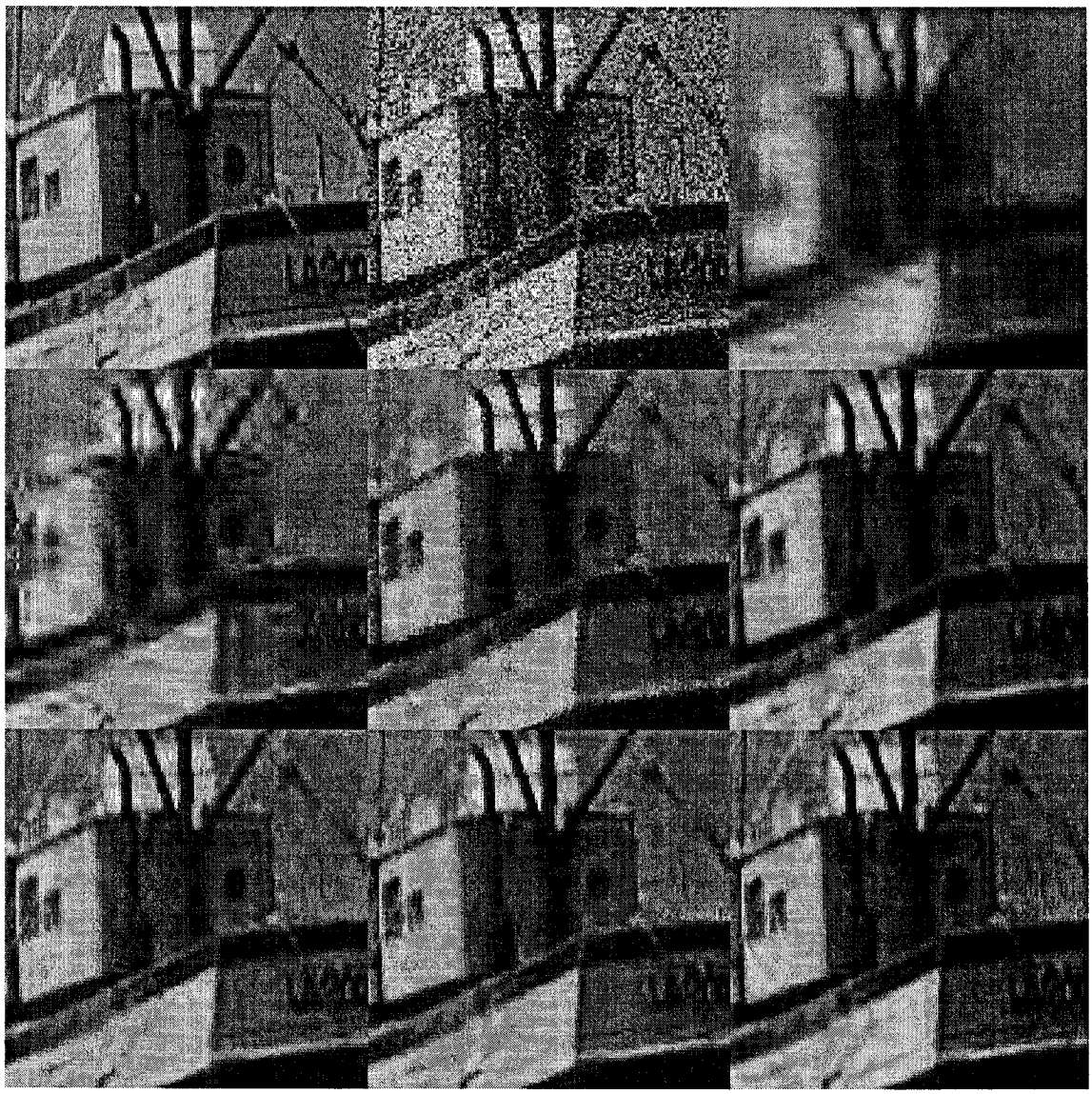


Figure 22: Cropped images (128×128) using proposed algorithms for 512×512 *Boat* image with $\sigma$=25 : Original (top-left), Noisy (top-center; 18.60dB), $VisuShrink$ soft (top-right; 24.06dB), $VisuShrink$ hard (middle-left; 25.03dB), $Wiener$ filter (middle-center; 27.22dB), $NeighSure$ (middle-right; 28.48dB), $NeighShrink$ (bottom-left; 28.90dB), $NeighLevel$ (bottom-center; 29.11dB), Multivariate (bottom-right; 29.12dB).

| Approach | Wavelet | | PSNR(dB) by noise level($\sigma$) | | | | |
|---|---|---|---|---|---|---|---|
| | | | $\sigma = 10$ | $\sigma = 15$ | $\sigma = 20$ | $\sigma = 25$ | $\sigma = 30$ |
| Noisy image | – | | 28.12 | 24.62 | 22.14 | 20.16 | 18.60 |
| `Wiener2` (5×5) | – | | 32.67 | 31.28 | 30.03 | 28.85 | 27.83 |
| Multivariate | Scalar | Daub. 8 | 34.55 | 32.71 | 31.44 | 30.46 | 29.64 |
| *NeighSure* | wavelets | | 33.89 | 32.00 | 30.76 | 29.75 | 28.98 |
| *NeighShrink* | | | 34.49 | 32.57 | 31.25 | 30.20 | 29.38 |
| *NeighLevel* | | | 34.51 | 32.60 | 31.30 | 30.27 | 29.47 |
| VisuShrink (soft) [33] | | | 28.70 | 27.40 | 26.59 | 26.04 | 25.60 |
| VisuShrink (hard) [33] | | | 30.65 | 28.89 | 27.76 | 27.02 | 26.33 |
| *SureShrink* [36] | | | 33.42 | 31.50 | 30.17 | 29.18 | 28.47 |
| BiShrink. [84] | | | 33.91 | 32.03 | 30.70 | 29.78 | 28.91 |
| Local BiShrink. [85] | | | 34.33 | 32.48 | 31.16 | 30.12 | 29.38 |
| HMT [29] | | | 33.81 | 31.73 | 30.36 | 29.21 | 28.32 |
| LAWMAP [70] | | | 34.24 | 32.27 | 30.92 | 29.90 | |
| GSM [78] | | | 34.23 | 32.35 | 31.03 | 30.23 | 29.21 |
| BayesShrink [14] | | | 33.29 | 31.38 | 30.14 | 29.19 | 28.45 |
| *NeighSure* | | Symm.8 | 34.08 | 32.27 | 31.00 | 30.00 | 29.21 |
| *NeighShrink* | | | 34.65 | 32.77 | 31.48 | 30.41 | 29.62 |
| *NeighLevel* | | | 34.61 | 32.75 | 31.47 | 30.43 | 29.63 |
| AdaptShr [15] | | | | 32.36 | 31.04 | 30.04 | |
| Multivariate | Complex | DT CWT | 35.35 | 33.70 | 32.46 | 31.48 | 30.68 |
| *NeighSure* | wavelets | | 35.17 | 33.31 | 31.92 | 31.02 | 30.17 |
| *NeighShrink* | | | 35.37 | 33.57 | 32.26 | 31.22 | 30.41 |
| *NeighLevel* | | | 35.41 | 33.72 | 32.50 | 31.48 | 30.70 |
| Local BiShrink. [85] | | | 35.31 | 33.64 | 32.37 | 31.37 | 30.51 |
| CHMT [23] | | | 34.90 | | | 29.90 | |
| *NeighShrink* | Over- | TI CWT | 35.41 | 33.62 | 32.31 | 31.27 | 30.47 |
| *NeighSure* | complete | | 35.22 | 33.33 | 32.01 | 31.05 | 30.23 |
| *NeighLevel* | wavelets | | 35.47 | 33.78 | 32.56 | 31.55 | 30.76 |
| MMSE [60] | | OEB[a] 10/18 | 34.93 | 33.01 | 31.69 | 30.60 | |
| SI-AdaptShr [15] | | SI Symm. 8 | | 33.37 | 32.09 | 31.07 | |
| GSM [78] | | Steer. pyramid | 35.61 | 33.90 | 32.66 | 31.69 | |

[a]OEB stands for *overcomplete expansion biorthogonal* filter

Table 3: Comparison table for proposed and existing methods with different Gaussian noise (*Lena* 512 × 512).

| Approach | Wavelet | | PSNR(dB) by noise level($\sigma$) | | | | |
|---|---|---|---|---|---|---|---|
| | | | $\sigma = 10$ | $\sigma = 15$ | $\sigma = 20$ | $\sigma = 25$ | $\sigma = 30$ |
| Noisy image | – | | 28.12 | 24.62 | 22.14 | 20.16 | 18.60 |
| `Wiener2` (5×5) | – | | 28.05 | 27.16 | 26.28 | 25.47 | 24.73 |
| *NeighSure* | Scalar wavelets | Daub. 8 | 31.94 | 29.54 | 28.03 | 27.05 | 26.16 |
| *NeighShrink* | | | 32.68 | 30.36 | 28.74 | 27.54 | 26.61 |
| *NeighLevel* | | | 32.67 | 30.36 | 28.75 | 27.56 | 26.63 |
| VisuShrink (soft) | | | 25.07 | 23.68 | 22.97 | 22.59 | 22.36 |
| VisuShrink (hard) | | | 27.46 | 25.25 | 23.93 | 23.16 | 22.74 |
| SureShrink [36] | | | 31.04 | 28.70 | 27.14 | 26.03 | 25.13 |
| BiShrink. [84] | | | 31.10 | 28.68 | 27.22 | 25.94 | 25.18 |
| Local BiShrink. [85] | | | 32.22 | 29.94 | 28.33 | 27.13 | 26.25 |
| BayesShrink [14] | | | 30.83 | 28.48 | 27.10 | 25.98 | 25.13 |
| HMT [29] | | | 31.33 | 29.57 | 27.91 | 26.72 | 25.77 |
| LAWMAP [70] | | | 32.51 | 30.13 | 28.57 | 27.40 | |
| *NeighSure* | | Symm. 8 | 32.19 | 29.85 | 28.40 | 27.45 | 26.45 |
| *NeighShrink* | | | 32.98 | 30.67 | 29.04 | 27.45 | 26.86 |
| *NeighLevel* | | | 32.88 | 30.61 | 29.01 | 27.82 | 26.87 |
| AdaptShr [15] | | | | 29.92 | 28.33 | 27.20 | |
| *NeighSure* | Complex wavelets | DT CWT | 33.36 | 31.22 | 29.61 | 28.48 | 27.22 |
| *NeighShrink* | | | 33.79 | 31.60 | 30.02 | 28.84 | 27.88 |
| *NeighLevel* | | | 33.78 | 31.64 | 30.08 | 28.89 | 27.93 |
| Local BiShrink. [85] | | | 33.32 | 31.28 | 29.77 | 28.58 | 27.62 |
| *NeighSure* | Over-complete wavelets | TI CWT | 33.40 | 31.28 | 29.69 | 28.58 | 27.38 |
| *NeighShrink* | | | 33.87 | 31.68 | 30.10 | 28.92 | 27.95 |
| *NeighLevel* | | | 33.87 | 31.72 | 30.15 | 28.97 | 27.99 |
| SI-AdaptShr [15] | | SI Symm.8 | | 31.11 | 29.49 | 28.30 | |
| MMSE [60] | | OEB 10/18 | 33.32 | 31.06 | 29.41 | 28.20 | |
| GSM [78] | | Steer. pyramid | 34.03 | 31.86 | 30.32 | 29.13 | |

Table 4: Comparison table for the proposed and existing methods with different Gaussian noise (*Barbara* 512 × 512).

| Approach | Wavelet | | PSNR(dB) by noise level($\sigma$) | | | | |
|---|---|---|---|---|---|---|---|
| | | | $\sigma = 10$ | $\sigma = 15$ | $\sigma = 20$ | $\sigma = 25$ | $\sigma = 30$ |
| Noisy image | – | | 28.13 | 24.63 | 22.10 | 20.18 | 18.59 |
| `Wiener2` (5×5) | – | | 30.02 | 29.04 | 28.09 | 27.22 | 26.40 |
| Multivariate | Scalar | Daub. 8 | 32.54 | 30.69 | 29.38 | 28.34 | 27.56 |
| *NeighSure* | wavelets | | 32.17 | 30.25 | 28.85 | 27.80 | 26.87 |
| *NeighShrink* | | | 32.67 | 30.63 | 29.23 | 28.17 | 27.31 |
| *NeighLevel* | | | 32.67 | 30.65 | 29.25 | 28.21 | 27.37 |
| VisuShrink (soft) [33] | | | 26.64 | 25.34 | 24.59 | 24.06 | 23.68 |
| VisuShrink (hard) [33] | | | 28.61 | 26.90 | 25.82 | 25.03 | 24.46 |
| SureShrink [36] | | | 31.83 | 29.88 | 28.55 | 27.50 | 26.73 |
| HMT [29] | | | 32.25 | 30.28 | 28.81 | 27.65 | 26.80 |
| LAWMAP [70] | | | 32.22 | 30.37 | 28.97 | 27.88 | 27.03 |
| BiShrink. [84] | | | 32.22 | 30.22 | 28.90 | 27.88 | 27.08 |
| Local BiShrink. [85] | | | 32.39 | 30.52 | 29.15 | 28.11 | 27.26 |
| GSM [78] | | | 32.39 | 30.41 | 29.03 | 27.99 | 27.15 |
| BayesShrink [15] | | | 31.77 | 29.84 | 28.45 | 27.37 | 26.57 |
| *NeighSure* | | Symm. 8 | 32.31 | 30.40 | 28.99 | 27.85 | 26.97 |
| *NeighShrink* | | | 32.78 | 30.75 | 29.35 | 28.26 | 27.40 |
| *NeighLevel* | | | 32.76 | 30.75 | 29.36 | 28.29 | 27.45 |
| Multivariate | Complex | DT CWT | 33.31 | 31.46 | 30.14 | 29.12 | 28.24 |
| *NeighSure* | wavelets | | 33.24 | 31.31 | 29.74 | 28.48 | 27.55 |
| *NeighShrink* | | | 33.30 | 31.31 | 29.94 | 28.90 | 28.04 |
| *NeighLevel* | | | 33.36 | 31.46 | 30.11 | 29.11 | 28.25 |
| Local BiShrink. [85] | | | 33.07 | 31.33 | 30.05 | 29.03 | 28.28 |
| *NeighSure* | Over- | TI CWT | 33.25 | 31.34 | 29.81 | 28.54 | 27.57 |
| *NeighShrink* | complete | | 33.32 | 31.33 | 29.97 | 28.93 | 28.08 |
| *NeighLevel* | wavelets | | 33.41 | 31.51 | 30.17 | 29.15 | 28.30 |
| GSM [78] | | Steer. pyramid | 33.58 | 31.70 | 30.38 | 29.37 | |

Table 5: Comparison table for proposed and the existing methods with different Gaussian noise (*Boat* 512 × 512).

# Chapter 7

# Conclusion and Discussion for Future Work

In this thesis, image denoising using wavelet transforms has been discussed. In the first part of the thesis, we have introduced some important wavelet transforms for image denoising. These include translation-invariant wavelets, complex wavelets, multiwavelets, directional wavelets, and other overcomplete wavelets. Then the existing denoising algorithms using various different approaches have been described as a literature review.

In the latter part of the thesis, we have proposed two different approaches for image denoising. The first approach is the specific shrinkage algorithms which take advantage of the higher order statistical coupling between neighbour wavelet coefficients and their corresponding coefficients in the parent level with effective translation-invariant wavelet transforms. Also the multiplying constant of a threshold which produces lower MSE for image denoising has been introduced and chosen empirically. The other image denoising method is based on the multivariate statistical model and estimates the clean wavelet coefficients using Bayesian probability solution. This method can produce more accurate estimation using various related information. The main advantage of the coefficient estimation method based on the multivariate theory is its generality. Various estimation expressions and different experimental results can be obtained by statistical modeling and parameterizing models. Both of our methods give fairly satisfying results in both visual and numerical

aspects. They outperform most of the existing algorithms as listed in the experimental results. In addition, the shrinkage algorithms in the first approach are considerably fast and effective in the aspects of both time and storage complexities. Even though the multivariate approach which does not have a general solution causes heavier computational complexity in general, it can be fast and effective in some specific cases. In this case, both of the proposed approaches are easy to implement.

Although our proposed approaches give the encouraging denoising results, we believe that there is more room of improvement to achieve high quality images. For example, it may be possible to develop other thresholding functions which are more coherent, better related to the neighbouring coefficients, and also representing better the hierarchical dependency between different wavelet decomposition levels. Furthermore more adaptive thresholds other than *SURE* threshold used in *NeighSure* algorithm may produce better results. For the multivariate statistical modeling approach, we can choose some different elements and parameters of our model.

Finally, it is also possible to combine our methods with the others to achieve high quality image restoration. In [92], Starck et al. reported that the combined method of curvelet and undecimated wavelet generates much higher quality of image by applying matching pursuit (MP) and basis pursuit (BP) algorithms. When they use noisy *Lena* image with $\sigma = 20$, the combined method yielded 32.72dB while undecimated wavelet and curvelet separately can only improve the noisy image to 32.10dB and 31.95dB respectively. In fact, we did a simple experiment which combines our proposed methods as a simple example. In our experiment, we combine the proposed multivariate method with *NeighLevel*. We simply take the average of two denoised images and achieve higher PSNR value. For example, when we use the same *Lena* image with $\sigma = 20$, *NeighLevel* and our proposed methods produce 32.54dB and 32.50dB respectively. By averaging these two images, we could obtain 32.61dB. This is a simple test, but we can see that combining different methods is probably very promising.

# Appendix A

# Mathematical Preliminaries

## A.1 Convolution

Let $\mathbf{f}(x)$ and $\mathbf{g}(x)$ be two functions of $x \in \mathbb{R}$. which map $\mathbf{f}, \mathbf{g} : \mathbb{R} \rightarrow \mathbb{R}$. Then a convolution of $\mathbf{f}(x)$ and $\mathbf{g}(x)$ can be defined as a production of two functions over a finite range $[0, x]$:

$$\mathbf{f} * \mathbf{g} = \int_0^t \mathbf{f}(x)\mathbf{g}(t-x)dx. \tag{50}$$

In an infinite range, the convolution is expressed by a function of time $t$:

$$\mathbf{f} * \mathbf{g} = \int_{-\infty}^{\infty} \mathbf{f}(x)\mathbf{g}(t-x)dx. \tag{51}$$

In descrete domain, $\mathbf{f}$ and $\mathbf{g}$ can be considered as vectors, which consist of $n_1$ and $n_2$ elements respectively. Then colvolution of $\mathbf{f}$ and $\mathbf{g}$ are defined as follows:

$$\mathbf{y}(t) = \sum_i \mathbf{f}(i)\mathbf{g}(t-i+1) \tag{52}$$

In this case $\mathbf{y}(t)$ is a $t$-th element of output vector $\mathbf{y} = \mathbf{f} * \mathbf{g}$, where $t = 1, \cdots, n_1 + n_2 - 1$.

Convolution satisfies a commutative property $\mathbf{f} * \mathbf{g} = \mathbf{g} * \mathbf{f}$, associative proerty $\mathbf{f} * (\mathbf{g} * \mathbf{h}) = (\mathbf{f} * \mathbf{g}) * \mathbf{h}$ and distributive property $\mathbf{f} * (\mathbf{g} + \mathbf{h}) = (\mathbf{f} * \mathbf{g}) + (\mathbf{f} * \mathbf{h})$.

## A.2 Definitions for numerical evaluation

Image denoising results can be measured numerically. We introduce some common definitions for the measurement of the image quality. Suppose that $A$ is a desired image with $N \times N$ size, $\hat{A}$ is an estimated image from a noisy image $B$. Then mean squared error (MSE) is expressed as follows:

$$\text{MSE} = \frac{||\hat{A} - A||^2}{N^2}. \tag{53}$$

Root mean squared error (RMSE) is the square root of MSE. MSE is noted for a good way to represent the error of a signal (or image). In fact, our fundamental goal for image denoising is to minimize MSE.

There are also the other ways to measure the error of the image, called signal-to-noise ratio (SNR) and peak signal-to-noise ration (PSNR). The unit for both of them is a decibel (dB). SNR and PSNR are defined in the following.

$$\text{SNR}(dB) = 10 \log_{10} \frac{\|A\|^2/N^2}{\text{MSE}} \tag{54}$$

$$\text{PSNR}(dB) = 10 \log_{10} \frac{(\text{maximum value})^2}{\text{MSE}}, \tag{55}$$

where the maximum value is the possible maximum value of $A$. For 8-bit gray level image, the maximum value is 255. Note that the bigger value, the better image quality for SNR and PSNR whereas the smaller value, the better image quality for MSE. Both SNR and PSNR are related to MSE, especially PSNR can be obtained only from MSE. PSNR is a popular way to measure the image quality in image processing field. For example, it is often used for compression performance. We mainly use PSNR in this thesis.

## A.3 Bayes rule and MAP/ML estimator

Bayes rule is the way to calculate a *posterior* probability from the prior probability. This can be expressed as follows:

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}, \tag{56}$$

where $X$, $Y$ are sets. In equation (56), when $X$ is a hypothesis and $Y$ is given as the observed data, $P(X)$ is the *prior* probability of $X$, the conditional probability $P(Y|X)$ is likelihood, and $P(Y)$ is the evidence [39].

Maximum *a posteriori* (MAP) estimator is one of the conventional ways of Bayesian solution and widely used for machine learining and pattern classification [71]. The main goal is to find the hypothesis which is maximally probable by using Bayes rule. Therefore such a hypothesis $x_{\mathrm{MAP}}$ can be expressed as follows:

$$\begin{aligned} x_{\mathrm{MAP}} &\equiv \arg\max_{x \in X} P(X = x|Y) \\ &= \arg\max_{x \in X} \frac{P(Y|X = x)P(X = x)}{P(Y)} \\ &= \arg\max_{x \in X} P(Y|X = x)P(X = x). \end{aligned} \tag{57}$$

In this case, the $P(Y)$ is dropped since it is an independent constant of $X$.

Another classical method by maximizing a given probability is maximum likelihood (ML). This is because we get $x_{\mathrm{ML}}$ by maximizing the likelihood probability $P(Y|X)$.

$$x_{\mathrm{ML}} \equiv \arg\max_{x \in X} P(Y|X = x). \tag{58}$$

As we can see, $x_{\mathrm{ML}}$ is one specific case of $x_{\mathrm{MAP}}$ when $P(X = x)$ is a constant.

## A.4 Probability density functions

Probability density function $p(x)$ (pdf) is the function to express the probability in terms of variable $x$. $p(x)$ is described as the derivative of the cumulative distribution function. So the probability function has the following property:

$$\int_{-\infty}^{\infty} p(x)dx = 1. \tag{59}$$

In this thesis, we use only continuous pdf rather than discrete pdf. We introduce some important continuous pdfs.

**(Univariate) Gaussian (or normal) distribution** is probably the most popular probability function since it has natural properties such as symmetry, continuity and smoothness. Gaussian distribution is simply written $N(\mu, \sigma^2)$, where $\mu$ is mean and $\sigma^2$ is variance. pdf of Gaussian distribution can be expressed as follows:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}. \tag{60}$$

**Generalized Gaussian distribution (GGD)** is generalized form of univariate Gaussian distribution with one more parameter. GGD is a good pdf to represent the wavelet coefficients [67]. If the mean $\mu = 0$ and $\sigma^2$ is variance,

$$p(x) = C \exp\left\{-\left|\frac{x}{\alpha}\right|^{\beta}\right\}, \tag{61}$$

where $\alpha = \sqrt{\frac{\Gamma(1/\beta)}{\Gamma(3/\beta)}}\sigma$ and the normalized constant $C = \frac{\beta}{2\alpha\Gamma(1/\beta)}$. Therefore, GGD has one parameter $\beta$ and $\beta > 0$.

**Multivariate Gaussian distribution** is an extended distribution of Gaussian distribution to the multidimensional space. So pdf $p(\mathbf{x})$ can be distributed in terms of a vector $\mathbf{x}$. Like univariate Gaussian distribution, multivariate Gaussian distribution is simply written $N(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{\mu}$ is mean vector and $\Sigma$ is covariance matrix. pdf of multivariate Gaussian

distribution for $d$-dimensional vector $\mathbf{x}$ can be expressed as follows:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}. \tag{62}$$

# Appendix B

# More Examples for Image Denoising

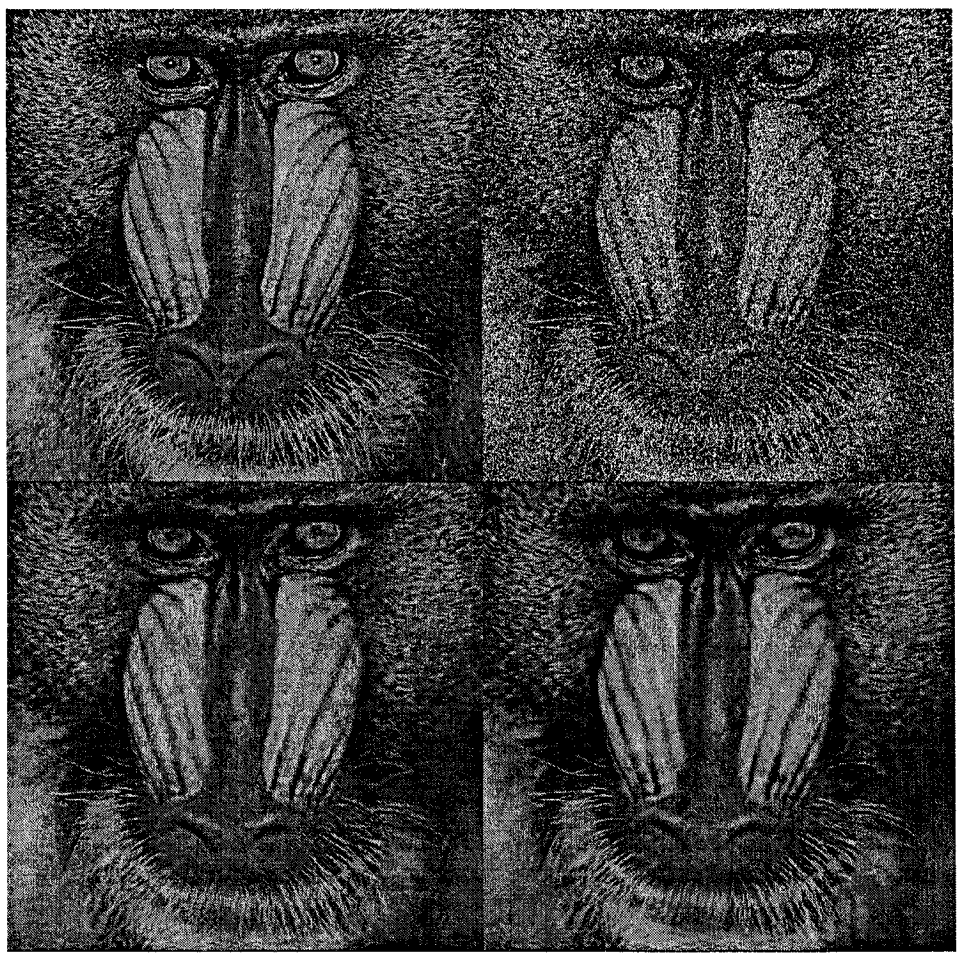


Figure 23: Denoised images using proposed algorithms for 512×512 *Baboon* image with $\sigma$=50 : Original (top-left), Noisy (top-right; 14.15dB), *NeighLevel* (bottom-left; 22.41dB), Multivariate (bottom-right; 22.21dB).

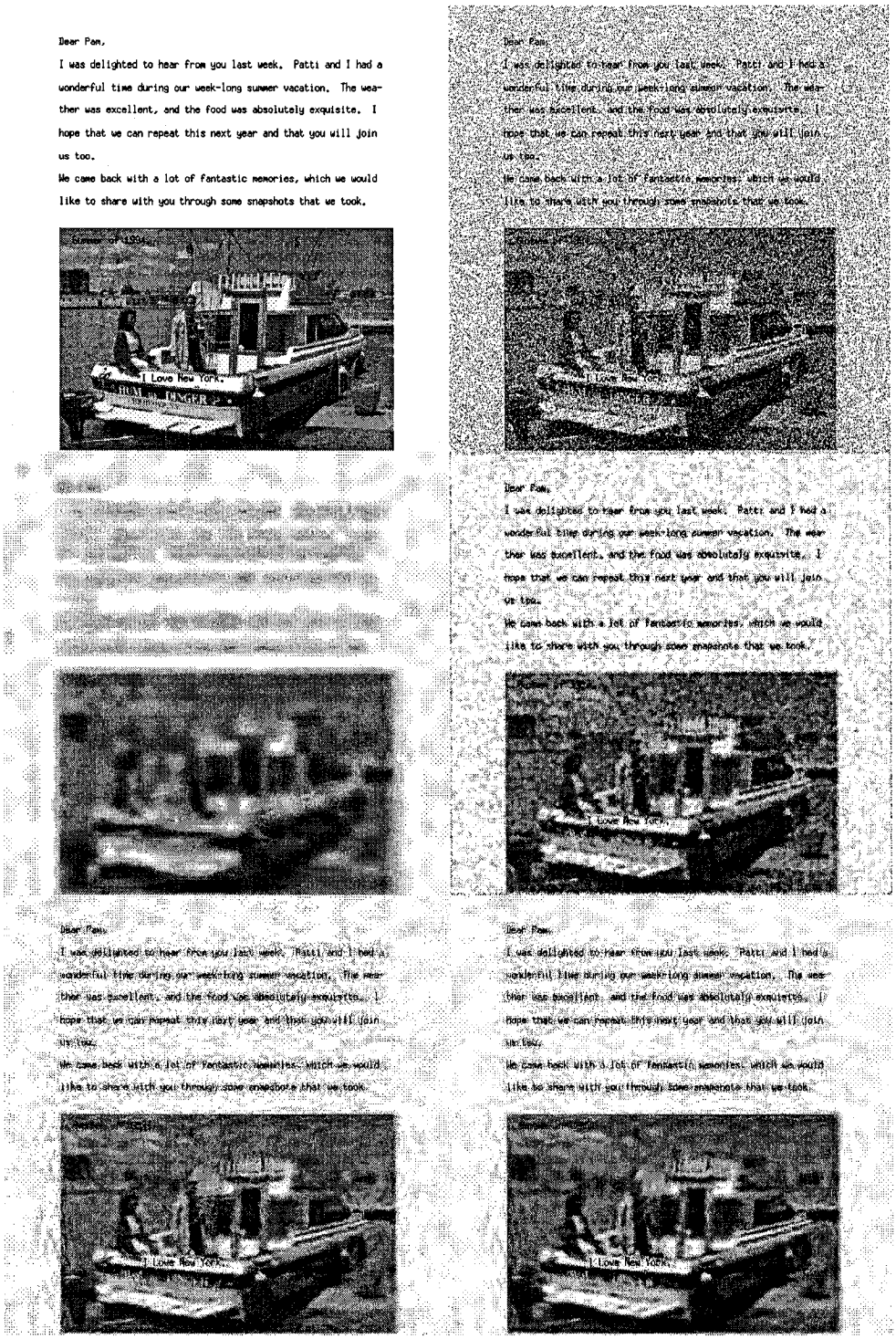


Figure 24: Denoised images using proposed algorithms for 512×512 image including characters with $\sigma$=60 : Original (top-left), Noisy (top-center; 12.57dB), *VisuShrink* soft (middle-left; 16.98dB), *Wiener* filter (middle-right; 20.99dB), *NeighLevel* (bottom-left; 21.66dB), Multivariate (bottom-right; 21.68dB).

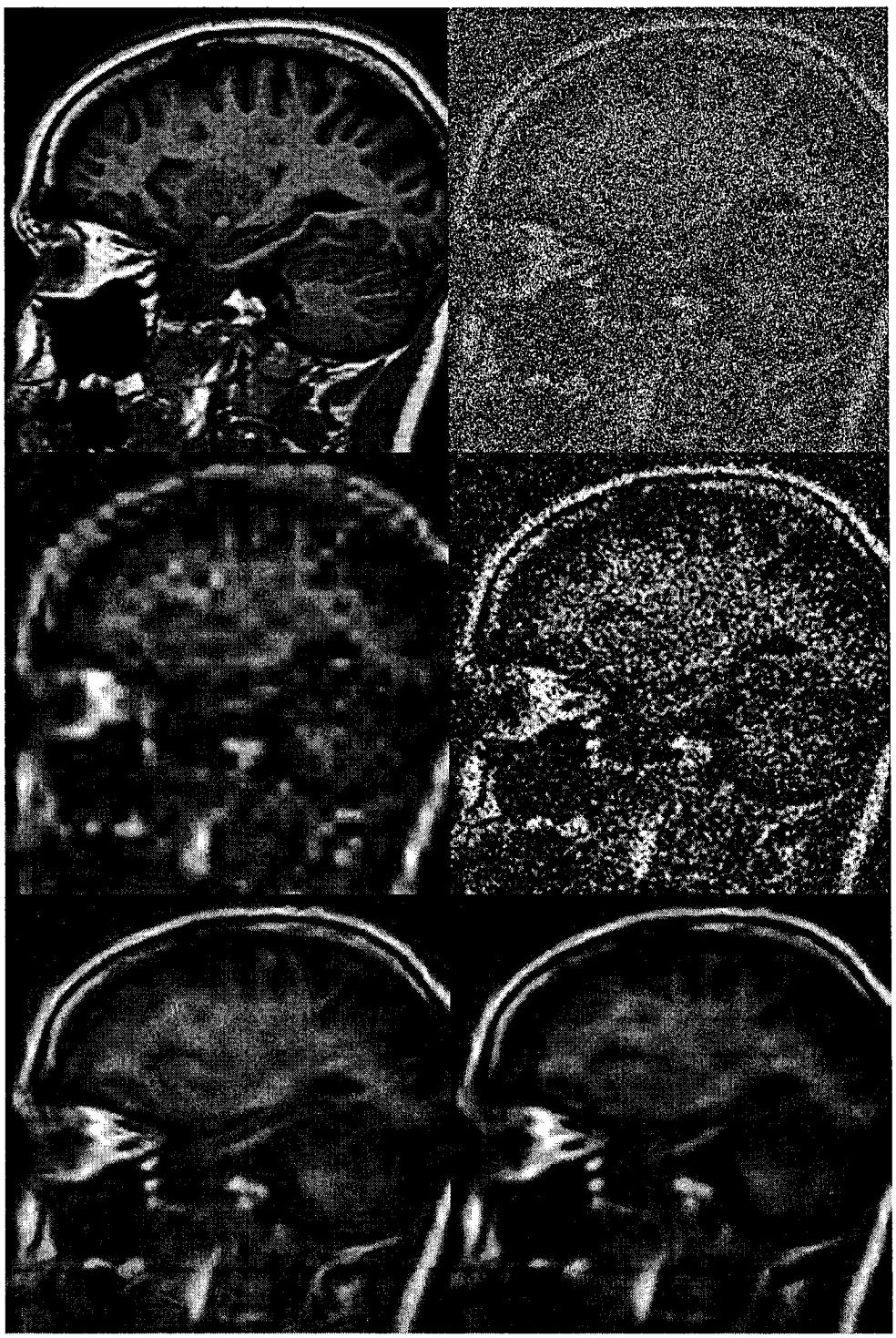


Figure 25: Denoised images using proposed algorithms for 512×512 *Brain* image with $\sigma$=255 : Original (top-left), Noisy (top-right; 0.00dB), *VisuShrink* soft (middle-left; 17.06dB), *Wiener* filter (middle-right; 11.51dB), *NeighLevel* (bottom-left; 20.02dB), Multivariate (bottom-right; 19.78dB).

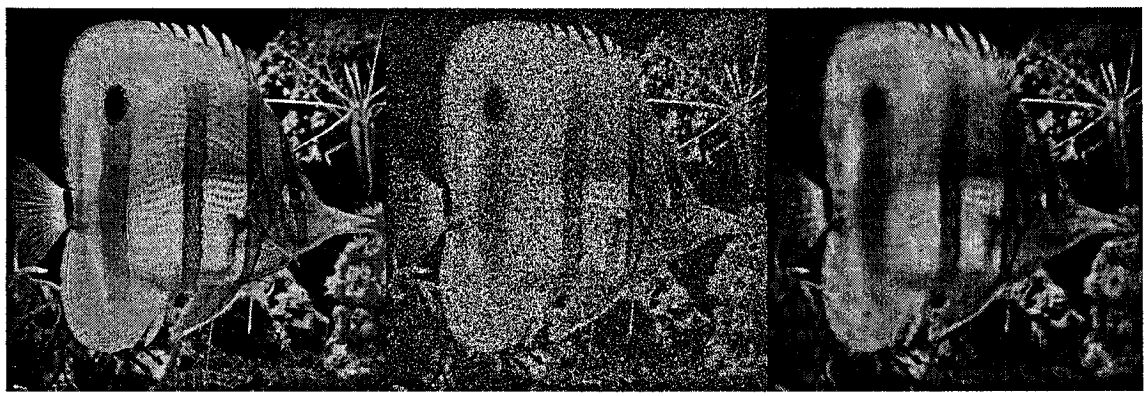


Figure 26: Denoised images using proposed algorithm for 512×512 *fish* image with $\sigma$=100 : Original (left), Noisy (right; 8.13dB), Denoised by *NeighLevel* (right; 21.57dB).

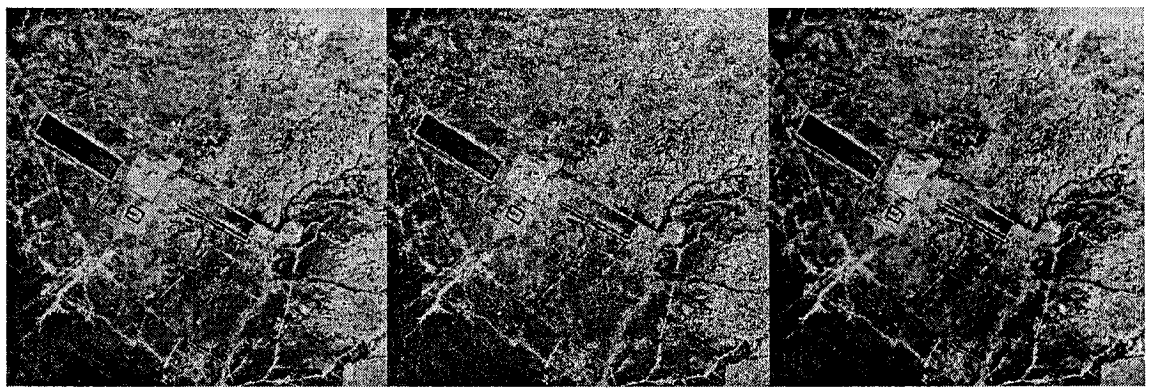


Figure 27: Denoised images using proposed algorithm for 512×512 satellite image with $\sigma$=30 : Original (left), Noisy (right; 18.59dB), Denoised by *NeighLevel* (right; 22.83dB).

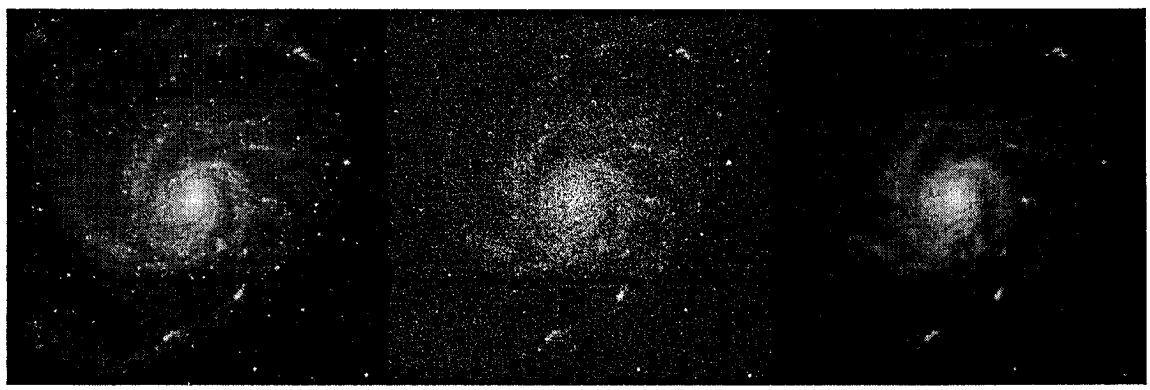


Figure 28: Denoised images using proposed algorithm for 512×512 astronomical image with $\sigma$=50 : Original (left), Noisy (right; 14.15dB), Denoised by *NeighLevel* (right; 28.48dB).

# Bibliography


[1] Test images from computer vision group in University of Granada, 2004. `http://decsai.ugr.es/cvg/dbimagenes/`.

[2] D. F. Andrews and C. L. Mallows. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society, Ser. B*, 36(1):99–102, 1974.

[3] A. Antoniadis, J. Bigot, and T. Sapatinas. Wavelet estimators in nonparametric regression: A comparative simulation study. *Journal of Statistical Software*, 6(6):1–83, 2001.

[4] S. Bacchelli and S. Papi. Matrix thresholding for multiwavelet image denoising. *Numerical Algorithms*, 33:41–52, 2003.

[5] E. Bala and A. Ertuzun. Applications of multiwavelet techniques to image denoising. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 581–584, 2002.

[6] K. Berkner and R. O. Wells. Smoothness estimates for soft-threshold denoising via translation-invariant wavelet transforms. *Applied and Computational Harmonic Analysis*, 12(1):1–170, January 2002.

[7] G. E. P. Box and G. C. Tiao. *Bayesian inference in statistical analysis*. Addison-Wesley, Reading, MA, 1992.

[8] R. W. Buccigrossi and E. P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Transactions on Image Processing*, 8(12):1688–1701, December 1999.

[9] T. D. Bui and G. Y. Chen. Translation invariant denoising using multiwavelets. *IEEE Transactions on Signal Processing*, 46(12):3414–3420, 1998.

[10] S. Cai and K. Li. Matlab implementation of wavelet transforms, 2003. `http://taco.poly.edu/WaveletSoftware/`.

[11] T. T. Cai and B. W. Silverman. Incorporating information on neighbouring coefficients into wavelet estimation. *Sankhya: The Indian Journal of Statistics, Serie A*, 63:127–148, 2001.

[12] E. J. Candes and D. J. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. Royal Society London A*, 357:2495–2509, 1999.

[13] E. J. Candes and D. J. Donoho. Curvelets, multiresolution representation, and scaling laws. In *Proceedings of SPIE Wavelet Applications in Signal and Image Processing VIII*, volume 4119, pages 1–12, 2000.

[14] S. G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532–1546, September 2000.

[15] S. G. Chang, B. Yu, and M. Vetterli. Spatially adaptive wavelet thresholding with context modeling for image denoising. *IEEE Transactions on Image Processing*, 9(9):1522–1531, September 2000.

[16] G. Y. Chen. Applications of wavelet transforms in pattern recognition and denoising. Master's thesis, Concordia University, Montreal, 1999.

[17] G. Y. Chen. *Wavelet and Ridgelet Transforms for Pattern Recognition and Denoising*. PhD thesis, Concordia University, Montreal, 2004.

[18] G. Y. Chen and T. D. Bui. Multiwavelet denoising using neighbouring coefficients. *IEEE Signal Processing Letters*, 10(7):211–214, 2003.

[19] H. Chipman, E. Kolaczyk, and R. McCulloch. Adaptive bayesian wavelet shrinkage. *Journal of the American Statistical Association*, 92(440):1413–1421, 1997.

[20] D. Cho and T. D. Bui. Multivariate statistical modeling for image denoising using wavelet transforms. Accepted by *Signal Processing : Image Communication*, 2004.

[21] D. Cho, T. D. Bui, and G. Y. Chen. Image denoising based on wavelet shrinkage using neighbour and level dependency. Submitted to *IEEE Transactions on Signal Processing*, 2004.

[22] H. Choi and R. G. Baraniuk. Analysis of wavelet-domain wiener filters. In *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pages 613–616, October 1998.

[23] H. Choi, J. K. Romberg, R. G. Baraniuk, and N. G. Kingsbury. Hidden markov tree modeling of complex wavelet transforms. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 133–136, Istanbul, Turkey, June 2000.

[24] C. K. Chui. *An Introduction to Wavelets*. Academic Press, 1992.

[25] C. K. Chui and J. A. Lian. A study of orthonormal multiwavelets. *J. Appl. Numer. Math.*, 20:272–298, 1996.

[26] I. Cohen, S. Raz, and D. Malah. Translation-invariant denoising using the minimum description length criterion. *Signal Processing*, 75(3):201–223, June 1999.

[27] R. R. Coifman and D. L. Donoho. Translation-invariant denoising. *Wavelets and Statistics, Springer Lecture Notes in Statistics 103, New York:Springer-Verlag*, pages 125–150, 1995.

[28] P. Craven and G. Wahba. Smoothing noisy data with spline functions – estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, 31:377–404, 1979.

[29] M. S. Crouse, R. D Nowak, and R. G. Baraniuk. Wavelet-based signal processing using hidden markov models. *IEEE Transactions on Signal Processing*, 46(4):886–902, 1998.

[30] I. Daubechies. *Ten Lectures On Wavelets*. SIAM, 1992.

[31] M. N. Do and M. Vetterli. Contourlets. In G. V. Welland, editor, *Beyond Wavelets*, pages 83–106. Academic Press, 2003.

[32] M. N. Do and M. Vetterli. The finite ridgelet transform for image representation. *IEEE Transactions on Image Processing*, 12(1):16–28, January 2003.

[33] D. L. Donoho. Denoising by soft-thresholding. *IEEE Transactions on Information Theory*, 41:613–627, 1995.

[34] D. L. Donoho, M. R. Duncan, X Huo, and O. Levi. Wavelab 8.02, 1999. `http://www-stat.stanford.edu/~wavelab/`.

[35] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 8(3):425–455, 1994.

[36] D. L. Donoho and I. M. Johnstone. Adaptating to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.

[37] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard. Wavelet shrinkage: asymptopia? *Journal of the Royal Statistical Society, Ser. B*, 57:301–337.

[38] T. R. Downie and B. W. Silverman. The discrete multiple wavelet transform and thresholding methods. *IEEE Transactions on Signal Processing*, 46(9):2258–2562, 1998.

[39] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley, New York, 2nd edition, 2000.

[40] S. Efromovich, J. Lakey, M. C. Pereyra, and N. Tymes. Data-driven and optimal denoising of a signal and recovery of its derivative using multiwavelets. *IEEE Transactions on Signal Processing*, 52(3):628–635, March 2004.

[41] G. L. Fan and X. G. Xia. Image denoising using local contextual hidden markov model in the wavelet domain. *IEEE Signal Processing Letters*, 8(5):125–128, May 2001.

[42] G. L. Fan and X. G. Xia. Improved hidden markov models in the wavelet-domain. *IEEE Transactions on Signal Processing*, 49(1):115–120, January 2001.

[43] F. Fernandes, I. Selesnick, R. van Spaendonck, and C. Burrus. Complex Wavelet Transforms with Allpass Filters. *Signal Processing*, 83(8):1689–1706, August 2003.

[44] F. Fernandes, R. van Spaendonck, and C. Burrus. A New Framework for Complex Wavelet Transforms. *IEEE Transactions on Signal Processing*, 51(7):1825–1837, July 2003.

[45] C. Fernández, J.Osiewalski, and M. Steel. Modelling and inference with $v$-spherical distributions. *Journal of the American Statistical Association*, 90(432):1331–1340, 1995.

[46] J. S. Geronimo, D. P. Hardin, and P. R. Massopust. Fractal functions and wavelet expansions based on several scaling functions. *J. Approx. Theory*, 78(3):373–401, 1994.

[47] R. C. Gonzalez and R. E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2nd edition, 2002.

[48] P. J. Green and B. W. Silverman. *Nonparametric regression and generalized linear models : a roughness penalty approach*. Chapman & Hall, London, 1994.

[49] T. C. Hsung and D.P.-K. Lun. Generalized cross validation for multiwavelet shrinkage. *IEEE Signal Processing Letters*, 11(6):549–552, June 2004.

[50] M. Jansen. *Noise reduction by wavelet thresholding*, volume 161 of *Lecture notes in statistics*. Springer, New York, 2001.

[51] M. Jansen and A. Bultheel. Multiple wavelet threshold estimation by generalized cross validation for images with correlated noise. *IEEE Transactions on Image Processing*, 8(7):947–953, July 1999.

[52] M. Jansen, M. Malfait, and A. Bultheel. Generalized cross validation for wavelet thresholding. *Signal Processing*, 56(1):33–44, January 1997.

[53] M. Kazubek. Wavelet domain image denoising by thresholding and wiener filtering. *IEEE Signal Processing Letters*, 10(11):324–326, November 2003.

[54] N. G. Kingsbury. The dual-tree complex wavelet transform: a new efficient tool for image restoration and enhancement. In *Proc. EUSIPCO*, pages 319–322, Rhodes, Greece, 1998.

[55] N. G. Kingsbury. Image processing with complex wavelets. *Phil. Trans. Royal Society London A*, 357:2543–2560, September 1999.

[56] N. G. Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Applied and Computational Harmonic Analysis*, 10(3):234–253, May 2002.

[57] A. Laine, E. Angelini, Y. Jin, P. Esser, and R. V. Heertum. Fusion of brushlet and wavelet denoising methods for nuclear images. In *Proc. IEEE Intl. Symp. Biomedical Imaging*, pages 1188–, Arlington, VA, 2004.

[58] W. Lawton. Applications of complex valued wavelet transforms to subband decomposition. *IEEE Transactions on Signal Processing*, 41(12):3566–3568, December 1993.

[59] J. S. Lee. Digital image enhancement and noise filtering by use of local statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:165–168, 1980.

[60] X. Li and M. T. Orchard. Spatially adaptive image denosing under overcomplete expansion. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 300–303, September 2000.

[61] J. M. Lina and B. MacGibbon. Non-linear shrinkage estimation with complex daubechies wavelets. In *Proceedings of SPIE Wavelet Applications in Signal and Image Processing V*, volume 3169, pages 67–79, 1997.

[62] J. M. Lina and M. Mayrand. Complex daubechies wavelets. *Applied and Computational Harmonic Analysis*, 2(3):219–229, 1995.

[63] W. Ling. Orthogonal multiwavelets transform for image denoising. In *Proc. ICSP*, volume 2, pages 987–991, 2000.

[64] J. Magarey and N. Kingsbury. An improved motion estimation algorithm using complex wavelets. In *IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 969–972, September 1996.

[65] M. Malfait and D. Roose. Wavelet-based image denoising using a markov random field a priori model. *IEEE Transactions on Image Processing*, 6:549–565, April 1997.

[66] S. G. Mallat. A compact multiresolution representation: the waavelet model. In *Proc. IEEE Computer Society Workshop on Computer Vision*, pages 2–7, Washington, D.C., 1987. IEEE Computer Society Press.

[67] S. G. Mallat. A theory for multiresolution signal decomposition:the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.

[68] S. G. Mallat. *A wavelet tour of signal processing*. Academic Press, 2nd edition, 1999.

[69] F. G. Meyer and R. R. Coifman. Brushlets: A tool for directional image analysis and image compression. *Applied and Computational Harmonic Analysis*, 4(2):147–187, April 1997.

[70] M. K. Mihcak, K. Ramchandran I. Kozintsev, and P. Moulin. Low-complexity image denoising based on statistical modeling of wavelet coefficients. *IEEE Signal Processing Letters*, 6(12):300–303, 1999.

[71] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[72] G. P. Nason. Wavelet shrinkage using cross-validation. *Journal of the Royal Statistical Society, Ser. B*, 58(2):463–479, 1996.

[73] M. Nikolova. Markovian reconstruction using a gnc approach. *IEEE Transactions on Image Processing*, 8(9):1204–1220, September 1999.

[74] A. Pizurica, W. Philips, I. Lemahieu, and M. Acheroy. A joint inter- and intrascale statistical model for bayesian wavelet based image denoising. *IEEE Transactions on Image Processing*, 11(5):545–557, May 2002.

[75] J. Portilla. Bls-gsm image denoising toolbox, 2004. `http://decsai.ugr.es/~javier/denoise/`.

[76] J. Portilla and E. Simoncelli. Image restoration using gaussian scale mixtures in the wavelet domain. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 965–968, Barcelona, Spain, September 2003.

[77] J. Portilla, V. Strela, M.Wainwright, and E. Simoncelli. Adaptive wiener denoising using a gaussian scale mixture model. In *IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 37–40, 2001.

[78] J. Portilla, M. Wainwright V. Strela, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, November 2003.

[79] M. Rangaswamy, D. D. Weiner, and A. Ozturk. Non-gaussian random vector identification using spherically invariant random processes. *IEEE Transactions on Aerospace and Electronic Systems*, 29(1):111–124, January 1993.

[80] S. Ray and B. K. Mallick. Bayesian transformation model for wavelet shrinkage. *IEEE Transactions on Image Processing*, 12(12):1512–1521, December 2003.

[81] S. J. Roberts, D. Husmeier, I. Rezek, and W. Penny. Bayesian approaches to gaussian mixture modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1133–1142, November 1998.

[82] J. K. Romberg, H. Choi, and R. G. Baraniuk. Bayesian tree-structured image modeling using wavelet-domain hidden markov models. *IEEE Transactions on Image Processing*, 10(7):1056–1068, July 2001.

[83] S. Sardy. Minimax threshold for denoising complex signals with waveshrink. *IEEE Transactions on Signal Processing*, 48(4):1023–1028, April 2000.

[84] L. Sendur and I. W. Selesnick. Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency. *IEEE Transactions on Signal Processing*, 50(11):2744–2756, 2002.

[85] L. Sendur and I. W. Selesnick. Bivariate shrinkage with local variance estimation. *IEEE Signal Processing Letters*, 9(12):438–441, 2002.

[86] J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, December 1993.

[87] W. Shengqian, Z. Yuanhua, and Z. Daowen. Adaptive shrinkage de-noising using neighbourhood characteristic. *Electronics Letters*, 38(11):502–503, 2002.

[88] E. P. Simoncelli. Statistical models for images: compression, restoration and synthesis. In *Conference Record of the 31st Asilomar Conference on Signals, Systems & Computers*, volume 1, pages 673–678, November 1997.

[89] E. P. Simoncelli. Bayesian denoising of visual images in the wavelet domain. In P Müller and B Vidakovic, editors, *Bayesian Inference in Wavelet Based Models*, volume 141 of *Lecture Notes in Statistics*, pages 291–308. Springer-Verlag, August 1999.

[90] E. P. Simoncelli and E. H. Adelson. Non-separable extensions of quadrature mirror filters to multiple dimensions. *Proceedings of the IEEE*, 78(4):652–664, 1990.

[91] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Transactions on Information Theory*, 38(2):587–607, 1992.

[92] J. L. Starck, E. J. Candes, and D. L. Donoho. Very high quality image restoration by combining wavelets and curvelets. *Proc. SPIE*, 4478:9–19, December 2001.

[93] J. L. Starck, E. J. Candes, and D. L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on Image Processing*, 11(6):670–684, June 2002.

[94] C. M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, November 1981.

[95] G. Strang and T. Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Wellesley, MA, 1997.

[96] V. Strela, P. N. Heller, G. Strang, P. Topiwala, and C. Heil. The application of multiwavelet filter banks to image processing. *IEEE Transactions on Image Processing*, 8(4):548–563, April 1999.

[97] V. Strela and A. T. Walden. Signal and image denoising via wavelet thresholding: Orthogonal and biorthogonal, scalar and multiple wavelet transforms. In W.F. Fitzgerald, R.L. Smith, A.T. Walden, and P.C. Young, editors, *Nonlinear and Nonstationary Signal Processing*, pages 395–441. Cambridge University Press, 2001.

[98] A. Tonazzini and L. Bedini. Monte carlo markov chain techniques for unsupervised mrf-based image denoising. *Pattern Recognition Letters*, 24(1-3):55–64, January 2003.

[99] P. Vandergheynst and J. F. Gobbers. Directional dyadic wavelet transforms: design and algorithms. *IEEE Transactions on Image Processing*, 11(4):363–372, April 2002.

[100] N. Weyrich and G. T. Warhola. Denoising using wavelets and cross-validation. In S. P. Singh, editor, *Approximation Theory, Wavelets and Applications*, volume 454, pages 523–532. Kluwer, 1995.

[101] N. Weyrich and G. T. Warhola. Wavelet shrinkage and generalized cross validation for image denoising. *IEEE Transactions on Image Processing*, 7(1):82–90, January 1998.

[102] X. G. Xia. A new prefilter design for discrete multiwavelet transform. *IEEE Transactions on Signal Processing*, 46(6):1558–1570, June 1998.

[103] X. G. Xia, J. S. Geronimo, D. P. Hardin, and B. W. Suter. Design of prefilters for discrete multiwavelet transforms. *IEEE Transactions on Signal Processing*, 44(1):25–35, January 1996.

[104] H. Xie, L. E. Pierce, and F. T. Ulaby. Sar speckle reduction using wavelet denoising and markov random field modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 40(10):2196–2212, October 2002.

[105] J. Xie, D. Zhang, and W. Xu. Spatially adaptive wavelet denoising using the minimum description length principle. *IEEE Transactions on Image Processing*, 13(2):179–187, February 2004.

[106] K. Yao. A representation theorem and its applications to spherically-invariant random processes. *IEEE Transactions on Information Theory*, 19(5):600–608, September 1973.