

Wavelet-based Image and Video Processing

by

Fu Jin

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Systems Design Engineering

Waterloo, Ontario, Canada, 2004

©Fu Jin 2004

I hereby declare that I am the sole author of this thesis.

I authorize the University of Waterloo to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Fu Jin

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

Fu Jin

Acknowledgements

I would like to thank my supervisor Professor Paul Fieguth for this continual guidance and financial support throughout this endeavor. I feel that a large portion of my learning at University of Waterloo is a direct result of working with Prof. Fieguth. Also, special thanks are due to my co-supervisor Dr. Lowell Winger for his generous financial support and introducing me to the interesting area of video processing.

I am also very grateful for the support and assistance of my fellow graduate students, especially those with the VIP lab. Their kindness makes my study a pleasurable period.

I would acknowledge with thanks the financial supports from the Department of Systems Design Engineering and University of Waterloo.

Last but not least, I would thank my family for their endless support. Special thanks go to my wife, Christine Liang, for her patience and love.

Abstract

Since the last decade the study of the wavelet transform has attracted many efforts and made great progresses. However, the research interest is still growing. In this thesis, we investigate the applications of the wavelet transform to image and video processing.

We first study the wavelet transform mainly from a statistical point of view. Specifically, we study the statistics of the wavelet coefficients of 1-D and 2-D Gaussian autoregressive processes, since these models are widely-used in signal and image processing. We found the covariance matrices of the wavelet coefficients are quite sparse with significant correlations concentrating in the inter- and intra-scale neighborhood. We thus propose to perform the strip Kalman filtering in the wavelet domain (instead of in the spatial domain) and propose a procedure to find optimal strips under complexity constraints. Experimentally, we showed that the structure of the wavelet-based strip Kalman filter is not sensitive to noise strength and achieved much better or comparable error variances than its spatial counterpart.

We then proceed to use the wavelet transform for natural image denoising. It is well known the current wavelet-based image denoising approaches suffers from Gibbs-like denoising artifacts. We found this problem is mainly due to neglecting strong correlations of wavelet coefficients, especially in the edge areas. We thus propose to remove edges (statistical means) prior to performing the wavelet transform. This is achieved through a clustering-based nonlinear weighting processing.

Another wavelet-based image processing we studied is the multiresolution image enhancement. In the wavelet domain, the image enhancement problem can be treated as estimating the (lost) high-frequency wavelet coefficients from the low-frequency ones. We

achieved this goal through estimating the Lipschitz exponents and using this quantity to describe the relationships of wavelet coefficients at different subbands.

For the wavelet-based video processing, we developed a new multiresolution motion estimation approach. This approach estimates motion vectors in all subbands jointly with a robust nonlinear regularization. To handle possible inaccurate motion estimates we used a motion detection procedure. If motion estimation is accurate we perform a scalar Kalman filtering along the motion trajectory to exploit high temporal correlation; otherwise we simply perform wavelet-based image denoising as different frames are assumed to be uncorrelated. Because all these processing is in the wavelet domain, our approach can well preserve spatial details and achieved small estimation error.

Contents

1	Introduction	1
1.1	Thesis Motivations and Contributions	1
1.2	Thesis Organization	3
2	Background	7
2.1	Discrete Wavelet Transform	7
2.1.1	Bases and Series Expansion	8
2.1.2	Discrete Wavelet Bases	10
2.1.3	Wavelet-based Statistical Models	17
2.2	Bayesian Estimation	21
2.2.1	Static Estimation	21
2.2.2	Dynamic Estimation	22
3	Complex Wavelet Transforms	27
3.1	Problems with Real-valued Wavelet Transforms	28
3.2	Complex Wavelet Transform Properties	36
3.2.1	Single-tree and Dual-tree Complex Wavelets	37

3.2.2	Shift-invariant 1-D Dual-tree Complex Wavelet	40
3.2.3	2-D Complex Wavelet Transform	51
4	Wavelet-based Image Enhancement	61
4.1	Problem Formulation	63
4.2	1-D Mallat-Zhong Wavelet	64
4.3	Lipschitz Exponent Estimation	71
4.4	Experimental Results	75
4.5	Discussion	86
5	Wavelet Statistics	91
5.1	1-D Gaussian AR Process	93
5.2	2-D Gaussian Process	110
5.3	Wavelet-based 2-D Strip Kalman Filtering	119
5.3.1	Dynamic State Equations of the NSHP Model and Strip Filter . . .	120
5.3.2	Optimal Strip Structures	124
5.3.3	Spatial and Wavelet-based Strip KFs	126
5.3.4	Strip-based Denoising Experiments	133
6	Wavelet-based Image Denoising	147
6.1	Problem formulation	150
6.2	Adaptive weighted averaging (AWA)	156
6.3	Wavelet-based image denoising system	164
6.4	Experimental results	167

7	Wavelet-based Video Denoising	175
7.1	Problem formulation	177
7.2	The Denoising System	180
7.3	Experimental results	186
7.4	Conclusions	189
8	Thesis Summary	193

List of Tables

3.1	Examples of orthogonal real and complex wavelet filters.	32
3.2	Table of coefficients for the complex analysis filters.	49
5.1	Compare the optimal spatial and wavelet strip filters for $\sigma_v^2 = 0.01$	130
5.2	Compare the optimal spatial and wavelet strip filters for $\sigma_v^2 = 0.1$	130
5.3	Compare the optimal spatial and wavelet strip filters for $\sigma_v^2 = 1$	131
5.4	Compare the optimal spatial and wavelet strip filters for $\sigma_v^2 = 10$	131
5.5	δ_{MSE} of the spatial and wavelet strip KFs	144
6.1	Comparison the traditional and AWA-based spatial Lee filters.	159
6.2	Compare the performance of 12 denoising approaches.	168
6.3	PSNR results of different methods for image <i>Lena</i>	169
6.4	PSNR results of different methods for image <i>Barbara</i>	169
7.1	Percentage increase δ_{MSE} in estimation error.	180
7.2	Comparison of PSNR of several denoising approaches	189

List of Figures

2.1	Compare The Fourier and wavelet basis functions	8
2.2	Nested spaces spanned by scaling functions.	11
2.3	Spaces spanned by the scaling and wavelet functions	12
2.4	The DWT can be implemented using filter banks followed by down-sampling.	14
2.5	Time-frequency tiles of the Fourier, STFT [1] and the wavelet transforms.	14
2.6	Discrete wavelet transform of a chirp signal.	16
2.7	An example of GGD distribution functions.	18
2.8	HMT structure.	20
2.9	An example of strip structures.	24
3.1	Real and complex wavelet bases.	29
3.2	Shift-variance of the Haar wavelet transform.	31
3.3	Decimated and undecimated wavelet transforms.	33
3.4	Idealized spectrum of the tensor product of two real-valued $1D$ wavelets.	34
3.5	Poor directional selectivity of the real-valued $2D$ separable wavelet.	35
3.6	Diagram of a two-level decomposition of 1-D complex wavelet transform	37
3.7	Two-level decomposition of the dual-tree complex wavelet transform.	39

3.8	Treat dual-tree complex wavelet transform as a black box.	39
3.9	One-level wavelet decomposition with decimation.	41
3.10	Comparison of aliasing effect of real and complex digital filters.	42
3.11	Basic configuration of the dual tree complex wavelets.	44
3.12	Scaling and wavelet functions for the dual-tree complex wavelet transform.	47
3.13	Shift sensitivity of the complex-valued and real-valued wavelets.	48
3.14	Two levels of the complex wavelet transform.	53
3.15	Compare the reconstructions of shifted circles.	55
3.16	2-D dual-tree complex wavelet transform.	56
3.17	The bell-shaped envelope of complex wavelet basis.	56
3.18	Compare the directionality of real and complex wavelets.	57
3.19	Compare the reconstruction of a circle from a real and complex wavelets.	58
4.1	The block diagram of the multiscale image enhancement system.	62
4.2	Mallat-Zhong scaling and wavelet functions.	64
4.3	An example of the 3-level 1-D Mallat-Zhong wavelet transform.	65
4.4	Reconstructed signal from the multiscale edge representation.	67
4.5	An example of the 3-level 2-D Mallat-Zhong wavelet transform.	69
4.6	2-D multiscale edges. Note the high dependence among 3 levels.	70
4.7	Image <i>House</i> and its multiscale edge reconstruction.	70
4.8	Different types of edges.	71
4.9	Compare the behavior of various edges in multiscale representation.	72
4.10	Enhancement results of two 1-D signals.	76
4.11	Examining enhancement results in the Fourier domain.	77

4.12	Enhancement results of the synthetic image <i>Square</i>	78
4.13	Spectra of synthetic image <i>Square</i>	79
4.14	Enhancement results of image <i>Lena</i>	80
4.15	Spectra of image <i>Lena</i>	81
4.16	Enhancement results of image <i>House</i>	82
4.17	Spectra of image <i>House</i> before and after enhancement.	83
4.18	Enhancement results of image <i>Mandrill</i>	84
4.19	Spectra of image <i>Mandrill</i>	85
4.20	Compare Greenspan's approach with the proposed one.	88
5.1	Autocovariance of a Gaussian AR process and its one-level Haar transform.	92
5.2	Correlation coefficient matrices of 4-levels of Haar transform.	97
5.3	Variances of the Haar transform of the Gaussian AR process $\alpha = 0.95$	98
5.4	Correlation coefficient matrices of Daubechies <i>db2</i> transform.	101
5.5	Variances of the <i>db2</i> transform of the Gaussian AR process $\alpha = 0.95$	102
5.6	Comparing the correlation structure of Haar and <i>db2</i> wavelets.	103
5.7	Correlation matrices of two nonstationary processes.	106
5.8	Correlation coefficient matrix of Haar transform of $y(n)$	107
5.9	Correlation coefficient matrix of Haar transform of $z(n)$	108
5.10	Synthesized $y(n)$ with independence model.	109
5.11	Non-symmetric half plane model	111
5.12	Covariances of the NSHP process.	112
5.13	Variances of the wavelet coefficients with Haar transform.	115
5.14	Variances of the two-level Haar transform.	117

5.15	Parent-children relationship.	118
5.16	Correlation coefficients of 5-level <i>db2</i> and Haar wavelets.	118
5.17	An example optimal KF gain matrix.	123
5.18	Spatial and wavelet (Haar and <i>db2</i>) strip structures.	128
5.19	Compare the denoising results of spatial and wavelet strip filters.	135
5.20	Compare length 1 strip filters.	136
5.21	Compare spatial and wavelet strip filters for $\sigma_v^2 = 10$	137
5.22	Plot of the error confidence intervals of 4 strip filters.	138
5.23	Periodic error pattern with independence model.	139
5.24	Denoising results of spatial and wavelet strip filters $\sigma_v^2 = 0.01$	141
5.25	Denoising results of spatial and wavelet strip filters $\sigma_v^2 = 0.1$	142
5.26	Denoising results of spatial and wavelet strip filters $\sigma_v^2 = 10$	143
5.27	Periodic error pattern of the wavelet strip KF with size 1.	145
6.1	Gibbs-like artifact with the independence model in the wavelet domain. . .	156
6.2	Local mean estimation using AWA and the traditional sample mean.	160
6.3	Denoising results of <i>Circle</i> of AWA-based and traditional spatial Lee filters	161
6.4	Denoising results of <i>Lena</i> of AWA-based and traditional spatial Lee filters .	162
6.5	Similar test as Fig. 6.4 but for the image <i>Lena</i>	163
6.6	Block diagram of the proposed image denoising system	165
6.7	The original and noisy images of the image <i>Cameraman</i>	167
6.8	Denoising results of <i>Barbara</i> of AWA-based and traditional wavelet Lee filters	170
6.9	Compare the denoising results of complex and overcomplete wavelets. . . .	172
7.1	The video denoising system	181

7.2	Examples of commonly-used penalty functions.	182
7.3	Motion estimation using traditional MRBM and the proposed approach . .	187
7.4	Averaged PSNR versus γ curve	188
7.5	Denoising results of the <i>Paris</i> sequence	190
7.6	Denoising results of the <i>Salesman</i> sequence	191

Chapter 1

Introduction

1.1 Thesis Motivations and Contributions

Since the last decade, the study of wavelet transforms has attracted many efforts. We can see great progresses both in theory and in practice. However, the research interest in this field is still growing.

Understanding wavelet transforms (especially from a statistical point of view) and applying them in image and video processing are the main motivations of this thesis. With these goals in mind, we achieved the following research results:

- A study of the statistics of wavelet coefficients. Specifically, our focus is on autoregressive (AR) processes since they are widely-used in image and video modeling. We show that the covariance of wavelet coefficients is typically sparse with remaining correlations only in the intra- and inter-scale neighborhood. To exploit this special correlation structure, we design the wavelet-based strip Kalman filter for image denoising, which compares favorably to its spatial counterpart.

- The image enhancement problem can also be addressed in the wavelet domain by utilizing the inter-scale dependencies of wavelet coefficients. By adapting to different edge shapes, we successfully estimate the wavelet coefficients in high-frequency subbands using those in lower-frequency subbands. In this way, we achieve high-quality frequency extrapolation and image enhancement.
- Wavelet-based image denoising has been studied extensively in the literature. However, the strong correlations of wavelet coefficients in edge areas are usually either neglected or improperly modelled, which results in annoying Gibbs-like artifacts in denoised images. To represent edges more accurately, and thus to significantly suppress the Gibbs-like artifacts, we propose a pre-processing algorithm which adaptively estimates edge strength and properly removes edges prior to the wavelet transform. We test the proposed approach on many standard test images and show impressive denoising results, in both subjective and objective senses.
- Although wavelet-based image denoising has been studied extensively, relatively little has been carried out on wavelet-based video denoising. To fully exploit the very high temporal correlations of many typical video sequences, robust motion estimation / compensation are needed. Unlike the standard multiresolution motion estimation approach, we propose to perform motion estimation on all scales *jointly*. Furthermore, to better resist noise and to utilize the strong correlations in the motion field, we use the ideas of robust statistics and impose a *nonlinear* smoothness constraint. This kind of regularization approach can well adapt to the discontinuities among motion objects and at the same time constrain smooth motions. Experimentally, our approach greatly outperforms traditional multiresolution motion estimation.

1.2 Thesis Organization

This thesis divides into eight chapters, beginning with a background discussion and proceeding through image and video denoising.

Chapter 2 presents the general background of wavelet transforms and statistical estimation. The material is meant to be concise yet comprehensive for readers who are not familiar with wavelets and / or estimation theory. The first part of Chapter 2 addresses the links between the discrete wavelet transform and linear basis functions. We review several important statistical models of wavelet coefficients proposed in the literature. In the second part of this chapter, we talk about the two main categories of estimation: static and dynamic, both of which are summarized for the Gaussian / second-order case. An approximate (but more practical) implementation of the optimal dynamic estimator is introduced as the strip Kalman filter.

Chapter 3 presents the background material of a very important class of wavelet transforms: the complex wavelet transform (CWT). We first compare the real-valued and complex-valued wavelet transforms, and show the special (and interesting) properties of the CWT. We then revisit the design process to get in-depth understanding of CWT. The ability of the CWT to conveniently construct directionally selective high dimensional wavelet bases is illustrated in theory and by experimental results. Finally, we use a synthetic image denoising problem to show the advantages of the CWT over the standard real-valued wavelet transform.

The contributions of this thesis are presented in chapters 4 through 7. In Chapter 4 we present a new wavelet-based multiscale image enhancement approach. This work is meant to exploit the interscale dependency of wavelet coefficients. By estimating Lipschitz

exponents and other parameters which control the smoothness of singularities, we are able to adapt to different shapes of discontinuities and thus to properly estimate the wavelet coefficients in a high-frequency subband from those in low-frequency subbands. We then achieve enhancement by performing inverse wavelet transform using the estimated high-frequency coefficients.

In Chapter 5, we study the statistics of the wavelet coefficients for Gaussian autoregressive processes. We show that the wavelet covariance is very sparse with a special structure: strong correlations exist only in intra- and inter-scale neighborhoods. Based on this property of the wavelet transform we design wavelet-based strip Kalman filters and apply them to image denoising. We show that the wavelet-based strip Kalman filters significantly outperform their spatial counterparts under both objective and subjective criteria.

Chapter 6 presents our new approach to wavelet-based image denoising. Noting that the existing models can not well represent the correlations in edge areas we propose a pre-processing step to estimate edge strengths in the spatial domain and then to try to remove edges prior to taking wavelet transform. Our pre-processor is a nonlinear edge-strength estimator, which works like the widely-used Sigma filter. After pre-processing the edges, the modified images become more stationary and better suited for wavelet transforms. We then apply the well-developed wavelet-based image denoising methods on the modified images to remove noise.

Chapter 7 addresses the wavelet-based video denoising problem. To utilize strong temporal correlations among frames motion analysis is critical. Unlike the standard multiresolution motion estimation, which performs motion estimation from scale to scale, we estimate motion on all levels jointly. In order to be robust to noise and well adapt to mo-

tion discontinuity, we perform a nonlinear regularization on the motion field. With robust motion information in hand, we then achieve denoising by a Kalman filter in the wavelet domain.

Finally, Chapter 8 presents a summary of this work and suggestions for future work.

Chapter 2

Background

This chapter presents the background related to wavelet transforms and to statistical estimation, which will be used extensively in the following chapters.

2.1 Discrete Wavelet Transform

Although the Fourier transform has been the mainstay of transform-based signal processing since the late 1950s, a more recent transformation, called the *wavelet transform*, has shown great promise for denoising, compression and classification. We are not claiming that the wavelet transform can replace the Fourier transform. However unlike the Fourier transform, whose basis functions are sinusoids, wavelet transforms are based on small waves of varying frequency and *limited* duration, as shown in Fig. 2.1. This allows wavelets to provide both spatial / time and frequency information (hence time-frequency analysis), whereas the non-local Fourier transform gives only frequency information.

Mallat [3] first showed that the wavelet transform provides the foundation of a powerful

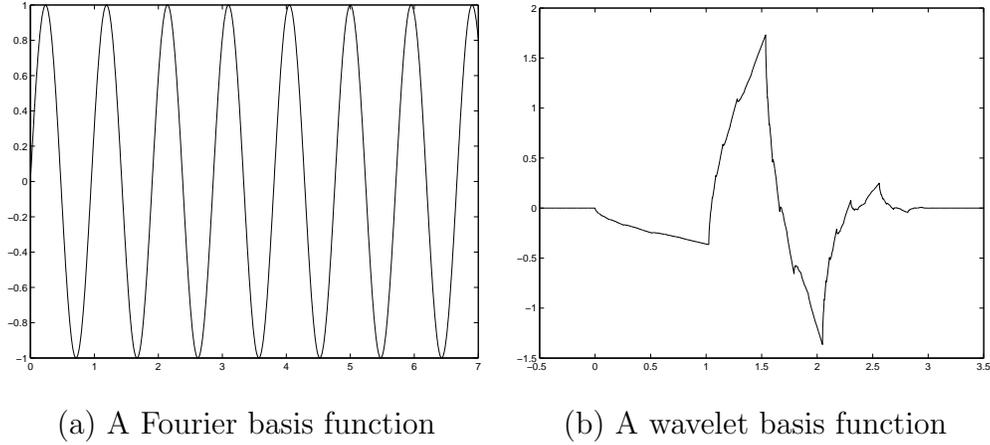


Figure 2.1: An example of the basis functions of the Fourier and wavelet transforms. Note the support of wavelet basis is finite but Fourier basis oscillates forever.

new approach to signal processing and analysis, called *multiresolution* analysis (MRA). Multiresolution theory unifies techniques from several fields, including subband coding from signal processing [4], quadrature mirror filtering (QMF) from speech recognition [5], and pyramidal image processing [6]. Due to the close links to these techniques, the wavelet transform has found many applications [7],[8],[9],[10].

2.1.1 Bases and Series Expansion

A set of vectors $\phi_k(x)$ spans a vector space \mathcal{V} if any element of that space can be expressed as a linear combination of members of that set. Given the set of functions $\phi_k(x)$ we define $Span_k \phi_k(x) = \mathcal{V}$ as the vector space with all elements of the space of the form:

$$\mathcal{V} = \left\{ f(x) = \sum_k \alpha_k \phi_k(x) \right\} \quad (2.1)$$

where $k \in \mathcal{Z}$ and $x, \alpha_k \in \mathcal{R}$ ¹.

¹ \mathcal{R} is the set of real numbers. \mathcal{Z} is the set of integers.

We say that the set $\phi_k(x)$ is a *basis set* or a *basis* for a given space \mathcal{V} if the set of $\{\alpha_k\}$ in (2.1) are unique for any $f(x) \in \mathcal{V}$. There are three typical cases for $\phi_k(x)$:

Case 1: If the set $\phi_k(x)$ form an *orthonormal* basis for \mathcal{V} , meaning that

$$\langle \phi_j(x), \phi_k(x) \rangle = \delta_{jk} = \begin{cases} 0 & j \neq k \\ 1 & j = k \end{cases} \quad (2.2)$$

where $\langle \cdot \rangle$ denotes the integral inner product ². In this case

$$\alpha_k = \langle \phi_k(x), f(x) \rangle \quad (2.3)$$

Case 2: When the basis set $\phi_k(x)$ is not orthonormal, one can still have the expression (2.1) and one similar to (2.3) by using a *dual basis set* $\tilde{\phi}_k(x)$, whose elements are not orthogonal to each other, but to the corresponding element of the expansion set

$$\langle \phi_j(x), \tilde{\phi}_k(x) \rangle = \delta_{jk} \quad (2.4)$$

Because this type of “orthogonality” requires two sets of vectors, the expansion set and the dual set, the system is called “biorthogonal” [11]. Using (2.4) we have

$$\alpha_k = \langle \tilde{\phi}_k(x), f(x) \rangle \quad (2.5)$$

Case 3: If the expansion set is not a basis for \mathcal{V} , but supports the expansion defined in (2.1), it is a spanning set in which there is more than one set of α_k for any $f(x) \in \mathcal{V}$. The expansion functions and their duals are said to be *overcomplete* or redundant. They form a *frame* [11] in which ³

$$A\|f(x)\|^2 \leq \sum_k |\langle \phi_k(x), f(x) \rangle|^2 \leq B\|f(x)\|^2 \quad (2.6)$$

²The integral inner product of two real functions $f(x)$ and $g(x)$ is $\langle f(x), g(x) \rangle = \int_{-\infty}^{\infty} f(x)g(x)dx$.

³The norm of $f(x)$, denoted $\|f(x)\|$, is defined as the square root of the inner product of $f(x)$ with itself.

for some $A > 0$, $B < \infty$, and all $f(x) \in \mathcal{V}$. Dividing this equation by $\|f(x)\|^2$, we see that A and B “frame” the normalized inner products of the expansion coefficients and the function. If $A = B$, the expansion set $\phi_k(x)$ is called a *tight frame* and it can be shown that [11]

$$f(x) = \frac{1}{A} \sum_k \langle \phi_k(x), f(x) \rangle \phi_k(x). \quad (2.7)$$

2.1.2 Discrete Wavelet Bases

We have given the general definition of basis functions $\phi_k(x)$ of the space \mathcal{V} in the previous subsection. Now consider the set of expansion functions composed of integer translations and binary scalings of the real, square-integrable function $\phi(x)$; that is, the set $\{\phi_{j,k}(x)\}$ where

$$\phi_{j,k}(x) = 2^{j/2} \phi(2^j x - k) \quad (2.8)$$

for all $j, k \in \mathcal{Z}$ and $\phi(x) \in L^2(\mathcal{R})$ ⁴. Here, k determines the position of $\phi_{j,k}(x)$ along the x -axis, and j determines $\phi_{j,k}(x)$'s width along the x -axis. $\phi(x)$ is normally called a *scaling function*. Let \mathcal{V}_j denote the span of $\phi_{j,k}$ over $k \in \mathcal{Z}$, i.e.:

$$\mathcal{V}_j = \underset{k}{Span}\{\phi_{j,k}(x)\}. \quad (2.9)$$

This means that if $f(x) \in \mathcal{V}_j$, then it can be expressed as

$$f(x) = \sum_{k \in \mathcal{Z}} \alpha_k \phi(2^j x - k). \quad (2.10)$$

For $j > 0$, the span can be larger since $\phi_{j,k}(x)$ is narrower and is translated in smaller steps. It can, therefore, represent finer details. For $j < 0$, $\phi_{j,k}(x)$ is wider and is translated

⁴The notation $L^2(\mathcal{R})$ denotes the set of measurable, square-integrable 1-D functions

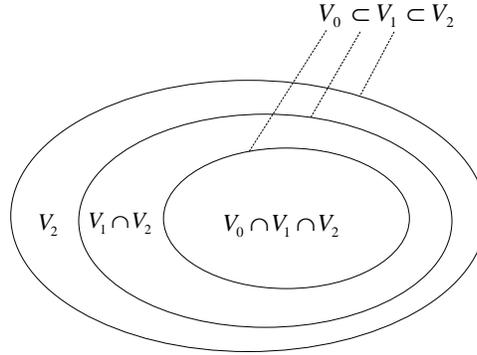


Figure 2.2: Nested spaces spanned by scaling functions.

in larger steps. So these wider scaling functions can represent only coarse information, and the space they span is smaller.

In order to properly define the wavelet function, Mallat [3] formulated the basic requirement of the scaling functions $\phi_{j,k}(x)$:

The subspaces spanned by the scaling function at lower scales are nested within those spanned at higher scales. i.e.,

$$\mathcal{V}_{-\infty} \subset \cdots \subset \mathcal{V}_{-1} \subset \mathcal{V}_0 \subset \mathcal{V}_1 \subset \mathcal{V}_2 \subset \cdots \subset \mathcal{V}_{+\infty}. \quad (2.11)$$

This is called the multiresolution analysis (MRA) requirement. The relationship of the spanned spaces is illustrated in Fig. 2.2. Given a scaling function that meets the MRA requirement, we can define a *wavelet function* $\psi(x)$ which, together with its integer translates and binary scalings, spans the difference (i.e., the orthogonal complement) between any two adjacent scaling subspaces, \mathcal{V}_j and \mathcal{V}_{j+1} , a relationship which is illustrated schematically in Fig. 2.3. The space \mathcal{W}_j is spanned by $\psi_{j,k}(x)$ over $k \in \mathcal{Z}$, where $\psi_{j,k}(x)$ is defined similarly to $\phi_{j,k}(x)$ as

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k) \quad (2.12)$$

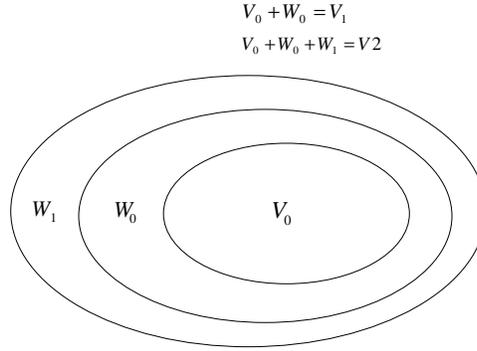


Figure 2.3: The relationship of spaces spanned by the scaling and wavelet functions. \mathcal{W}_j is the difference between \mathcal{V}_{j+1} and \mathcal{V}_j .

From Fig. 2.2 and Fig. 2.3, we can see that

$$\mathcal{V}_0 \subset \mathcal{V}_1$$

$$\mathcal{W}_0 \subset \mathcal{V}_1$$

These relationships lead to two well-known equations of MRA [3]:

$$\begin{aligned} \phi(x) &= \sum_n h_0(n) \sqrt{2} \phi(2x - n) \\ \psi(x) &= \sum_n h_1(n) \sqrt{2} \phi(2x - n), \end{aligned} \quad (2.13)$$

for $n \in \mathcal{Z}$. $h_0(n)$ and $h_1(n)$ are called the scaling filter and wavelet filter, respectively.

Daubechies [11] proved that by choosing $\phi(x)$ wisely, $\{\phi_{j,k}(x)\}$ can be made to span $\mathcal{L}^2(\mathcal{R})$, i.e.,

$$\mathcal{L}^2 = \mathcal{V}_{j_0} \oplus \mathcal{W}_{j_0} \oplus \mathcal{W}_{j_0+1} \oplus \cdots$$

Thus, for any $f(x) \in \mathcal{L}^2(\mathcal{R})$ we have a generalized form of (2.1):

$$f(x) = \sum_k a_{j_0}(k) 2^{j_0/2} \phi(2^{j_0} x - k) + \sum_k \sum_{j=j_0}^{\infty} d_j(k) 2^{j/2} \psi(2^j x - k)$$

or

$$f(x) = \sum_k a_{j_0}(k) \phi_{j_0,k}(k) + \sum_k \sum_{j=j_0}^{\infty} d_j(k) \psi_{j,k}(x) \quad (2.14)$$

where j_0 could be any integer, setting the coarsest scale whose space is spanned by $\phi_{j_0,k}(x)$. The rest of $\mathcal{L}^2(\mathcal{R})$ is spanned by the wavelets which provide the high resolution details of the signal.

$a_{j_0}(k)$ and $d_j(k)$ in (2.14) are called coefficients of the *discrete wavelet transform* (DWT) of signal $f(x)$. If the wavelet system is orthonormal, these coefficients can be calculated by

$$a_j(k) = \langle f(x), \phi_{j,k}(x) \rangle$$

and

$$d_j(k) = \langle f(x), \psi_{j,k}(x) \rangle$$

From (2.13) we can get the formula for the fast DWT:

$$\begin{aligned} a_j(k) &= \sum_m h_0(m - 2k) a_{j+1}(m) \\ d_j(k) &= \sum_m h_1(m - 2k) a_{j+1}(m). \end{aligned} \quad (2.15)$$

That is, the wavelet coefficients can be calculated recursively from scale to scale, which yields the complexity of $O(N)$, where N is the length of the data to be transformed. For comparison, the fast DWT algorithm is even faster than the widely-used fast Fourier transform (FFT) at $O(N \ln N)$. Based on (2.15), the DWT can be implemented by filter banks as shown in Fig. 2.4.

There are many important properties of the discrete wavelet transform [3],[1]. However, from the application point of view two of them should be emphasized:

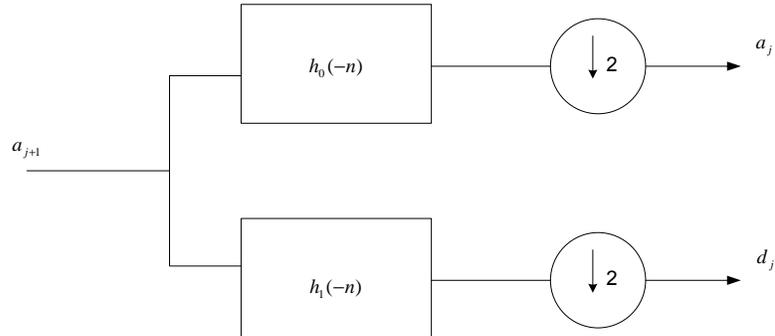


Figure 2.4: The DWT can be implemented using filter banks followed by down-sampling.

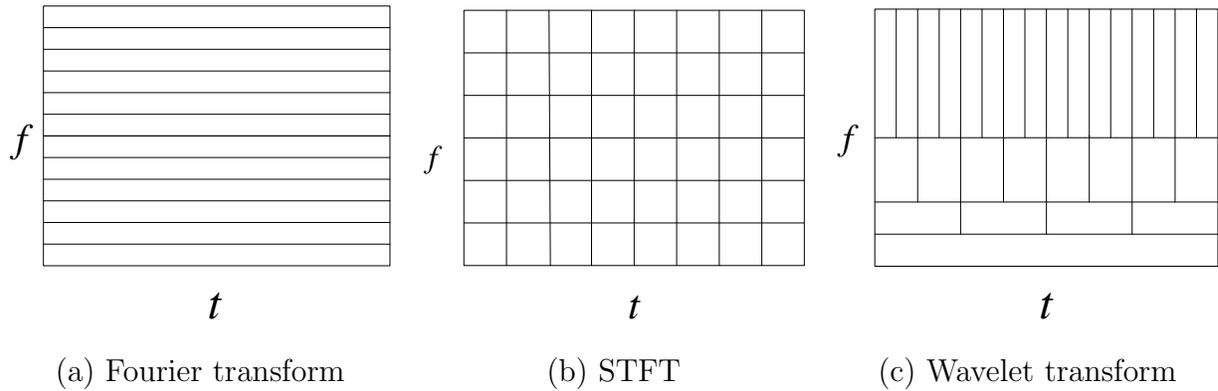


Figure 2.5: Time-frequency tiles of the Fourier transform, the STFT [1] and the wavelet transform. (a) has no time-localization; (b) only has fixed time-frequency resolution; (c) has different resolutions for different frequency bands. In the high-frequency bands, the wavelet transform has lower frequency resolution but higher time resolution. This is often desired in signal analysis.

Adaptive time-frequency analysis: Adaptive time-frequency analysis is critical for non-stationary signal analysis due to time-varying frequency components, whereas the Fourier transform is an ideal analysis tool for stationary signals. Unfortunately, the Fourier transform is ill-suited for nonstationary signals because of its poor time-localization. The classical solution to this problem is to use the short-time (or windowed) Fourier transform (STFT) [1]. However, the STFT has several problems, the most severe being the *fixed* time-frequency resolution of the basis functions. Wavelet techniques give a new class of bases that have desired time-frequency resolution properties. Furthermore, since wavelet bases constitute a very large library of basis functions, one may choose the optimal bases from a library for different signals, thus making the analysis signal-dependent and adaptive. Fig. 2.5 illustrates the time-frequency tiles for the Fourier transform, STFT and the wavelet transform.

Vanishing moments: If the wavelet basis functions $\psi(x)$ satisfy the condition:

$$\int_{-\infty}^{\infty} x^k \psi(x) dx = 0, \text{ for } k = 1, 2, \dots, M \quad (2.16)$$

then $\psi(x)$ is said to have M vanishing moments. Daubechies [11] proved that the vanishing moment determines the smoothness of $\psi(x)$. And the famous Daubechies wavelets are designed to maximize the vanishing moments under the constraint of the support length of basis functions. For the fractal Brownian motion process, Tewfik *et.al.* [12] showed that the higher the vanishing moment of $\psi(x)$, the less the wavelet coefficients are correlated. The role of the vanishing moments in image compression was studied by [13]. In practice, researchers often choose wavelets with relatively large vanishing moments. However, there is a penalty of losing locality since large vanishing moments usually mean a large basis support.

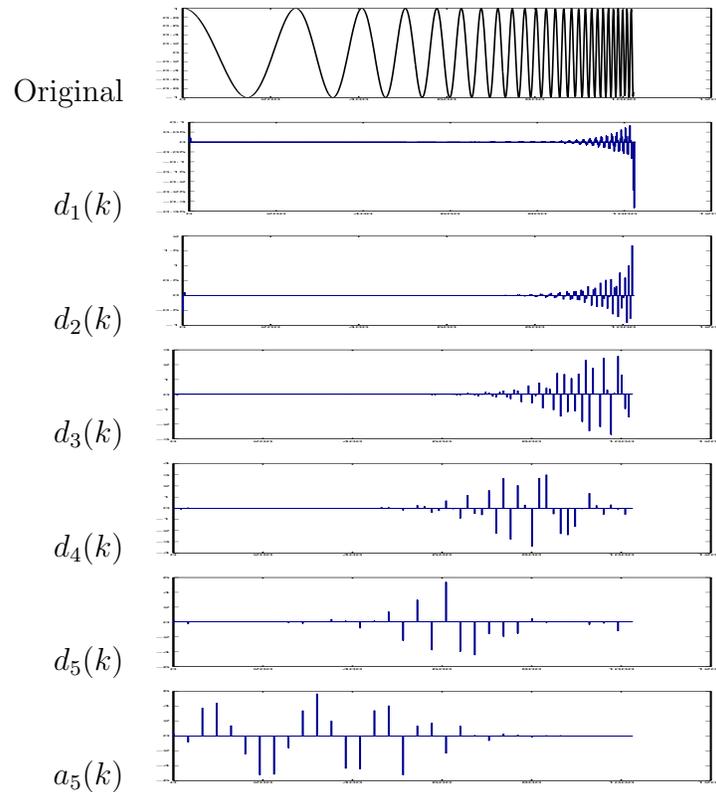


Figure 2.6: Discrete wavelet transform of a chirp signal. Notice how the location in k tracks the frequencies in the signal in a way that the Fourier transform cannot. This suggests that the wavelet transform is well-suited for time-frequency analysis.

Fig. 2.6 shows the wavelet coefficients of a Chirp signal, which has a time-varying frequency. Notice how the location in k tracks the frequencies in the signal in a way that the Fourier transform cannot.

So far, our discussion has been limited to the one dimensional DWT. To construct multi-dimensional wavelet transforms, the common approach is to use the tensor product [3] of 1-D wavelet basis functions, mainly because of simplicity. Other sophisticated methods [14] [15] have also been developed to design (non-separable) multi-dimensional wavelets, but with greater implementation complexity. A more detailed discussion of the 2-D wavelet transform is given in Chapter 3.

2.1.3 Wavelet-based Statistical Models

In the above subsection we introduced the mathematical background of the wavelet transform from the deterministic functional analysis point of view. Since many practical problems can only be specified by statistical models, in this subsection we briefly review several important wavelet-based statistical models proposed in literatures.

A. Independence Model

Due to its simplicity and effectiveness, the independence model [9],[16],[17] was widely used for image denoising and compression, among other applications. It is well-known now that the wavelet coefficients of natural signals obey *non-Gaussian* distributions. The most common statistical model for each coefficient is the generalized Gaussian distribution (GGD) [9]:

$$p_x(x|\sigma_x, \beta) = C(\sigma_x, \beta) \exp\{-[\alpha(\sigma_x, \beta)|x|]^\beta\}, \quad (2.17)$$

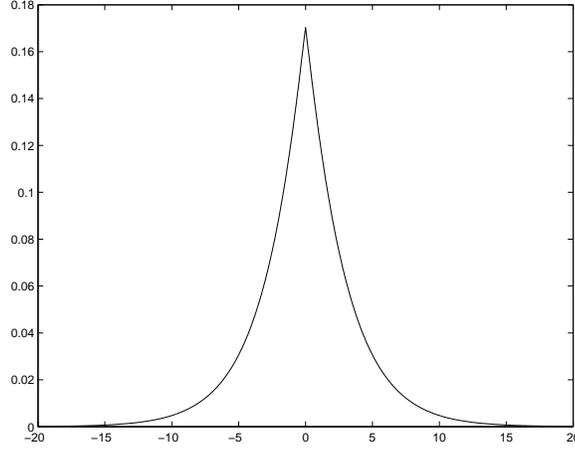


Figure 2.7: An example generalized Gaussian distribution function. This class of distributions are dominated by small coefficients (in magnitude) with long tails.

where $-\infty < x < \infty$, $\sigma_x > 0$ and $\beta > 0$. Also,

$$\alpha(\sigma_x, \beta) = \sigma_x^{-1} \left[\frac{\Gamma(3/\beta)}{\Gamma(1/\beta)} \right]^{1/2}$$

and

$$C(\sigma_x, \beta) = \frac{\beta \cdot \alpha(\sigma_x, \beta)}{2\Gamma(1/\beta)}$$

where $\Gamma(\cdot)$ is the Gamma function. A plot of the GGD $p_x(x|\sigma_x, \beta)$ with $\sigma_x = 3.87$, $\beta = 1.06$ is shown in Fig. 2.7. This class of distributions is dominated by small wavelet coefficients x (in magnitude) with long tails (few large coefficients).

B. Gaussian Scale Mixture (GSM) Model

The GSM model was proposed about 30 years ago [18]. The GGD model discussed above is a special case of GSM models. Recently, Portilla *et.al.* [19] developed the GSM for modeling wavelet coefficients. They used this model for wavelet-based image denoising

and obtained impressive results. Formally, a random vector \mathbf{x} is a Gaussian scale mixture if and only if it can be expressed as the product of a zero-mean Gaussian vector \mathbf{u} and an independent positive scalar random variable \sqrt{z}

$$\mathbf{x} \stackrel{d}{=} \sqrt{z}\mathbf{u} \quad (2.18)$$

where $\stackrel{d}{=}$ indicates equality in distribution. The variable z is known as the *multiplier*. The vector \mathbf{x} is thus an infinite mixture of Gaussian vectors, whose density is determined by the covariance matrix $C_{\mathbf{u}}$ of vector \mathbf{u} and the mixing density, $p_z(\mathbf{z})$

$$\begin{aligned} p_{\mathbf{x}}(\mathbf{x}) &= \int_{-\infty}^{\infty} p(\mathbf{x}|z)p_z(z)dz \\ &= \int_{-\infty}^{\infty} \frac{\exp(-\mathbf{x}^T(zC_{\mathbf{u}})^{-1}\mathbf{x}/2)}{(2\pi)^{N/2}|zC_{\mathbf{u}}|^{1/2}} p_z(z)dz \end{aligned} \quad (2.19)$$

where N is the size of \mathbf{x} and \mathbf{u} . From (2.19) we see that given $C_{\mathbf{u}}$ the GSM distribution function $p_{\mathbf{x}}(\mathbf{x})$ is determined by the a prior distribution $p_z(z)$. For the wavelet coefficients of natural images, Portilla *et.al.* empirically proposed to use the GGD model (2.17) for $p_z(z)$.

C. Markov Models

In the wavelet domain, two most influential Markov models are Markov random fields (MRF) [20],[21] and hidden Markov trees (HMT) [22],[23].

The MRF models intend to exploit the clustering property of the wavelet coefficients. Noticing that large/small (in magnitude) wavelet coefficients tend to stay clustered, Malfait *et.al.* [20] introduced a binary label field \mathbf{l} in representing the wavelet coefficient vector \mathbf{x} :

$$p_x(\mathbf{x}) = \sum_{\mathbf{l}} p(\mathbf{x}|\mathbf{l})p_l(\mathbf{l}) \quad (2.20)$$

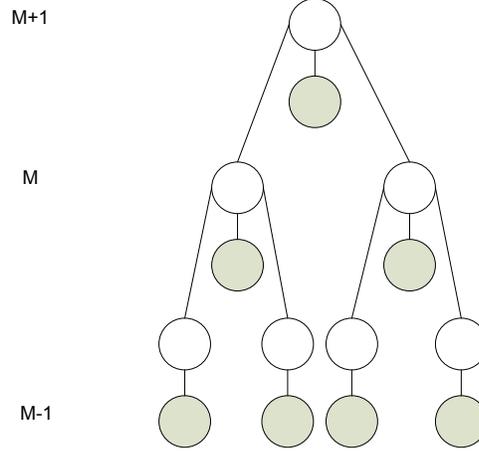


Figure 2.8: HMT structure. Each black node represents a continuous wavelet coefficient x_i . Each white node represents the hidden state variable s_i for κ_i . Note that all of the states on the tree constitute a Markov chain.

where the conditional probability $p(\mathbf{x}|\mathbf{l})$ assumes an independence model, i.e., given the label field \mathbf{l} , the wavelet coefficients \mathbf{x} are independent of each other. An auto-logistic Markov model was used for $p_l(\mathbf{l})$. The details of the auto-logistic Markov model, including neighborhood structure and definition of clique potentials, are given in [20],[21].

Besides the clustering property, the wavelet coefficients have another important property: persistence [22]. That is, large/small values of wavelet coefficients tend to propagate across scales. The hidden Markov tree model (HMT) proposed firstly by Crouse *et.al* [22] successfully captured the persistence property of wavelet coefficients. To do this, they first group wavelet coefficients using a tree structure (see Fig. 2.8). Then they introduce a hidden state variable for each wavelet coefficient. For example, for the i th coefficient on level m , x_i^m , the hidden state variable is s_i^m . We thus have

$$p_x(x_i^m) = \sum_{s_i^m} p(x_i^m | s_i^m) p_s(s_i^m), \quad (2.21)$$

where $p(x_i^m | s_i^m)$ is a Gaussian distribution function. $p_s(s_i^m)$ is determined by assuming all the state variables on a tree constitute a Markov chain.

2.2 Bayesian Estimation

In this section, we briefly review Bayesian estimation theory. This will be used in the following chapters of image and video denoising.

2.2.1 Static Estimation

The static estimation problem can be formally defined as:

Given the measurement \mathbf{y} of a random vector \mathbf{x} , we want to estimate \mathbf{x} in terms of \mathbf{y} by minimizing some cost function $E\{C(\tilde{\mathbf{x}})\}$, where

$$\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}(\mathbf{y})$$

is the estimation error. $\hat{\mathbf{x}}(\mathbf{y})$ is the estimation function.

There are various estimation criteria based on the selection of the cost function [24]:

Mean-square estimation: If we define

$$C(\tilde{\mathbf{x}}) = \|\tilde{\mathbf{x}}\|^2,$$

the resulting estimate $\hat{\mathbf{x}}(\mathbf{y})$ is called minimum mean-square estimation (MMSE).

Maximum a posterior estimation (MAP) Define

$$C(\tilde{\mathbf{x}}) = \begin{cases} 1 & |\mathbf{x} - \hat{\mathbf{x}}| \geq \epsilon \\ 0 & |\mathbf{x} - \hat{\mathbf{x}}| < \epsilon \end{cases} \quad (2.22)$$

Observe that the MAP criterion treats all the errors above a given value ϵ equally by a cost function that is equal to one. The above cost function corresponds to selecting an estimate $\hat{\mathbf{x}}(\mathbf{y})$ that maximizes the a posteriori probability $p(\mathbf{x}|\mathbf{y})$.

It should be noted that if \mathbf{x} and \mathbf{y} are jointly Gaussian distributed then the MMSE and MAP estimates are equivalent.

If we have the measurement equation

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \quad (2.23)$$

and want to estimate \mathbf{x} , then this is a denoising problem of additive noise \mathbf{v} . If we further assume that \mathbf{x} and \mathbf{v} are both Gaussian random vectors, then \mathbf{x} and \mathbf{y} are jointly Gaussian, thus the MMSE and MAP estimates are equivalent.

In (2.23), if

$$\begin{aligned} \mathbf{x} &\sim N(\mu_{\mathbf{x}}, P_x) \\ \mathbf{v} &\sim N(0, P_v) \end{aligned}$$

where $N(\cdot)$ denotes the Gaussian distribution (P_x and P_v are covariance matrices of \mathbf{x} and \mathbf{v} , respectively), and \mathbf{x} and \mathbf{v} are independent, then the MMSE estimate of \mathbf{x} is

$$\hat{\mathbf{x}}(y) = \mu_{\mathbf{x}} + P_x(P_x + P_v)^{-1}(\mathbf{y} - \mu_{\mathbf{x}}) \quad (2.24)$$

and the associated error covariance matrix is

$$E(\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T) = \tilde{P} = P_x - P_x(P_x + P_v)^{-1}P_x \quad (2.25)$$

2.2.2 Dynamic Estimation

1. Concept

In the static estimation problem (2.24, (2.25) we need to compute $(P_x + P_v)^{-1}$. This is often a very difficult task if the size N of the vector \mathbf{x} is large, which is especially the case in image and video processing. For example, for a moderate size $N = 256 \times 256$ image, P_x and P_v are huge matrices of the size $256^2 \times 256^2$. Any straightforward approach to computing $(P_x + P_v)^{-1}$ will be prohibitive. However, if \mathbf{x} evolves with time/space according to some dynamic rule then the much more efficient dynamic estimation method can be applied for optimal estimation.

Assume that \mathbf{x} obeys the following dynamic equation (state equation)

$$\mathbf{x}(n+1) = A\mathbf{x}(n) + B\mathbf{w}(n+1), \quad n = 1, 2, \dots, N \quad (2.26)$$

where

A : A matrix which relates the state at time n to the state at time $n+1$. A describes the deterministic part of the underlying process dynamics.

B : A matrix that contains the stochastic dynamics of the underlying system.

$\mathbf{w}(n+1)$: A vector that represents the driving noise. This is assumed to be a white sequence. Note $\mathbf{w}(n+1)$ is independent of $\mathbf{x}(m)$, $m = 1, 2, \dots, n$.

If the corresponding measurement equation is

$$\mathbf{y}(n) = \mathbf{x}(n) + \mathbf{v}(n), \quad n = 1, 2, \dots, N \quad (2.27)$$

then the estimate $\hat{\mathbf{x}}(\mathbf{y})$ can be computed *recursively* by Kalman filtering [24] and Rauch-Tung-Striebel (RTS) smoothing [25]. This is a very efficient algorithm. For example, assume we have N vectors $\mathbf{x}(n)$, $n = 1, 2, \dots, N$, each of size M . Then the complexity of

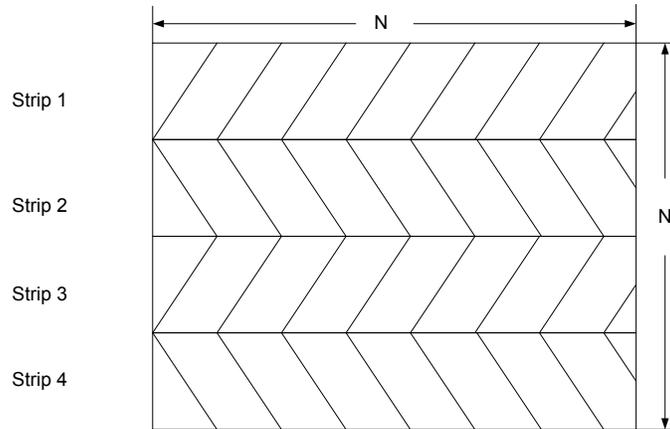


Figure 2.9: An example of strip structures. The big box illustrate an image of size $N \times N$. Each shaded rectangle is a strip. We divide this image into four strips. For clarity the strips shown are non-overlapping. However, they can be overlapped in any forms.

dynamic estimation is $O(NM^3)$, whereas the static estimation has complexity $O((MN)^3)$.

2. Strip Kalman Filtering

Although the above Kalman filtering process is very efficient, the complexity of the optimal estimator is often still too high for large-scale problems, such as image and video denoising problems. To further reduce complexity, Woods *et.al.* [26] proposed to perform *approximate* strip Kalman filtering, instead of *optimal* Kalman filtering. The goal is to significantly reduce complexity with small increase of estimation error $\tilde{\mathbf{x}}$. In Chapter 5 we will design and compare the spatial and wavelet-based strip Kalman filters. Here, we give a brief introduction to this type of filter.

The strip Kalman filtering aims towards reducing the dimensionality of the state vector $\mathbf{x}(n)$, $n = 1, 2, \dots, N$ by decoupling the original dynamic system into smaller subsystems that are processed independently. For example, in image denoising if the state $\mathbf{x}(n)$ rep-

resents one column, we can divide each column into several segments (see Fig. 2.9) and process each strip independently. Although in Fig. 2.9, the strips are not overlapped, in principle, they can be highly overlapped of any forms. Strip design is a constrained optimization problem, which will be discussed in Chapter 5. Mathematically, if we divide $\mathbf{x}(n)$ into two parts (i.e., $\mathbf{x}(n) = [\mathbf{x}_1(n), \mathbf{x}_2(n)]^T$), for example, then the corresponding state and measurement equations are, respectively

$$\begin{bmatrix} \mathbf{x}_1(n+1) \\ \mathbf{x}_2(n+1) \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1(n) \\ \mathbf{x}_2(n) \end{bmatrix} + \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \begin{bmatrix} \mathbf{w}_1(n+1) \\ \mathbf{w}_2(n+1) \end{bmatrix} \quad (2.28)$$

and

$$\begin{bmatrix} \mathbf{y}_1(n+1) \\ \mathbf{y}_2(n+1) \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1(n+1) \\ \mathbf{x}_2(n+1) \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1(n+1) \\ \mathbf{v}_2(n+1) \end{bmatrix} \quad (2.29)$$

If $\mathbf{x}_1(n)$ and $\mathbf{x}_2(n)$ have lengths N_1 and N_2 , respectively (i.e., $N_1 + N_2 = N$), then the computational complexity for the strip Kalman filter is $O(N_1^3 + N_2^3)$ instead of $O((N_1 + N_2)^3)$.

It should be noted that the strip Kalman filter is an approximate filter. Its performance depends heavily on the complexity constraint and correlation structure of the original problem. We will show in Chapter 5 that performing the strip Kalman filtering in the wavelet domain has considerable advantages over spatial strip Kalman filtering.

Chapter 3

Complex Wavelet Transforms

Chapter 2 discussed the real-valued wavelet transform which, although very widely-used, is subject to a number of shortcomings: (1) lack of symmetry, (2) sensitive to shifts, and (3) poor directionality with high-dimensional wavelet transforms. In this chapter we discuss in detail a new class of wavelet transform: the complex wavelet transform [27],[28]. Due to its desirable properties, *orthogonality and symmetry, near shift-invariance and good directionality*, complex wavelets are well suited for use in image and video applications.

The first generation of the complex wavelet transform is normally called the single-tree complex wavelet [29]. It has a history dating back to the early 1990s, when several researchers tried to address drawbacks (e.g., shift-variance and poor direction-selectivity) of the real-valued wavelet transforms [30],[31]. However, research and interest in the single-tree complex wavelet transform has not gained much momentum mainly because the associated complex wavelet filters do not present good frequency selectivity. In 1998 Kingsbury [32], [33] proposed the so-called dual-tree complex wavelet transform to achieve a multiresolution analysis with approximate shift-invariance, good directionality and good frequency

frequency selectivity.

Section 3.1 begins the discussion by summarizing problems with the real-valued wavelet transform which hampers its application in practice. Then Section 3.2 will develop the dual-tree complex wavelet and its properties. The material of this chapter will be used in Chapters 6 and 7 for image and video denoising.

3.1 Problems with Real-valued Wavelet Transforms

The present standard wavelet transforms are all *real-valued* transforms, such as orthogonal and biorthogonal wavelets [11],[34],[35]. They have many interesting properties (e.g., local space/frequency analysis, energy compaction and signal decorrelation) and have successfully been used in many image processing applications, such as compression [10], denoising/restoration [9], [20], [22], and feature extraction [36]. However, under the constraint of being real-valued they suffer from a few problems which hamper their application.

Asymmetry:

Except for the Haar wavelet, the *real-valued orthogonal* wavelet filter can not be symmetric (i.e., odd or even). However, symmetry is expected for many applications (e.g., boundary artifact suppression in subband coding [30],[27]). This non-symmetry property of the real-valued orthogonal can be explained as follows.

Let $h(n)$, $n = 0, 2, \dots, N - 1$ be an *orthogonal* finite impulse response (FIR) filter that can be used to construct the scaling function $\phi(x)$, i.e.

$$\phi(x) = \sqrt{2} \sum_{k=0}^{N-1} h(k) \phi(2x - k) \quad \text{and} \quad \int_{-\infty}^{+\infty} \phi(x) dx = 1. \quad (3.1)$$

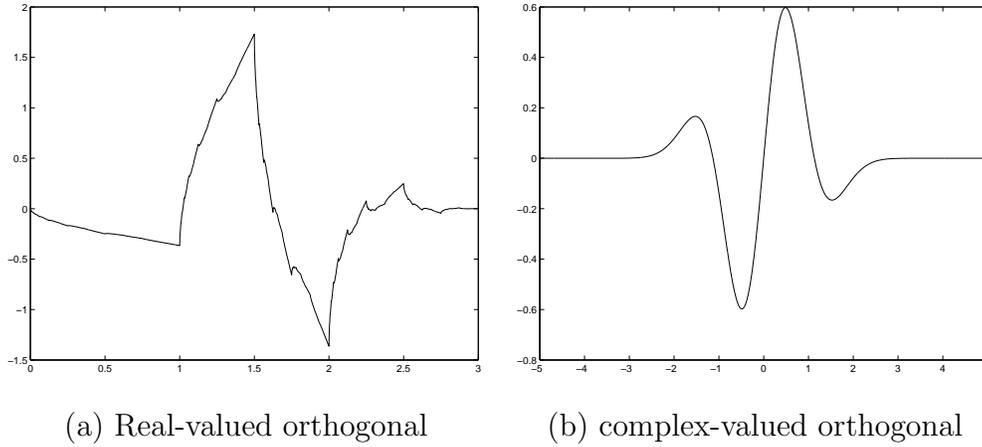


Figure 3.1: Wavelet functions of a real-valued orthogonal basis and the real component of a complex-valued orthogonal basis. Note the clear, desirable symmetry of the complex case, which is absent in the real-valued function.

In (3.1), if $\phi(2x - k)$, $k \in \mathbb{Z}^1$ are orthogonal functions, then $h(n)$ should satisfy two conditions [11]:

$$\sum_k h(k)h^*(k + 2n) = \delta(n) \quad (\text{orthogonality}) \quad (3.2)$$

and

$$\sum_n h(n) = \sqrt{2} \quad (3.3)$$

where $\delta(n)$ denotes the unit impulse sequence and h^* the complex conjugate of h . Now suppose we further limit $h(n)$ to be *symmetric* and *real-valued*. Daubechies proved such a $h(n)$ can only be $[\sqrt{2}/2, \sqrt{2}/2]$, that is, the Haar wavelet [11]. For clarity, we show why this is true for $N = 4$ as follows.

¹ \mathbb{Z} is the set of integers

In (3.2), if we set $n = 1$ then we have

$$h(0)h(2) + h(1)h(3) = 0. \quad (3.4)$$

Suppose that $\{h(n), n = 1, 2, 3, 4\}$ is real and evenly symmetric. From (3.3), if $h(n)$ is symmetric then it must be even, thus $h(0) = h(3)$, $h(1) = h(2)$. Then (3.4) reduces to

$$2h(0) \cdot h(1) = 0. \quad (3.5)$$

Thus $h(0)$ or $h(1)$ has to be zero, but not both because of (3.3). If we set $h(1) = 0$ (by symmetry $h(2) = 0$), then from (3.3) we get

$$h(0) + h(3) = \sqrt{2}. \quad (3.6)$$

i.e., $h(0) = h(3) = \sqrt{2}/2$, which is the simple Haar wavelet filter. In (3.5) if we assume $h(0) = 0$, we will reach the same conclusion. To summarize, we have shown that if $\{h(n), n = 1, 2, 3, 4\}$ is real, orthogonal and symmetric then it must be the Haar wavelet filter. However, if $h(n)$ is allowed to take *complex* values then the symmetry conditions are more flexible. First, (3.4) is written as

$$h(0)h^*(2) + h(1)h^*(3) = 0, \quad (3.7)$$

which, under symmetry, is equivalent to

$$h(0)h^*(1) + (h(0)h^*(1))^* = 0 \quad (3.8)$$

which just requires the *real* part of $h(0)h^*(1)$ to be 0. Thus, many complex orthogonal symmetric wavelet filters (besides the Haar wavelet filter) can be designed [30]. Table 3.1 shows a symmetric 6-tap complex orthogonal wavelet filter; For comparison, Table 3.1 also

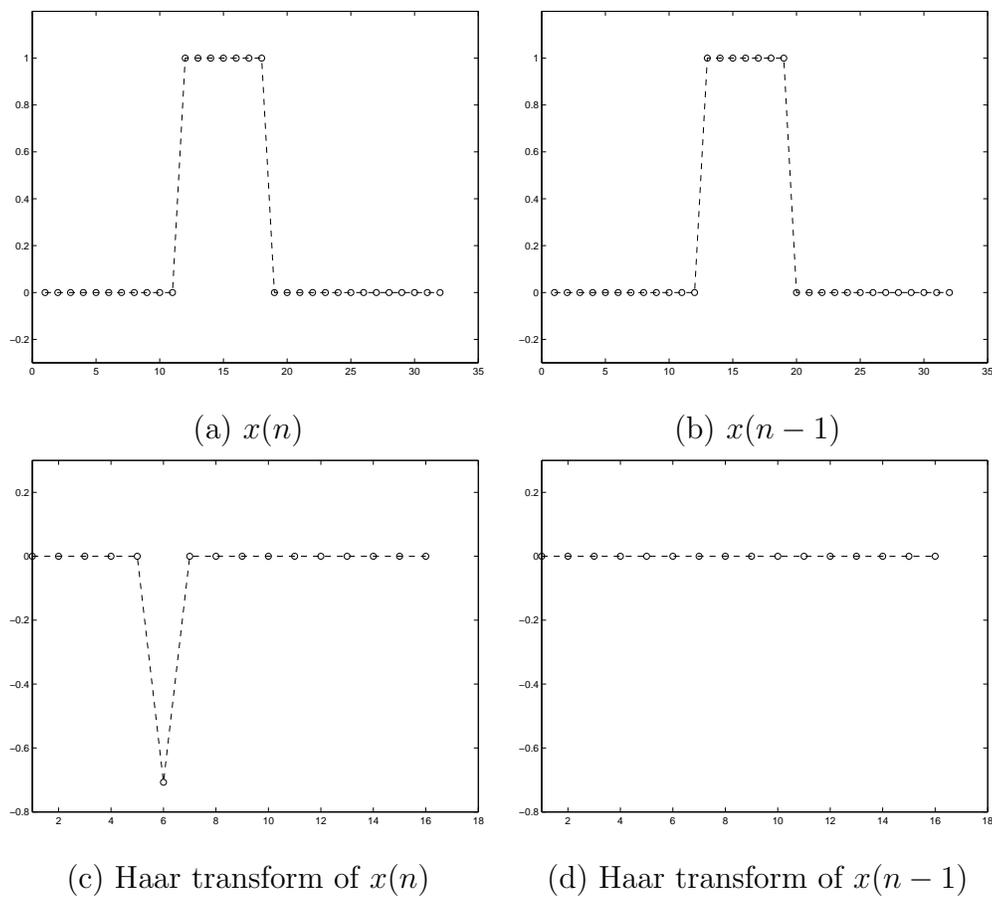


Figure 3.2: Shift-variance of the Haar wavelet transform: The wavelet coefficients (c) of the original signal $x(n)$ are quite different from those of a shifted version of $x(n)$ (d).

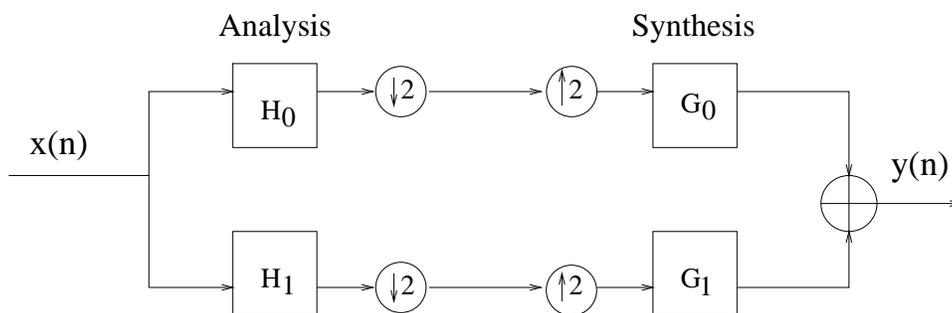
Wavelet Family	Filter Coefficients
Haar	$\sqrt{2}/2, \sqrt{2}/2$
Daubechies Length 6	0.33, 0.81, 0.46, -0.14, -0.09, 0.04
Complex Length 6	$(-0.07, -0.11, 0.66, 0.66, -0.11, -0.07) +$ $j(-0.09, -0.09, 0.17, 0.17, -0.09, -0.09)$

Table 3.1: Examples of orthogonal real and complex wavelet filters. Note that the Daubechies is non-symmetric, whereas the complex filter is.

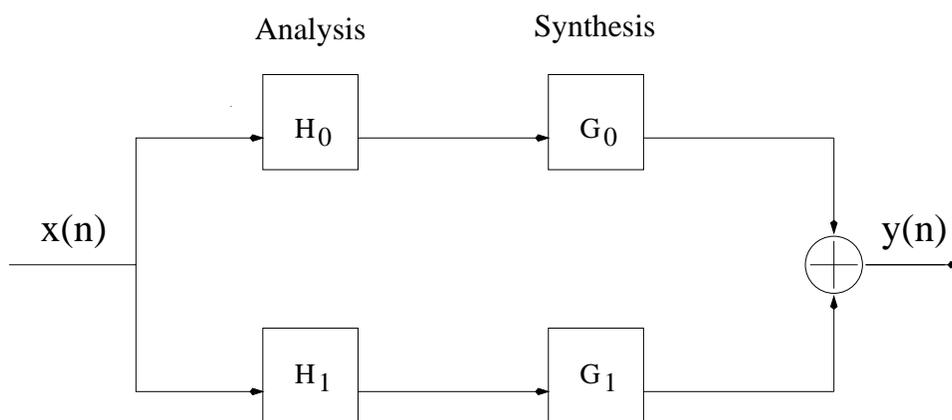
shows the Daubechies-6 wavelet filter, which is real-valued and orthogonal but not symmetrical. The corresponding wavelet functions are shown in Fig. 3.1.

Shift Sensitivity

The compact real-valued wavelets (including both orthogonal and non-orthogonal wavelets) are sensitive to signal shift under critical decimation. This phenomenon is usually called shift-variance. A formal definition of shift-variance was proposed by Simoncelli [37]: *shift-variance means small shifts in the input signal can cause major variations in the distribution of energy between coefficients at different scales.* Fig. 3.2 shows the shift-variance property of the Haar wavelet. Note the large energy change due to a single sample shift. Shift-invariance has been shown by many researchers (e.g., [9], [20], [38], [21]) to be very useful for suppressing artifacts in image denoising. Simoncelli *et al.* [37] relate shift-invariance of energy distribution to interpolability. It should be noted that shift-variance comes from the frequency-aliasing due to subsampling. Wavelet filters with better frequency selectivity generally have smaller shift-variance. But better frequency



(a) Decimated wavelet transform



(b) Undecimated wavelet transform

Figure 3.3: Decimated and undecimated wavelet transforms. Note in (a), there are down-sampling and up-sampling units which are absent in (b).

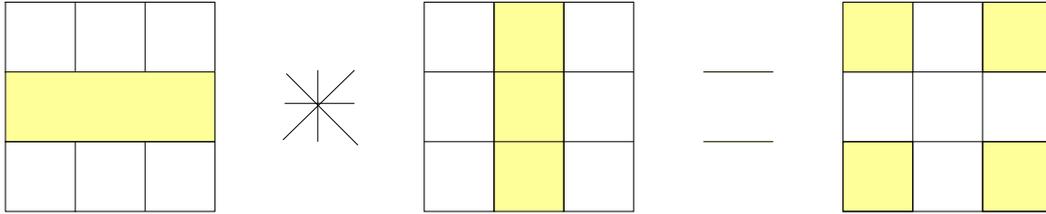


Figure 3.4: Idealized spectrum of the tensor product of two real-valued 1D wavelets.

selectivity always means longer support length for *real-valued* filters, which causes a complexity penalty [13]. Filters of long length also risk the loss of locality, which is, however, one of the expected properties of the wavelet transform. Another way of providing shift-invariance is to use the undecimated form of the dyadic filter tree (Fig. 3.3) [39]. However, undecimation increases the redundancy¹ of the wavelet transform and thus the computational complexity. We will see how the complex wavelet transform can handle these problems.

Directional Selectivity

The most convenient way to implement 2-D and higher dimensional wavelet transforms is by the tensor products of several 1-D wavelets. This is normally called the separable implementation of multi-dimensional wavelets. However, when we use the tensor product of two *real-valued* 1-D wavelet bases to implement 2-D wavelet transforms the resulting wavelets suffer from a poor directional selectivity for diagonal features. The easiest way to understand the poor directionality that arises in the separable implementation of the

¹In this thesis *redundancy* of a transform is defined as the ratio of the number of the outputs to the number of inputs. A complex coefficient is counted as two outputs.

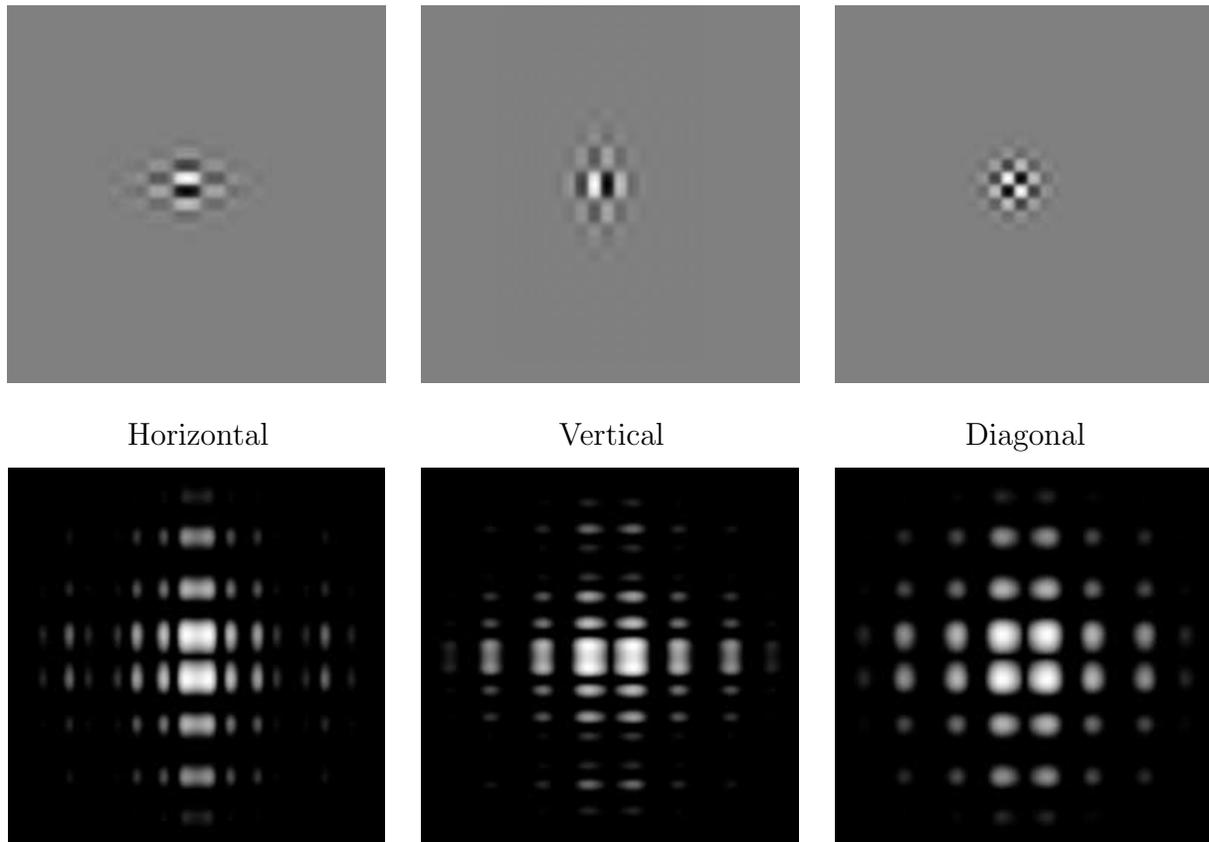


Figure 3.5: An example of the poor directional selectivity of the real-valued $2D$ separable discrete wavelet transform. Top row illustrates the wavelets in the spatial domain, the second row illustrates the power spectrum of each wavelet in the 2-D frequency plane. The checkerboard artifact of the third wavelet is apparent in the frequency domain as well as the spatial domain.

2-D wavelet transform is in the frequency domain. If $\psi(x)$ is a real-valued 1-D wavelet and the 2-D separable wavelet is given by $\psi(x, y) = \psi(x) \cdot \psi(y)$, then the spectrum of $\psi(x, y)$ is the convolution of the spectrum of $\psi(x)$ and $\psi(y)$. This is illustrated by the idealized diagram in Fig. 3.4. Because $\psi(x)$ is real-valued its spectrum must be two-sided; therefore the spectrum of $\psi(x, y)$ must have support in all four quadrants, which results in poor directionality. As a result, the resulting 2-D wavelet can not distinguish between the 45° and -45° features.

An example of the separable 2-D wavelet bases from real-valued 1-D wavelets is shown in Fig. 3.5. Their Fourier spectra are also shown for clarity. The poor directionality of the third wavelet is apparent.

3.2 Complex Wavelet Transform Properties

In the previous section we showed several problems with *real-valued* wavelet transforms. Now, we introduce the complex wavelet transform (CWT) and discuss its properties. We will show why the complex wavelet transform can handle the problems inherent with the real-valued wavelet transform. In particular, we desire the following properties:

1. Approximate shift-invariance under critical decimation;
2. Good directionality in two dimensions, where the 2-D wavelet is implemented with the simple tensor product of 1-D wavelets. This is also true for higher dimensionality;
3. Limited redundancy (thus a moderate increase of complexity), independent of the number of decomposition levels;
4. Perfect reconstruction.

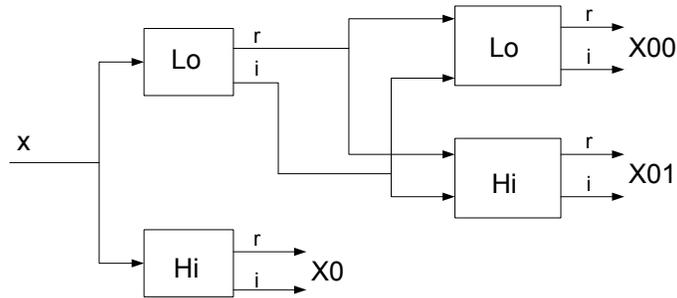


Figure 3.6: Diagram of a two-level decomposition of the single-tree 1-D complex wavelet transform. r and i represent the real and imaginary parts, respectively.

3.2.1 Single-tree and Dual-tree Complex Wavelets

Generalizing the diagram of the standard wavelet transform to accommodate complex numbers leads to the diagram for the 1-D complex wavelet transform, shown in Fig. 3.6. This structure is normally called the single-tree complex wavelet transform. Magarey and Kingsbury [29] used the single-tree complex wavelet for motion estimation and showed that the complex wavelets could provide approximate shift invariance and good directionality, but not perfect reconstruction. A few other authors, such as Lawton [30] and Belzer *et.al* [31], have experimented with complex factorizations of the standard Daubechies polynomials and obtained perfectly reconstructed (PR) complex wavelets. But these do not give filters with good frequency selectivity. To provide both shift invariance and directional selectivity the complex wavelet should be designed to emphasize positive frequencies and reject negative frequencies, or vice-versa [27], the reasons for which are discussed in the following subsection. Unfortunately, the single-tree complex wavelet can not provide perfect reconstruction and good frequency selectivity at the same time. We explain this fact briefly as follows.

We denote a complex wavelet filter as $h(n) = h_r(n) + jh_i(n)$, where $h_r(n)$ and $h_i(n)$ are real-valued functions and $j = \sqrt{-1}$. For $h(n)$ to emphasize positive frequencies and reject negative frequencies we want

$$H(e^{j\omega}) = \begin{cases} \neq 0 & , \quad 0 < \omega < \pi \\ 0 & , \quad -\pi < \omega < 0 \end{cases} \quad (3.9)$$

where $H(e^{j\omega})$ denotes the Fourier transform of $h(n)$. So to have good frequency selectivity the complex wavelet filter $h(n)$ must meet (3.9). However, at the same time the wavelet filter $H(e^{j\omega})$ needs to satisfy the perfect reconstruction condition [34]:

$$H(e^{j\omega})H(e^{-j\omega}) + H(-e^{j\omega})H(-e^{-j\omega}) = 2 \quad (3.10)$$

which means that $H(e^{j\omega})$ must have non-zero values for both positive and negative frequencies, contradicting (3.9). Thus it is impossible to achieve good frequency selectivity and perfect reconstruction simultaneously with the single-tree complex wavelet. A tradeoff has to be made depending on context and application.

To overcome this particular problem with the single-tree complex wavelet, Kingsbury [27] introduced the concept of the dual-tree complex wavelet. The diagram of a two-level decomposition of the dual-tree complex wavelet is given in Fig. 3.7. The upper tree and the lower tree produce the real and imaginary parts of the resulting wavelet coefficients, respectively. The dual-tree complex wavelet has many interesting properties such as approximate shift invariance, good directional selectivity, good frequency selectivity and perfect reconstruction. It should be noted the so-called dual-tree complex wavelet transform proposed by Kingsbury [27] is not a common-sense complex wavelet transform since it actually consists of two wavelet transforms, processing the real and imaginary parts separately (Fig. 3.7). However, since the output is a multiresolution representation of the

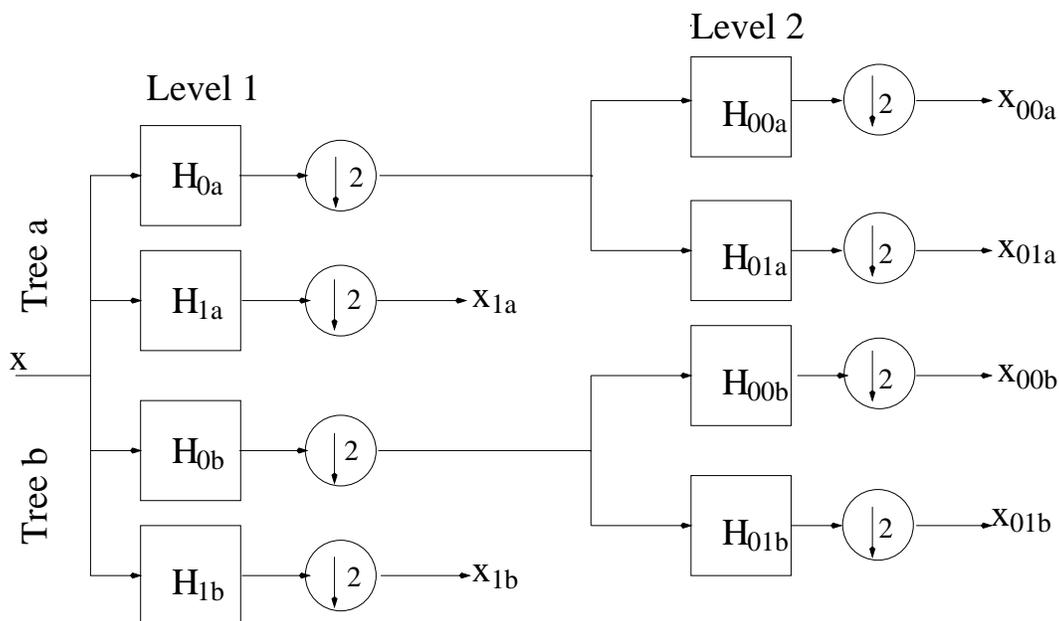


Figure 3.7: Two-level decomposition of the dual-tree complex wavelet transform. Tree *a* produces the real parts of wavelet coefficients while Tree *b* gives the imaginary parts.

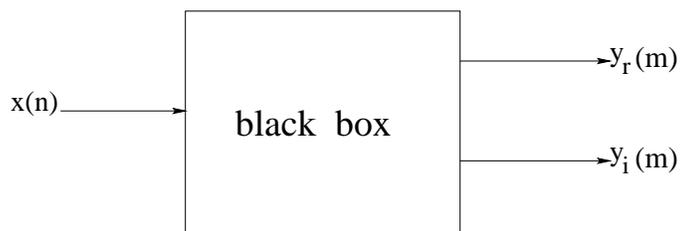


Figure 3.8: Treat dual-tree complex wavelet transform as a black box. $y_r(m)$ and $y_i(m)$ are real and imaginary parts of the wavelet coefficient, respectively.

input, if we focus only on the relationship between the input and output and see the middle part as a black box (Fig. 3.8) then it is still reasonable to treat the system in Fig. 3.7 as a *generalized* complex-valued wavelet transform. We will see later that this viewpoint is important in understanding the shift-invariance and direction-selectivity properties of the dual-tree complex-valued wavelet.

In the next subsection we will show how to design shift-invariant dual-tree complex-valued wavelets because this helps us to understand the properties of this wavelet transform.

3.2.2 Shift-invariant 1-D Dual-tree Complex Wavelet

In [27] Kingsbury showed the basic ideas on how to design the shift-invariant dual-tree complex-valued wavelet. However, his derivation is somewhat heuristic. Later, Selesnick [28] used a different design approach and found an interesting relationship between the wavelet functions of the two trees in the dual-tree complex wavelet transform. We will mainly discuss the approach of Kingsbury because it is easier to understand, and just list Selesnick's results as a theorem. When discussing Kingsbury's approach we will give several examples and some simple additional derivations to clarify his idea. The main reason that we are interested in this design process is that it reveals why the dual-tree complex wavelet is approximately shift-invariant. Before we show the design process it would be helpful to say a few words about why critically-decimated wavelet transforms are generally *shift-variant*.

Fig. 3.9 shows the usual one-level real wavelet decomposition with decimation. The relationship between $X(z)$ and $Y_0(z)$ is

$$Y_0(z) = \frac{1}{2} [H_0(z^{1/2})X(z^{1/2}) + H_0(-z^{1/2})X(-z^{1/2})] \quad (3.11)$$

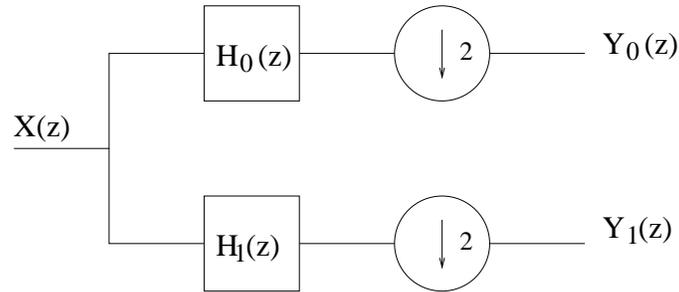


Figure 3.9: One-level wavelet decomposition with decimation. $H_0(z)$ and $H_1(z)$ are low-pass and high-pass filters, respectively.

We can see in (3.11) that downsampling causes aliasing in the frequency domain (note the second term of the right-hand side of (3.11)), which then leads to shift-variance for the critically-decimated wavelet transforms. That is, the shift-variance of the decimated wavelet transform comes from the frequency aliasing due to down-sampling.

To suppress frequency aliasing $h_0(n)$ usually has to be very long to achieve good frequency selectivity (if $h_0(n)$ is real-valued), however this clearly has a complexity penalty. On the other hand, because the spectrum of a complex signal can be just in the positive (or negative) frequency range (thus just half the bandwidth of a real signal) several researchers have developed *complex* $h_0(n)$, which is of short length but is also able to efficiently suppress frequency aliasing [27],[28]. The main idea is based on the observation that the spectrum of a real-valued filter must be *two-sided*, which tends to create aliasing (Fig. 3.10(a),(c)), whereas a carefully designed complex filter can have a *single-sided* spectrum (Fig. 3.10(b),(d)), thus eliminating serious aliasing. In particular, if a complex signal has a single-sided Fourier spectrum its real and imaginary parts constitute a Hilbert

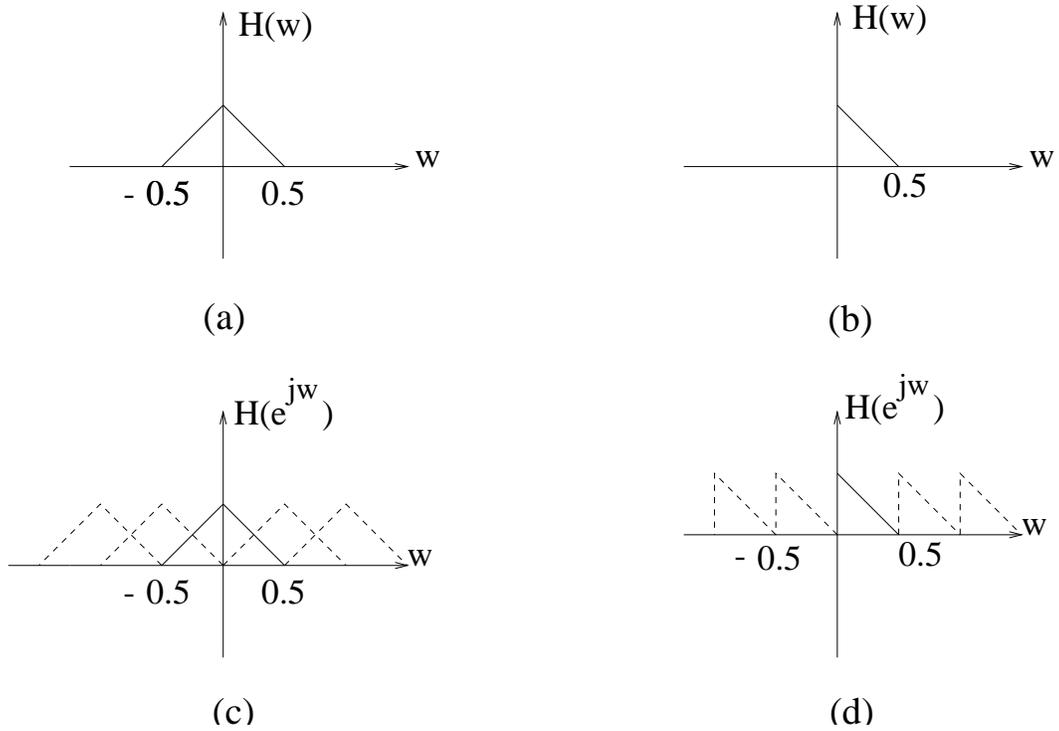


Figure 3.10: Comparison of the aliasing effect of real and complex digital filters. (a) Frequency response of a real-valued analog filter; (b) Frequency response of a complex-valued analog filter; (c) Frequency response of a real-valued digital filter; (d) Frequency response of a complex-valued digital filter. Note the frequency aliasing with the real-valued digital filter (c) which is absent in the complex case (d).

transform pair. That is, if $h(x) = h_r(x) + jh_i(x)$ has a single-sided Fourier spectrum then

$$h_r(x) = \mathcal{H}[h_i(x)] = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{h_i(t)}{t-x} dt \quad (3.12)$$

In the Fourier domain this Hilbert pair relationship is expressed as

$$H_r(\omega) = \begin{cases} -jH_i(\omega) & , \omega > 0 \\ jH_i(\omega) & , \omega < 0 \end{cases} \quad (3.13)$$

Observing the importance of the Hilbert transform pair, researchers have tried to apply it to various practical problems. Abry *et. al.* [40] used it for transient detection. Ozturk [41] suggested it for waveform coding. Hilbert transform pair is also very useful in the design of wavelet transforms. Selesnick [28] employed the Hilbert transform pair in the development of the dual-tree complex wavelet transforms. Freeman and Adelson's steerable filters [42] were based on Hilbert transform. The paper by Beylkin and Torresani [43] is also of related interest. We will soon see that the upper part and lower part of the dual-tree complex wavelet approximately constitute a Hilbert transform pair, precisely why the dual-tree complex wavelet is approximately shift-invariant.

To obtain an in-depth understanding of the dual-tree wavelet transform we now come back to the main ideas of Kingsbury and Selesnick's approaches to designing the shift-invariant dual-tree complex wavelet [27], [28].

Let us consider what happens when the signal is reconstructed using wavelet coefficients of just one type (scaling or wavelet coefficients) from just one level of the dual tree. This models (in an extreme way) the virtual elimination of certain subbands which commonly occurs in many algorithms. For example we might choose to retain only the level-2 wavelet coefficients x_{01a} and x_{01b} from Fig. 3.7, and set all others to zero. If the signal y , reconstructed from just these coefficients, is free of aliasing then the transform can be said to

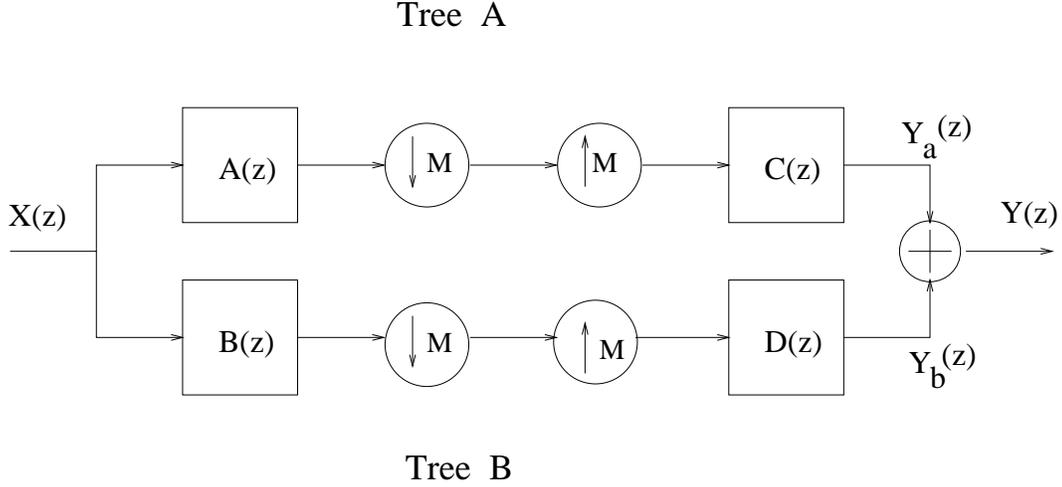


Figure 3.11: Basic configuration of the dual tree if either wavelet or scaling-function coefficients from just level m are retained ($M = 2^m$).

be shift invariant at that level.

Fig. 3.11 shows the simplified analysis and reconstruction parts of the dual tree when coefficients of just one type and level are retained. All down(up)-sampling operations are moved to the output (input) of the analysis (reconstruction) filter banks and the cascaded filter transfer functions are combined. $M = 2^m$ is the total downsampling factor. For example if we retain only x_{01a} and x_{01b} (see Fig. 3.7) then $M = 2^2 = 4$, $A(z) = H_{0a}(z)H_{00a}(z^2)$ and $B(z)$, $C(z)$, $D(z)$ are obtained similarly.

Letting $W = e^{j2\pi/M}$, multirate analysis of Fig. 3.11 gives:

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(W^k z) [A(W^k z)C(z) + B(W^k z)D(z)] \quad (3.14)$$

For shift invariance, the aliasing terms (for which $k \neq 0$) must be negligible. So Kingsbury designed $B(W^k z)D(z)$ to cancel out $A(W^k z)C(z)$ for all non-zero k which give overlap of the pass or transition bands of the filters $C(z)$ or $D(z)$ with those of the shifted filters

$A(W^k z)$ or $B(W^k z)$. Separate strategies are needed depending on whether the filters are lowpass (for scaling functions) or bandpass (for wavelet functions).

For level m in the dual tree, the lowpass filters have passbands $[-f_s/2M, f_s/2M]$, where f_s is the input sampling frequency. The W^k terms in (3.14) shift the passbands in multiples, k , of f_s/M . If $A(z)$ and $C(z)$ have similar frequency responses (as required for near-orthogonal filter sets) and significant transition bands, it is not possible to make $A(Wz)C(z)$ small at all frequencies $z = e^{j\omega}$, because the frequency shift, f_s/M , of $A(z)$ due to W is too small (i.e., the $A(z)$ and $C(z)$ responses tend to overlap at their $-3dB$ points). However, it is quite easy to design $A(W^2z)C(z)$ to be small since the frequency shift of $A(z)$ is twice as great and so the responses no longer overlap significantly. Hence for the lowpass case, Kingsbury suggested to specify $B(W^k z)D(z)$ to cancel out $A(W^k z)C(z)$ when k is odd by letting:

$$B(z) = z^{\pm M/2}A(z) \quad \text{and} \quad D(z) = z^{\mp M/2}C(z) \quad (3.15)$$

so that $B(W^k z)D(z) = (-1)^k A(W^k z)C(z)$. In this way, the unwanted aliasing terms, mainly at $k = \pm 1$, are approximately cancelled out. This is equivalent to a single tree with the decimation of $M/2$ rather than M . (3.15) implies that the tree b samples should interpolate midway between the tree a samples, effectively doubling the sampling rate.

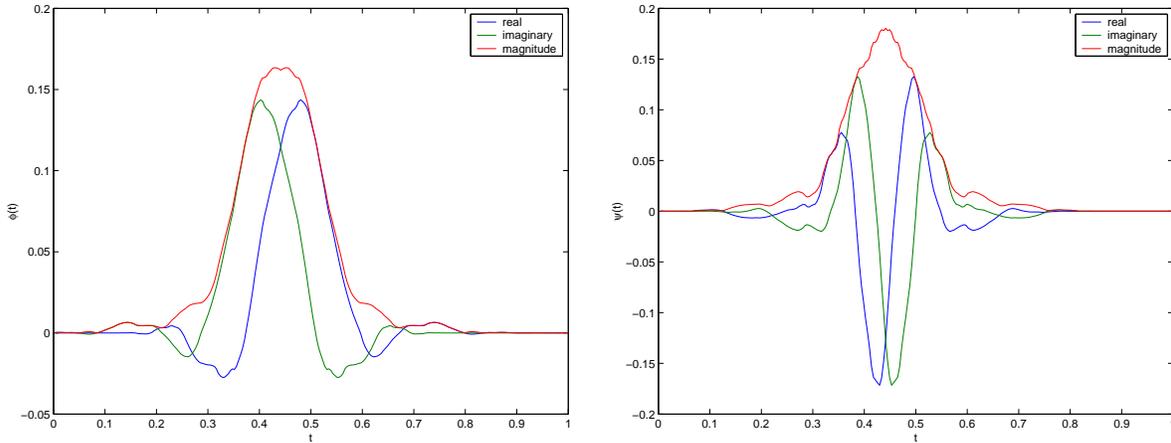
For the bandpass case, the edges of the positive frequency passband of $C(z)$ or $D(z)$, $[f_s/2M, f_s/M]$, will tend to overlap with the edges of the negative frequency passband of $A(z)$ or $B(z)$, which gets shifted either to $[0, f_s/2M]$ or to $[f_s/M, 3f_s/2M]$ when $k = 1$ or 2 , respectively. Similarly for the opposite passbands when $k = -1$ or -2 . Since the main aliasing terms are always caused by the overlap of opposing frequency passbands (i.e., passbands which have opposite polarity of center-frequency in the unshifted filters),

Kingsbury proposed to give B and D positive and negative passbands of opposite polarity while A and C have passbands of the same polarity (or vice versa). For this purpose Kingsbury suggested to construct $A(z)$, $B(z)$, $C(z)$ and $D(z)$ from two prototype *complex* filters $P(z)$ and $Q(z)$, each with just a *single* passband $[f_s/2M, f_s/M]$ and zero gain at all negative frequencies by letting:

$$\begin{aligned}
 A(z) &= \operatorname{Re}(2P(z)) = P(z) + P^*(z) \\
 B(z) &= \operatorname{Im}(2P(z)) = -j(P(z) - P^*(z)) \\
 C(z) &= \operatorname{Re}(2Q(z)) = Q(z) + Q^*(z) \\
 D(z) &= \operatorname{Im}(2Q(z)) = -j(Q(z) - Q^*(z))
 \end{aligned} \tag{3.16}$$

where conjugation is given by $P^*(z) = \sum_n P^*(n)z^{-r}$ and produces negative frequency passbands. In [27] Kingsbury showed that such $P(z)$ and $Q(z)$ are easy to find.

We have shown the basic idea of Kingsbury's approach to designing shift-invariant dual-tree complex wavelets. We now give several examples to further illustrate the shift-invariance property of the 1-D dual-tree wavelet transform. Fig. 3.12 shows the scaling and wavelet basis functions of the complex transform. We can see that the magnitudes change much slower than the real and imaginary parts, which means the energy/magnitude of the dual-tree complex wavelet coefficients varies slowly with shift. Fig. 3.13 is a more concrete example. In Fig. 3.13 we apply the dual-tree complex wavelet transform to 16 shifted ideal unit step signals and keep coefficients of only one type and level and then perform the inverse wavelet transform. We show the results from level-4 wavelet coefficients and scaling coefficients. For comparison, we also do the same experiments for a typical real-valued wavelet transform, the Daubechies-4 wavelet. Good shift-invariance is clear from the fact that each of the reconstructed output components in Fig. 3.13(a),(c) has a shape



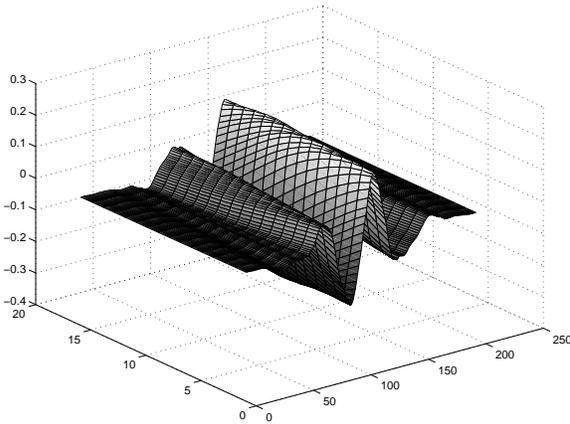
(a) Scaling functions

(b) Wavelet functions

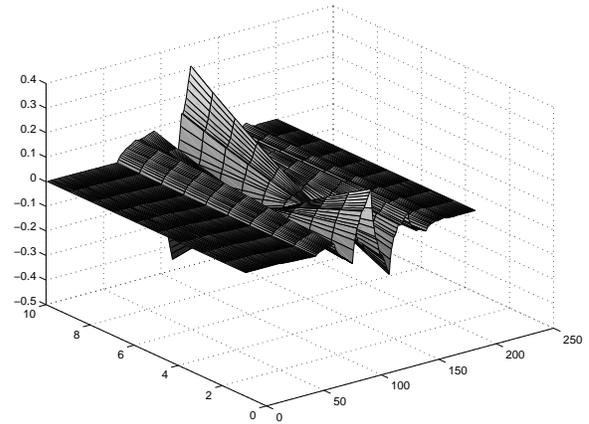
Figure 3.12: Scaling and wavelet functions for the dual-tree complex wavelet transform proposed in [27]. The blue curves are the real components. The green curves are the imaginary parts. The red curves represent the magnitudes.

and amplitude which hardly varies as the input is shifted. Looking at Fig. 3.13(b),(d) we can easily see the reconstruction of the real-valued transform are much less consistent with shift. In particular, over the 16 shifts the energy of the complex wavelet coefficients at any given level vary by no more than 2.5%, whereas the real-valued coefficient energies vary by up to a factor of 5.4. This is a big difference! If we are trying to detect features (e.g., abrupt transients) of a signal from its wavelet decomposition, then it is highly confusing to have such unwanted variations with shift. The impulse responses of the DT-CWT analysis filters we used are listed in Table 3.2.

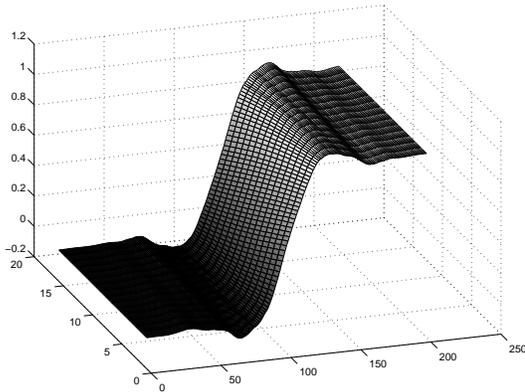
As shown above, Kingsbury found that the dual-tree complex wavelet is approximately shift-invariant when the lowpass filters of one tree interpolate midway between the lowpass filters of the second tree (see (3.15)). This result is derived by forcing shift-invariance



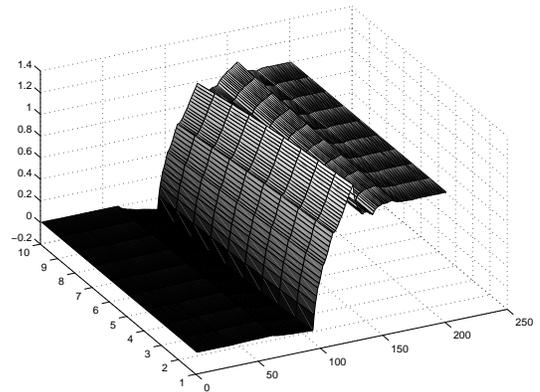
(a) From complex-valued wavelet coefficients



(b) From real-valued wavelet coefficients



(c) From complex-valued scaling coefficients



(d) From real-valued scaling coefficients

Figure 3.13: To test the sensitivity of the complex-valued and real-valued wavelets we reconstruct 16 shifted ideal step signals from the wavelet or scaling coefficients at level 4. It is clear the reconstructions from the real-valued wavelet coefficients (right column) have much larger deformations than those from the complex coefficients (left column).

Real part		Imaginary part	
Low-pass	Band-pass	Low-pass	Band-pass
0.035	0.000	0.000	-0.035
0.000	0.000	0.000	0.000
-0.088	-0.114	-0.114	0.088
0.234	0.000	0.000	0.234
0.760	0.588	0.588	-0.760
0.588	-0.760	0.760	0.588
0.000	0.234	0.234	0.000
-0.114	0.088	-0.088	-0.114
0.000	0.000	0.000	0.000
0.000	-0.035	0.035	0.000

Table 3.2: Table of coefficients for the complex analysis filters. The reconstruction filters are obtained by negating alternate coefficients and swapping bands.

condition (i.e., cancelling out frequency aliasing terms) in the reconstruction equation (3.14). However, Kingsbury designed lowpass and bandpass filters independently and did not explicitly discuss the relationships between them. Because the lowpass and bandpass filters are scaling and wavelet filters respectively, they must satisfy certain constraints set by the wavelet transform, for example the perfect reconstruction condition. Under these additional constraints, are there any analytic relationship between the wavelet functions of the two trees? First recall the conjugate quadrature filter (CQF) pair (refer to Fig. 3.7 for notation):

CQF pair: The analysis filters h_{0a} and h_{1a} of Tree A constitute a CQF pair if

$$\sum_n h_{0a}(n)h_{0a}(n+2k) = \delta(k) = \begin{cases} 1 & , k = 0 \\ 0 & , k \neq 0 \end{cases} \quad (3.17)$$

and $h_{1a}(n) = (-1)^{1-n}h_{0a}(1-n)$. Equivalently, in terms of the Z-transform

$$H_{0a}(z) + H_{0a}(1/z) + H_{0a}(-z) + H_{0a}(-1/z) = 2$$

and

$$H_{1a}(z) = \frac{1}{z}H_{0a}(-1/z)$$

We use the notation $H(z)$ for the Z-transform of $h(n)$. Then the frequency response of the filter is $H(e^{j\omega})$. The corresponding scaling and wavelet functions are denoted by $\phi(t)$ and $\psi(t)$, respectively. And their Fourier transforms are $\Phi(t)$ and $\Psi(t)$. The filters $h_{0b}(n)$ and $h_{1b}(n)$ of the Tree B represent another CQF pair.

We now state the theorem of Selesnick [28] which describes the relationship between the wavelet functions (wavelet filters) of Tree A and Tree B if the lowpass filter of one tree interpolates midway between the lowpass filter of the second tree.

Selesnick's theorem If $H_{0a}(e^{j\omega})$ and $H_{0b}(e^{j\omega})$ are lowpass CQF filters (scaling filters) with

$$H_{0b}(e^{j\omega}) = H_{0a}(e^{j\omega})e^{j(\omega/2)} \text{ for } |\omega| < \pi$$

then the corresponding wavelets are a Hilbert transform pair, i.e.,

$$\psi_{h_b}(t) = \mathcal{H}\{\psi_{h_a}(t)\}$$

It is clear now that since the wavelet functions of Tree A and Tree B are a Hilbert transform pair, thus the complex wavelet $\psi_{h_a}(t) + j\psi_{h_b}(t)$ has a single-sided Fourier frequency and tends to be frequency aliasing-free and shift-invariant for subsampling.

3.2.3 2-D Complex Wavelet Transform

A. Definition of 2-D dual-tree complex wavelet

According to the above discussions the scaling and wavelet functions of the 1-D dual-tree wavelet transforms are

$$\phi_a(x) + j\phi_b(x) \quad \text{and} \quad \psi_a(x) + j\psi_b(x) \quad (3.18)$$

The 2-D dual-tree complex wavelets are obtained by the *efficient* tensor product of the 1-D complex basis functions. It has two scaling functions and six wavelet functions (so it

is 4-times expansive). If we denote the two scaling functions by $\phi_i(x, y), i = 1, 2$ and six wavelet functions by $\psi_i(x, y), i = 1, 2, \dots, 6$ we have

$$\begin{aligned}\phi_1(x, y) &= (\phi_a(x) + j\phi_b(x))(\phi_a(y) + j\phi_b(y)) \\ &= [\phi_a(x)\phi_a(y) - \phi_b(x)\phi_b(y)] + j[\phi_a(x)\phi_b(y) + \phi_b(x)\phi_a(y)]\end{aligned}$$

and

$$\begin{aligned}\phi_2(x, y) &= (\phi_a(x) + j\phi_b(x))(\phi_a(y) + j\phi_b(y))^* \\ &= [\phi_a(x)\phi_a(y) + \phi_b(x)\phi_b(y)] + j[-\phi_a(x)\phi_b(y) + \phi_b(x)\phi_a(y)]\end{aligned}$$

The six 2-D complex wavelet functions are defined similarly:

$$\begin{aligned}\psi_1(x, y) &= [\phi_a(x)\psi_a(y) - \phi_b(x)\psi_b(y)] + j[\phi_a(x)\psi_b(y) + \phi_b(x)\psi_a(y)] \\ \psi_2(x, y) &= [\phi_a(x)\psi_a(y) + \phi_b(x)\psi_b(y)] + j[-\phi_a(x)\psi_b(y) + \phi_b(x)\psi_a(y)] \\ \psi_3(x, y) &= [\psi_a(x)\phi_a(y) - \psi_b(x)\phi_b(y)] + j[\psi_a(x)\phi_b(y) + \psi_b(x)\phi_a(y)] \\ \psi_4(x, y) &= [\psi_a(x)\phi_a(y) + \psi_b(x)\phi_b(y)] + j[-\psi_a(x)\phi_b(y) + \psi_b(x)\phi_a(y)] \\ \psi_5(x, y) &= [\psi_a(x)\psi_a(y) - \psi_b(x)\psi_b(y)] + j[\psi_a(x)\psi_b(y) + \psi_b(x)\psi_a(y)] \\ \psi_6(x, y) &= [\psi_a(x)\psi_a(y) + \psi_b(x)\psi_b(y)] + j[-\psi_a(x)\psi_b(y) + \psi_b(x)\psi_a(y)]\end{aligned}$$

Note the 2-D complex wavelet has a *separable* implementation, which implies low complexity in computation. However, unlike the usual real-valued separable 2-D wavelet transform, the 2-D dual-tree complex scaling and wavelet functions is not separable. Instead, it is constructed by *combining* several separable wavelets, as shown in the above equations. Indeed, the 2-D dual-tree complex wavelet is direction-selective, a property which the separable real-valued 2-D wavelet lacks. We will give a more detailed discussion about the property of

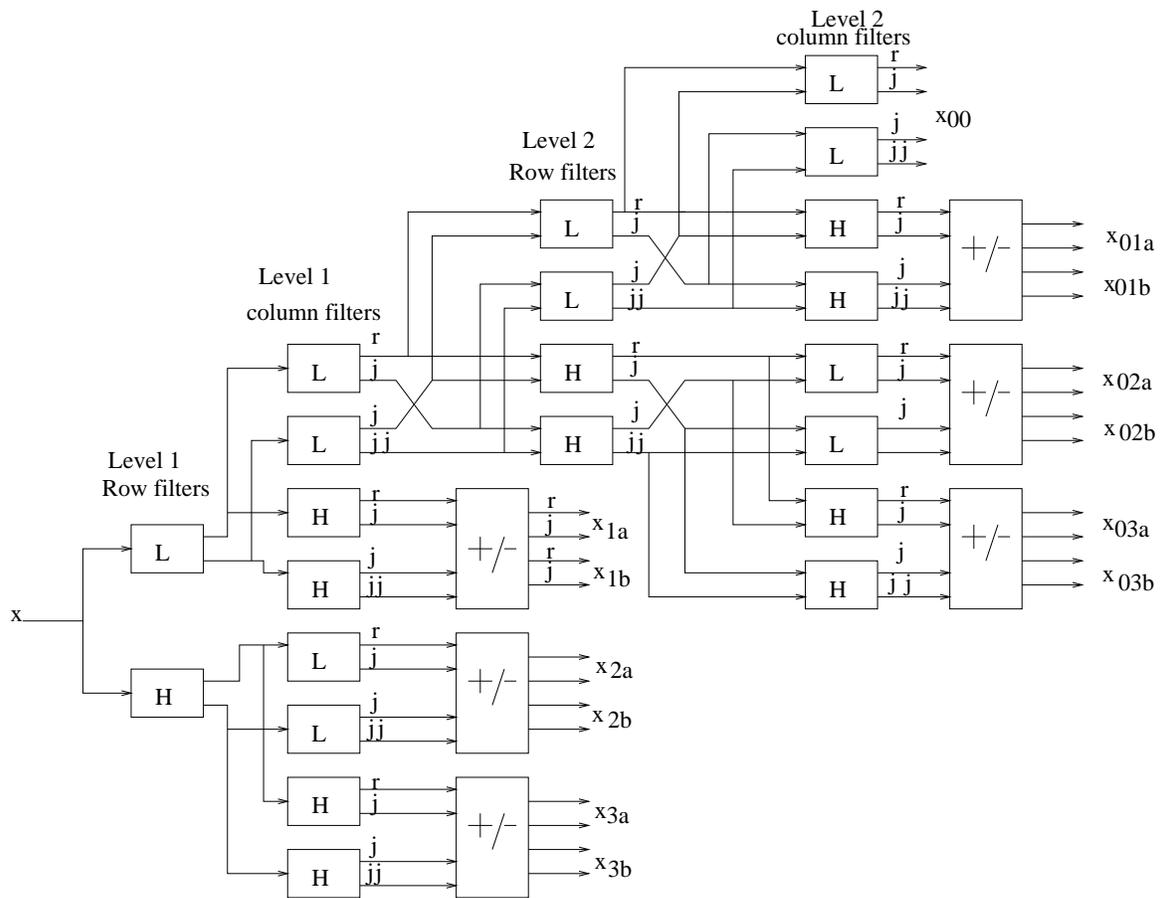


Figure 3.14: Two levels of the complex wavelet transform.

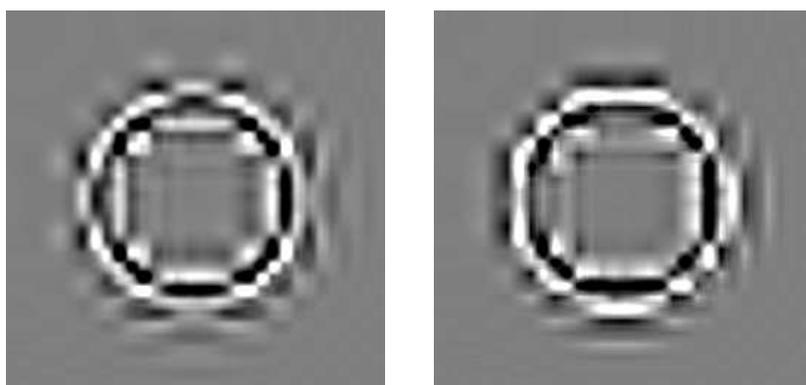
direction-selectivity in the following. The block diagram of 2-D dual-tree complex wavelet transform is shown in Fig. 3.14.

B. 2-D dual-tree complex wavelet properties

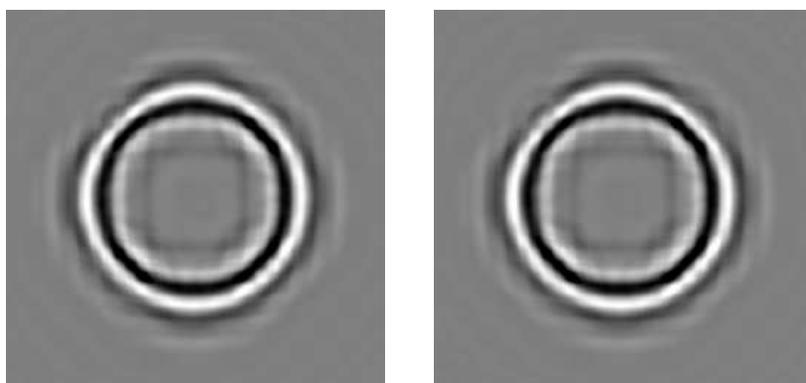
In this subsection we will discuss several important properties of the 2-D dual-tree wavelet through examples. These properties are

1. Shift-invariance;
2. Direction-selectivity, and
3. Low complexity.

Shift invariance: We have shown that the 1-D dual-tree complex wavelet transform is approximately shift-invariant. This is a very important property of the complex wavelet transforms, which the real-valued decimated wavelet transforms lack. Fig. 3.13 shows the shift-variance of a decimated real-valued 1-D wavelet transform and the approximate shift-invariance of the dual-tree complex wavelet transform. To illustrate the sensitivity of the 2-D wavelet transforms to shifts we used an image of a circle, applied a decimated real-valued 2-D and the dual-tree complex 2-D wavelet transforms to the circle image and its shift, respectively, and then performed a reconstruction from the wavelet coefficients of level 4. Fig. 3.15(a) and (b) give the reconstructions of the original circle and its shift from the wavelet coefficients of the real-valued wavelet transform, respectively. The difference between Fig. 3.15(a) and (b) is clear. For comparison, Fig. 3.15(c) and (d) display the reconstructions from the wavelet coefficients of the dual-tree complex wavelet transform, which are much more similar. This simple experiment clearly shows that the 2-D dual-tree



(a) Using real-valued wavelet (b) Using real-valued wavelet



(c) Using complex wavelet (d) Using complex wavelet

Figure 3.15: Compare the reconstructions of shifted circles from the decimated real-valued wavelet ((a) and (b)) with those from the dual-tree complex wavelet ((c) and (d)). The shift-variance between (a) and (b) is apparent, whereas (c) and (d) looks quite similar. In addition, the reconstructions in (a) and (b) suffer from much more serious artifacts. The real-valued wavelet used here is the 8-tap orthogonal Daubechies wavelet.

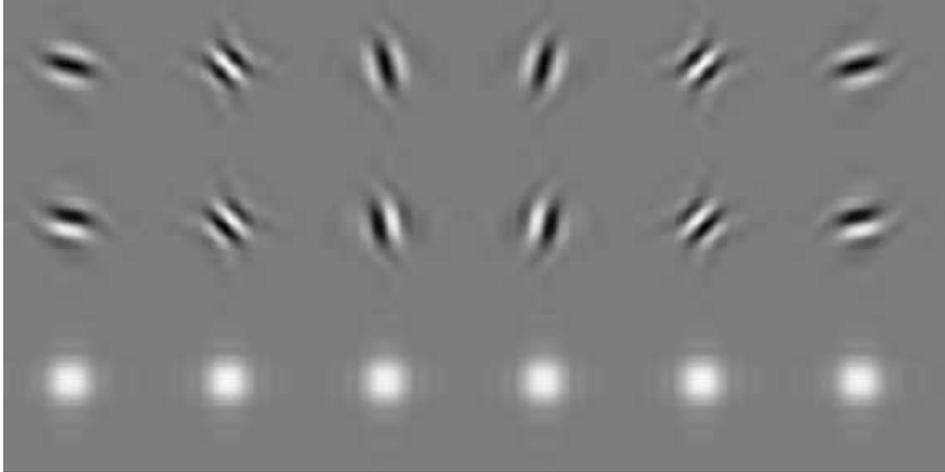


Figure 3.16: Typical wavelets associated with the 2-D dual-tree complex wavelet transform. The first and second rows illustrate the real and imaginary components of the wavelet in the spatial domain. Both real and imaginary components have good directional selectivity. Also note the absence of the checkerboard phenomenon in these functions. The third row plots the magnitudes of the complex wavelet. The magnitude does not have an Gibbs-like oscillatory behavior. Instead they are bell-shaped envelopes.

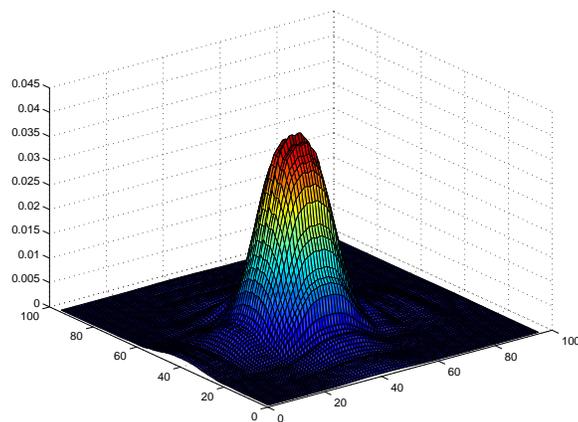
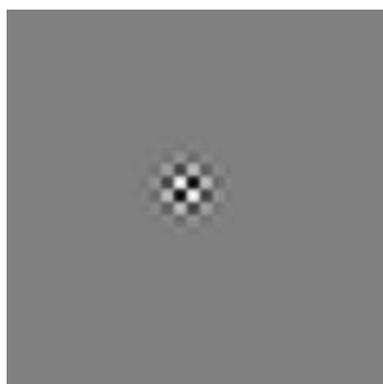


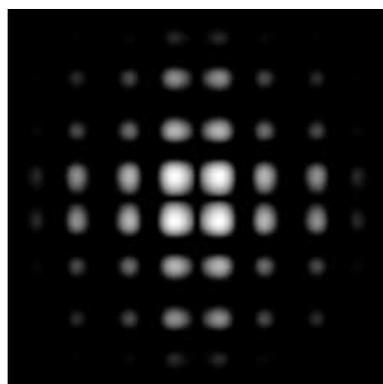
Figure 3.17: The bell-shaped envelope of one complex wavelet basis in Fig. 3.16.



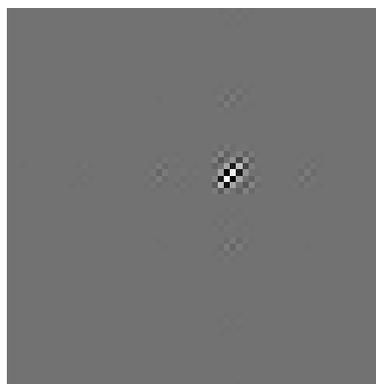
(a) Real basis function



(b) Complex basis functions (real and imaginary)



(c) Spectrum of (a)



(d) Spectrum of (b)

Figure 3.18: Compare the directionality of separable 2-D real-valued and 2-D dual-tree complex wavelets. The top row is the basis functions in the spatial domain. The bottom row is the corresponding Fourier spectra of the top functions. Note the good directional selectivity of the complex wavelet bases. Also note how the energy of the complex wavelet concentrates in the first quadrant, effectively reducing decimation-induced frequency aliasing.

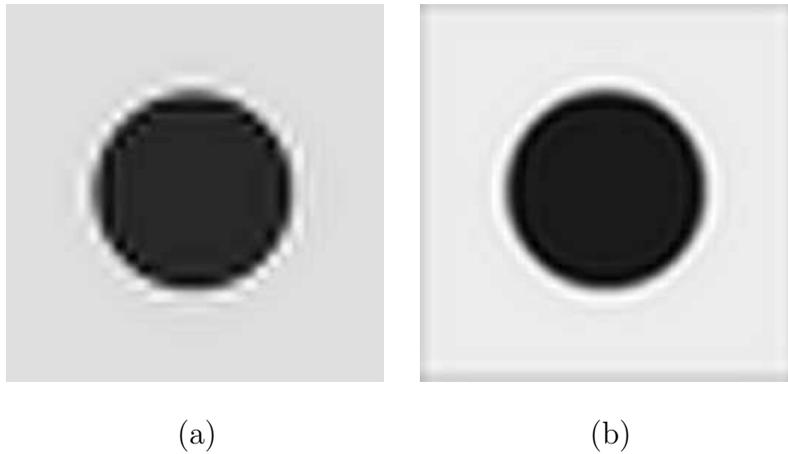


Figure 3.19: Compare the reconstruction of a circle from (a) the decimated real-valued wavelet with (b) that from the dual-tree complex wavelet. Note, in particular, the clear artifacts in (a).

complex wavelet transform is almost shift invariant, whereas the usual real-valued wavelet transform is very sensitive to shifts. Another sharp difference in the reconstructions is that the real-valued wavelets lead to much more serious artifacts than those of the complex wavelet. This type of artifact is due to decimation-induced frequency aliasing. Because of the intrinsic Hilbert relationship in the complex wavelet, frequency aliasing is largely alleviated.

Direction-selectivity: In addition to shift-invariance, good direction-selectivity is also expected for 2-D transforms, which means that the transform basis functions should be anisotropic. We already showed in Fig. 3.5 the bases of a real-valued *separable* 2-D wavelet transform. A typical real-valued separable 2-D wavelet transform has three detail components: horizontal, vertical and diagonal. Note the poor directionality of the diagonal component, which mixes the 45 and -45 degree orientations. Poor directionality can also

be seen from the checkerboard artifact of the basis function, which often create artifacts. For comparison, the basis functions of the dual-tree complex wavelet are given in Fig. 3.16, where we note the absence of checkerboard artifacts. Shown in the third row of Fig. 3.16 is the magnitude of the complex wavelet, which does not have an oscillatory behavior (Gibbs-like oscillation), but instead they are bell-shaped envelopes like the popular Gabor filters (Fig. 3.17). The directionality of the 2-D complex bases can also be seen in the Fourier frequency domain in Fig. 3.18(d), where the frequency energy concentrates in the first quadrant (for comparison, the frequency energy of the real-valued wavelet spreads in all four quadrants (Fig. 3.18(c))). It is interesting to note the special spectrum structure of the complex wavelet can also reduce the decimation-related frequency aliasing and related artifacts.

The advantage of the directional selectivity of the complex wavelet is shown in Fig. 3.19, where we compare the reconstruction results of the circle image from the level-4 scaling and wavelet coefficients of both the real- and complex-valued wavelet transforms. This models (in an extreme way) the virtual elimination of almost all wavelet subbands which happens in practice in low bit-rate compression or denoising in low SNR circumstances. Note the annoying artifact with the result of the decimated real-valued 2-D wavelets (Fig. 3.19(a)), which is, however, absent in the reconstruction of the 2-D dual-tree complex wavelet (Fig. 3.19(b)).

Low redundancy/complexity: One may argue that the overcomplete real-valued 2-D wavelets (e.g., [39]) are also shift-invariant and that the *non-separable* real-valued 2-D wavelet-like transforms (e.g., ridgelet [44], curvelet [15] and contourlet [45]) are also direction-selective. However, the overcomplete real-valued 2-D wavelet transforms have much higher redundancy (which usually means higher complexity) than the 2-D dual-tree

complex wavelet transform. For example, the redundancy of the 2-D dual-tree complex wavelet is 4, independent of the number of levels of decomposition, whereas the redundancy of the 2-D overcomplete wavelet is normally $4L$, where L denotes the number of decomposition levels. The 2-D dual-tree complex wavelet also compares favorably with the directional (non-separable) real-valued 2-D wavelets in terms of complexity. Although it is non-separable, the dual-tree complex wavelet is implemented by combining *separable* 1-D wavelets, which is much faster than the *non-separable* implementations of ridgelets, curvelets or contourlets.

The interesting properties of the dual-tree complex wavelet make it well shaped for its applications in image processing, such as restoration/denoising, analysis/synthesis and feature detection. For wavelet-based image denoising many researchers found shift-invariance [9],[20],[21] and direction-selectivity [44],[15],[14] of the transform bases are critical in terms of MSE and human visual effect. Owing both properties, the dual-tree complex wavelet should be promising for image denoising. Indeed, several authors [27],[46] have reported encouraging results using the 2-D dual-tree complex wavelet for image denoising. In Chapter 6 we will use the complex wavelet for image denoising and compare it with other wavelet transforms.

Chapter 4

Wavelet-based Image Enhancement

This chapter presents a wavelet-based technique for enhancing the perceptual sharpness of an image. We stress that the goal of this work is not in complete reconstruction, but rather in augmenting image sharpness, as perceived by a human observer. This goal, which is intrinsically comprised of more subjective constraints, is of importance in a variety of application domains. One such scenario, for example, is when a given input image is blurred with no exact degradation model known. If a degradation model does in fact exist, restoration techniques can be applied, together with other frequency enhancement techniques [47, 48]. The enhancement scheme described in this chapter can then be applied as an additional enhancement utility.

Given a blurred image, the classic problem of image enhancement is to recover completely lost high-frequency components so that the processed image looks sharper and more pleasing to human observers. Traditionally this was performed by so-called unsharp filters [48], which are *linear* (and generally shift-invariant) processors. It is well known that linear shift invariant (LSI) filters can modify only the *existing* frequencies but can

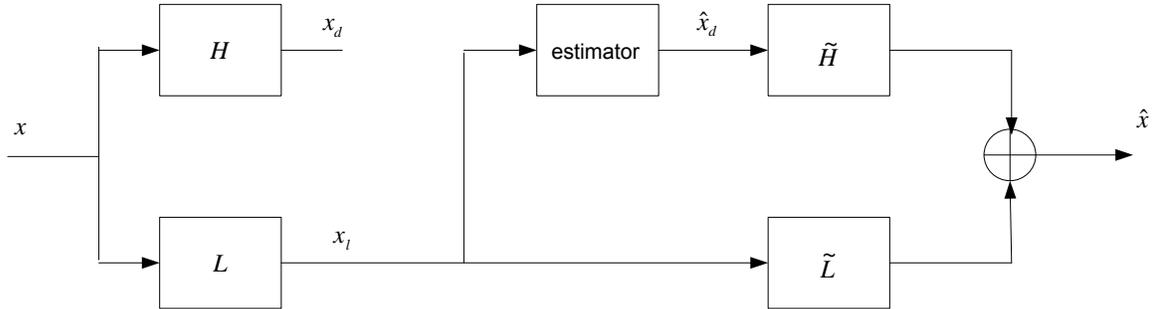


Figure 4.1: Multiscale image enhancement: given a lowpass filtered signal x_l , we use it to estimate the detail (highpass filtered) component x_d . x_l and \hat{x}_d are combined to give the enhanced result.

not generate new frequency components, and thus can not recover the lost high-frequency component in principle. Non-linear filtering methods were also studied by several authors [49],[50],[51],[52]. However, so far there are only *ad hoc* solutions and designing general-purpose non-linear filters remains difficult.

Recently, several multiscale image enhancement approaches have been proposed with interesting results [53],[54],[55],[56]. Image enhancement in a multiscale context can be considered as the estimation of coefficients in high frequency subbands based on those in lower-frequency subbands (Fig. 4.1). All of these approaches attempt to utilize the inter-scale dependency (mainly the dependency among edges) to extrapolate lost high-frequency components. Greenspan *et al.* [53] and Burt *et al.* [54] used zero-crossings of the second derivative of smoothed images to locate edges, and based on the ideal step edge model they estimated the inter-scale relations of edges. They then used these relations to estimate edges in finer scales from those of the low-frequency subbands. Kinebuchi and Woo [55],[56] assumed a different approach: they first used a hidden Markov tree model (HMT) [57],[58] to infer the probability of each hidden state and corresponding variances.

Then a Gaussian mixture model (GMM) (corresponding to the hidden states) is used for each coefficient and wavelet coefficients in the highest subband are generated randomly (by sampling) using the estimated state probabilities and variances. In estimating variances the property of exponential decay of variances was assumed [59] with roughly estimated exponents.

In this chapter, we also try to handle the image enhancement problem in the wavelet-based multiscale framework. Specifically, we will use the so-called Mallat-Zhong wavelet transform [39], which is an approximate multiscale Canny edge detector [60]. Like [53], we treat the image enhancement as *edge* enhancement, as smooth areas of an image are well preserved in the approximation low frequency part of the wavelet transform, which often does not need to be enhanced [39],[61]. However, unlike [53] we use local maxima of first-order derivatives to detect edges (instead of zero-crossings of second order derivatives in [53]) since the first-order derivative is more robust to noise. In the following sections we will first introduce the Mallat-Zhong wavelet transform, then describe our enhancement approach, followed by experimental results and comparison with other enhancement methods.

4.1 Problem Formulation

Image enhancement in the wavelet domain can be treated as the estimation of wavelet coefficients in the highest frequency subband from coefficients in the lower frequency subbands. This can be explained with the help of Fig. 4.1: the high resolution image is at the input to a filter bank. The low-pass filter, L , represents the effects of the image acquisition system, which is generally modeled by a Gaussian function [47]. If we were able to

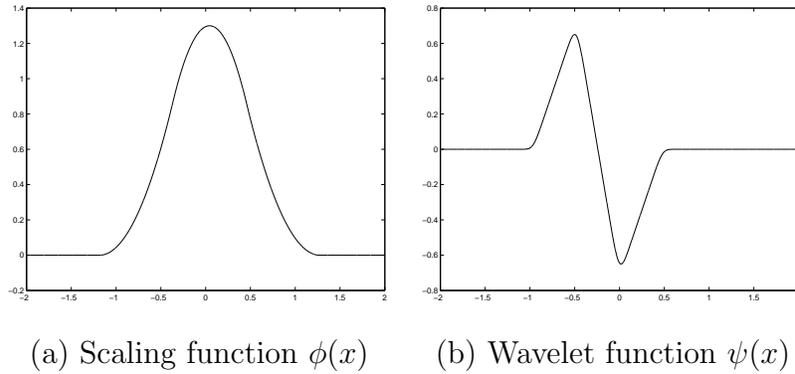


Figure 4.2: Mallat-Zhong scaling and wavelet functions: $\psi(x) = \frac{d}{dx}\phi(x)$.

filter the original high-resolution image \mathbf{x} with the high-pass filter H to obtain the detail signal, x_d , and if we had a perfect reconstruction filter bank, it would then be possible to reconstruct the original image perfectly. In image enhancement problems we clearly do not have access to the detail signal, x_d , so we have to estimate it. We will achieve this goal in a wavelet-based multiscale framework, where the wavelet transform used in this chapter is the Mallat-Zhong wavelet [39].

4.2 1-D Mallat-Zhong Wavelet

For clarity, we use a 1-D example to illustrate the Mallat-Zhong wavelet transform [39]. 2-D transforms are implemented as the tensor products of the 1-D transforms, as defined in Chapter 3.

The Mallat-Zhong wavelet is a quadratic spline function, which is the first-order derivative of a smoothing function $\phi(t)$:

$$\psi(t) = \frac{d\phi(t)}{dt} \quad (4.1)$$

Fig. 4.2(a) and (b) show the scaling and wavelet functions, respectively. As $\phi(t)$ is designed

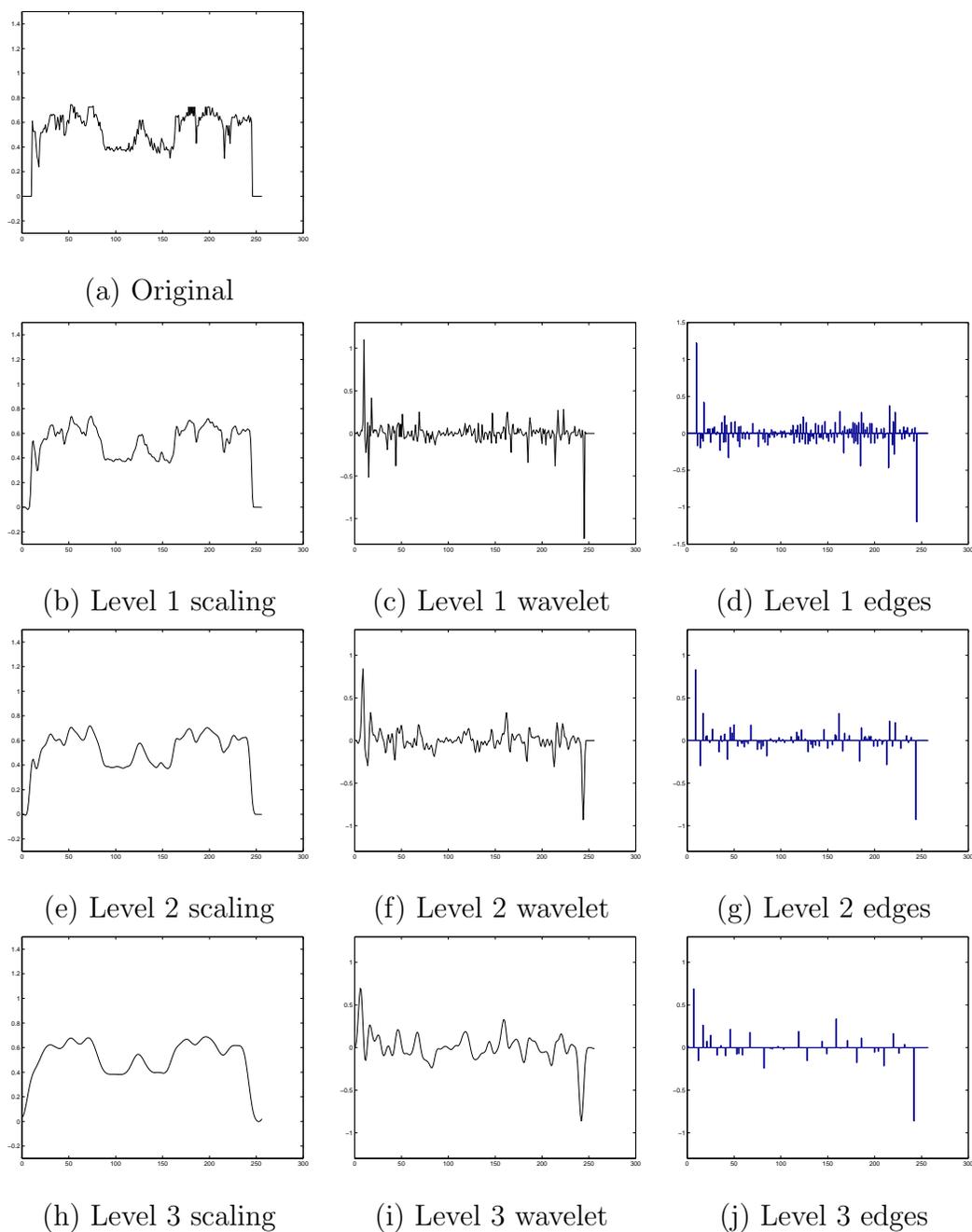


Figure 4.3: An example of the 3-level 1-D Mallat-Zhong wavelet transform. The multiscale edges in (d),(g) and (j), found as the extrema of the wavelet signal, are highly correlated.

to approximate a Gaussian function, $\psi(t)$ approximates the first-order derivative of a Gaussian. The resulting wavelet decomposition can then be thought of as a multiscale edge representation, and thus can be used as an approximate multiscale Canny edge detector [60]. In the Canny edge detector, edges are marked as the local maxima of the first-order derivative of a Gaussian smoothed version of the original signal $x(t)$. The variances of the Gaussian smoothing functions determine the scales of edges to be detected.

If we denote by

$$\xi_s(t) = \frac{1}{s} \xi\left(\frac{t}{s}\right) \quad (4.2)$$

the dilation by a scaling factor of s of any function $\xi(t)$, then the wavelet transform of $x(t)$ at scale s and position t is

$$W_s^d x(t) = x(t) * \psi_s(t) \quad (4.3)$$

where $*$ is the convolution operation and the superscript d denotes the detail component. According to (4.1), we derive that

$$W_s^d x(t) = x(t) * \left(s \frac{d\phi_s(t)}{dt}\right) = s \frac{d}{dt} (x(t) * \phi_s(t)) \quad (4.4)$$

The related scaling coefficients are computed by

$$W_s^a x(t) = x(t) * \phi_s(t) \quad (4.5)$$

where the superscript a denotes the approximation component. Since $\phi_s(t)$ approximates a Gaussian function of scale s , the Canny edges can be found by computing the local maxima of $W_s^d x(t)$. For discrete dyadic wavelet transforms, the scale s is constrained to be 2^j , where $j \in \mathbb{Z}$ is a non-negative integer. Let $(t_n^j)_{n \in \mathbb{Z}}$ be the abscissa where $|W_{2^j}^d x(t)|$ is locally maximum. Then $(t_n^j)_{n \in \mathbb{Z}}$ are the locations of edges at scale j and $|W_{2^j}^d x(t_n^j)|$ are the strengths of edges. For a J -level wavelet transform, the collection $\{W_{2^j}^a(t^j), W_{2^j}^d x(t_n^j), j =$

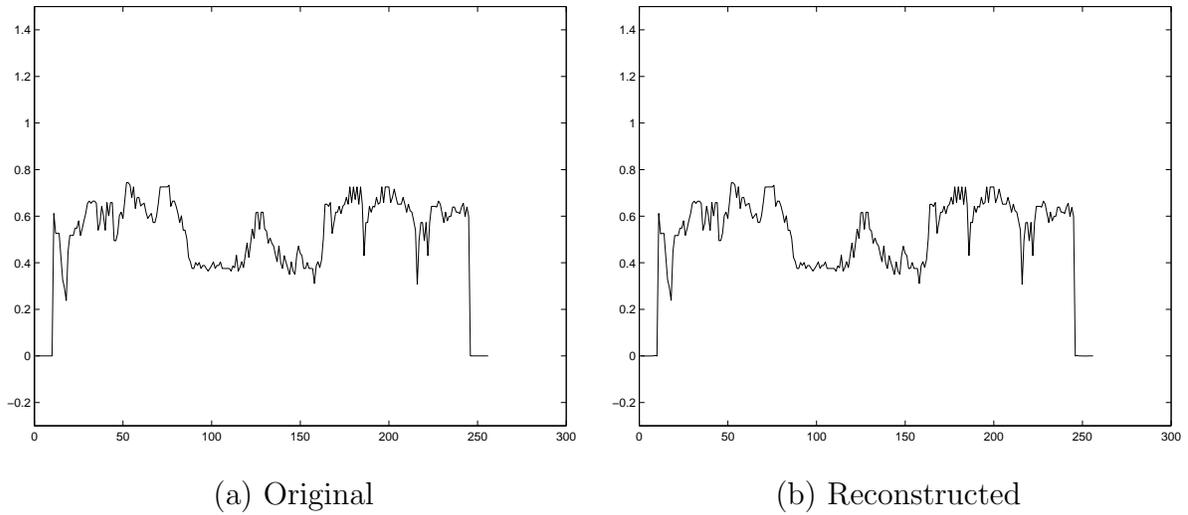


Figure 4.4: Reconstructed signal from the multiscale edge representation shown in Fig. 4.3. The difference between (a) and (b) is very small.

$1, 2, \dots, J\}$ is called the multiscale edge representation of a signal [39]. Fig. 4.3 gives an example of a 3-level 1-D Mallat-Zhong wavelet transform. It is clear that the wavelet coefficients at different levels ((c),(f),(i)) are highly correlated, thus so are the multiscale edges. Empirically, we should be able to use the relationship among the scales to perform estimation. Specifically, if we can estimate the level 1 edges from the edges at higher levels then through the inverse wavelet transform we can sharpen the original image by incorporating the estimated level 1 edges. The key problem, then, is how to estimate the highest frequency subband from other subbands.

Although the multiscale edge representation $\{W_{2^j}^a(t^j), W_{2^j}^d(t^j), j = 1, 2, \dots, J\}$ is not complete, Mallat *et al.* [39] showed that a *close* approximation $\hat{x}(t)$ of the original $x(t)$ can be reconstructed from it. Their reconstruction algorithm imposed two conditions on the local maxima of $W_{2^j}\hat{x}(t)$:

1. At each scale 2^j , for each local maximum located at t_n^j , $W_{2^j}^d \hat{x}(t_n^j) = W_{2^j}^d x(t_n^j)$.
2. At each scale 2^j , the local maxima of $W_{2^j}^d \hat{x}(t^j)$ are located at abscissa $(t_n^j)_{n \in Z}$.

The reconstruction procedure turns out to be a minimization problem, for the details of which we refer the reader to [39]. Fig. 4.4 shows the reconstruction of the signal in Fig. 4.3(a) from its 3-level multiscale edge representation after 20 iterations. The differences between the original and reconstructed signals are so small that we can not see any differences. This high-quality reconstruction result suggests that predicting the high-frequency subband (for enhancement purposes) requires only a good prediction of the high-frequency edges.

This completes the discussion of 1-D Mallat-Zhong wavelet transform and related multiscale edge representation of signals. The extension to 2-D cases is straightforward: a separable Kronecker product is used to construct 2-D wavelet functions from 1-D functions, and edges are recognized as local maxima of gradient magnitudes along gradient directions. It should be noted that, unlike many standard wavelet transforms [1], 2-D Mallat-Zhong wavelet has only two detail components (horizontal and vertical) instead of three. Fig. 4.5 gives an example of the 2-D Mallat-Zhong wavelet transform. Fig. 4.6 shows the related multiscale edges. It is clear that the multiscale edges are *highly* dependent across scales, a property which we will exploit for image enhancement. Like 1-D signals, images can also be almost perfectly reconstructed from their multiscale edge representations (see Fig. 4.7).



(a) Original



(b) Level 1 scaling



(c) Level 1 wavelet (vertical)



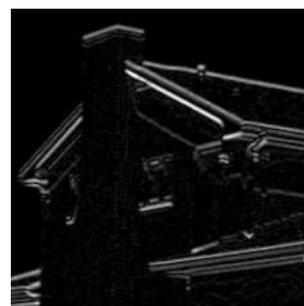
(d) Level 1 wavelet (horizontal)



(e) Level 2 scaling



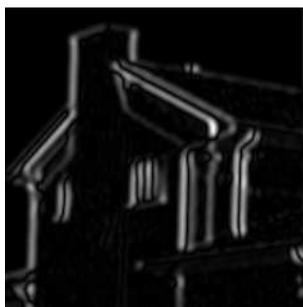
(f) Level 2 wavelet (vertical)



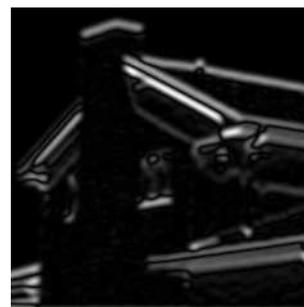
(g) Level 2 wavelet (horizontal)



(h) Level 3 scaling



(i) Level 3 wavelet (vertical)



(j) Level 3 wavelet (horizontal)

Figure 4.5: An example of the 3-level 2-D Mallat-Zhong wavelet transform. Note the wavelet coefficients (both horizontal and vertical) at different levels are closely correlated.



(a) Level 1

(b) Level 2

(c) Level 3

Figure 4.6: 2-D multiscale edges. Note the high dependence among 3 levels.



(a) Original

(b) reconstructed

Figure 4.7: Image *House* and its reconstruction based on multiscale edges. The difference between (a) and (b) is so small that we can not see any differences.

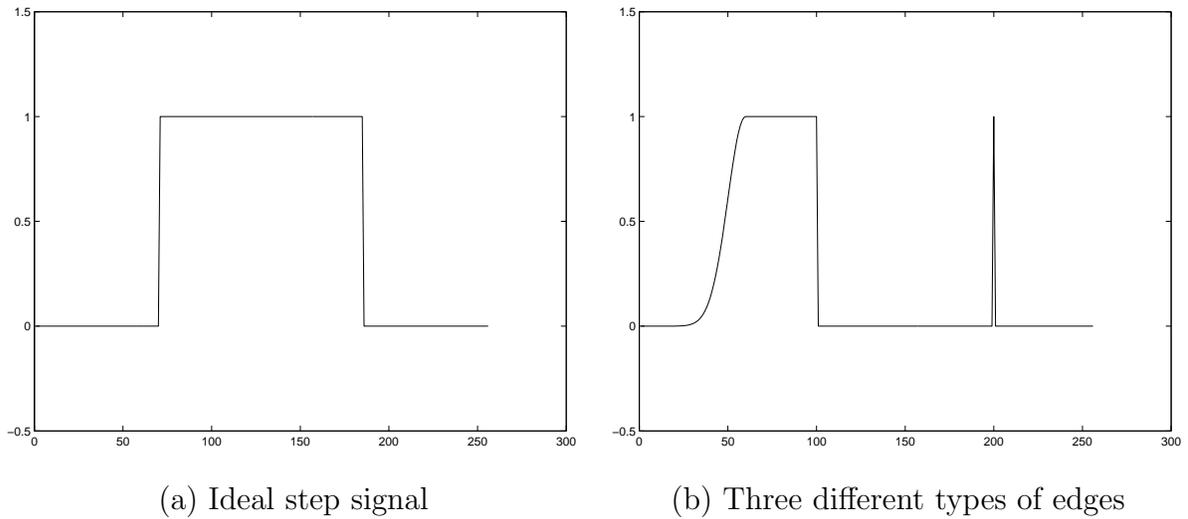


Figure 4.8: The three edges in (b) can be characterized by parameters α and σ (from left to right): smooth ($\alpha = 0, \sigma = 1$), step ($\alpha = 0, \sigma = 0$) and delta ($\alpha = -1, \sigma = 0$).

4.3 Lipschitz Exponent Estimation

As shown in Fig. 4.4 and Fig. 4.7, signals and images can be very closely reconstructed from their multiscale edge representations. Therefore we can treat an image enhancement problem as an edge estimation problem. That is, given edges in the low-frequency bands we hope to estimate edges in the high-frequency bands.

For any edges with a fixed shape (e.g., an ideal step edge in Fig. 4.8(a)), we can easily find the relationship between inter-scale edges simply by taking the wavelet transform and measuring, for example the magnitude ratios between them. However, there are many different types of edges in natural images (see, for example, Fig. 4.8(b)), which can behave quite differently in their multiscale representations. To show this important property we give in Fig. 4.9 the 3-level Mallat-Zhong wavelet decomposition of the signal in Fig. 4.8(b). We can see that the modulus of the edge at abscissa 50 increases as the scale increases,

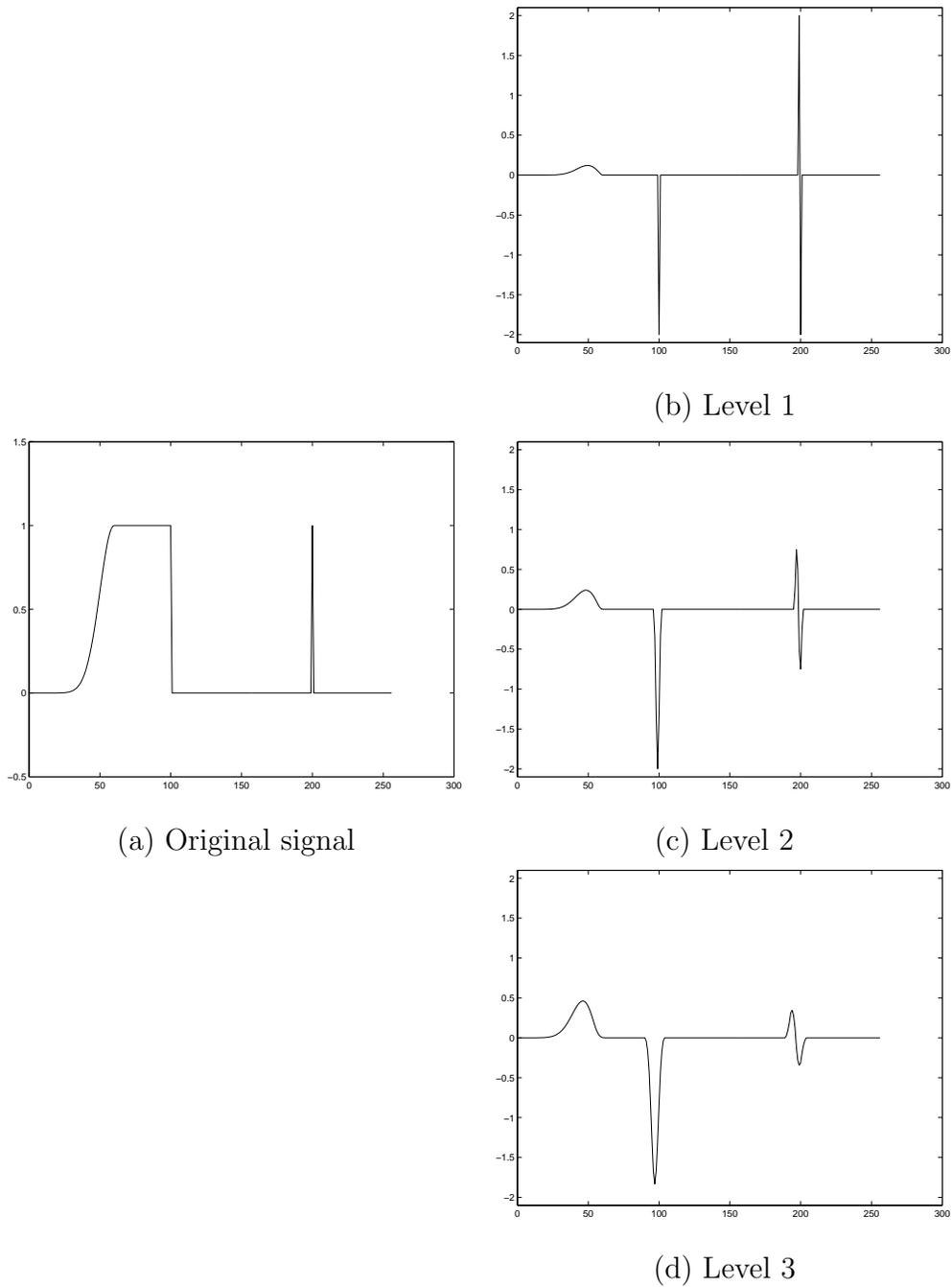


Figure 4.9: Compare the behavior of different types of edges in multiscale representation. From level 1 to level 3, the magnitude of the left-most edge gradually increases, the middle one has little change while the right-most one decreases.

whereas the modulus at abscissa 200 decreases as the scale increases. So to exploit the inter-scale dependency of edges we need to be able to adapt to edge shapes in an image. Mathematically, the scale-dependent behavior is generally characterized by Lipschitz exponents α .¹ Conveniently, for the Mallat-Zhong wavelet these Lipschitz exponents can be computed from the evolution of the multiscale edges across scales. The relationship between α and wavelet coefficients $W_{2^j}x(t)$ is described in the following theorem [62]:

Theorem A function $x(t)$ is uniformly Lipschitz α over (a, b) if and only if there exists a constant $K > 0$ such that for all $x \in (a, b)$, the wavelet transform satisfies

$$|W_{2^j}x(t)| \leq K \cdot 2^j \cdot s_0^{\alpha-1}, \quad \text{with } s_0 = \sqrt{2^{2j} + \sigma^2} \quad (4.7)$$

where σ^2 is a parameter which defines the local smoothness of an edge. So the evolution behavior of an edge is completely determined by α and σ^2 . The values of α and σ^2 for the three edges in Fig. 4.8(b) are (from left to right) $(\alpha = 0, \sigma = 1)$, $(\alpha = 0, \sigma = 0)$ and $(\alpha = -1, \sigma = 0)$ [39]. Substituting these values into (4.7), respectively, we can see why those edges changes in the ways as shown in Fig. 4.9. When $\sigma^2 = 0$ the edges are singular (e.g., edges at abscissas 100 and 200), in which case (4.7) reduces to

$$|W_{2^j}x(t)| \leq K(2^j)^\alpha \quad (4.8)$$

From (4.7) it is clear that if σ and α are given, we can then estimate the magnitudes

¹Let $0 \leq \alpha \leq 1$. A function is uniformly Lipschitz α over an interval $[a, b]$ if and only if there exists a constant K such that for any $(x_0, x_1) \in [a, b]^2$

$$|f(x_0) - f(x_1)| \leq K|x_0 - x_1|^\alpha \quad (4.6)$$

of wavelet coefficients at level $j - 1$ from level j by

$$\frac{|W_{2^j}x(t)|}{|W_{2^{j-1}}x(t)|} = 2 \left(\frac{2^{2j} + \sigma^2}{2^{2(j-1)} + \sigma^2} \right)^{(\alpha-1)/2}. \quad (4.9)$$

If we further assume the same sign, then we can finally estimate $W_{2^{j-1}}x(t)$ from $W_{2^j}x(t)$.

Clearly the key issue is the estimation of σ and α . First, it should be noted that although (4.7) is true for all the wavelet coefficients, we choose to work only on local coefficient maxima (i.e., edges) because they are more robust to noise and because the signal or image can be very nearly reconstructed from the multiscale edges, as previously discussed. We use a least-squares method to estimate σ and α . Let us suppose a sequence of modulus maxima that propagate from scale 2^J up to scale 2^2 and converge to the abscissa t_0 (a sequence referred to as a “fingerprint” by Witkin [63]). Let a_j be the magnitude of the edge on the fingerprint at scale 2^j . We compute σ and α (and K) so that the inequality of (4.7) is as close as possible to an equality for each $\{a_j, j=2, \dots, J\}$. These values are obtained by minimizing

$$\sum_{j=2}^J (\log_2|a_j| - \log_2(K) - j - \frac{\alpha - 1}{2} \log_2(\sigma^2 + 2^{2j}))^2 \quad (4.10)$$

where the minimum was found using a steepest descent algorithm.

Finding σ and α completes our enhancement algorithm, which is summarized as follows:

1. Take the blurred signal/image $x_l(t)$ as input. Here we assume the blurred signal is the first-level approximation signal of the Mallat-Zhong wavelet transform, i.e. $x_l(t) = W_{2^1}^a x(t)$.
2. Perform the Mallat-Zhong wavelet transform of $x_l(t)$ [39].
3. Detect multiscale edges as the local maxima of wavelet coefficient magnitudes.

4. Estimate K , σ and α from the multiscale edges using (4.10).
5. Estimate the edges of level 1, $W_{2^1}^d x(t_n^j)$, using (4.9).
6. Combine $x_l(t)$ and $W_{2^1}^d x(t_n^j)$ to reconstruct an enhanced signal $\hat{x}(t)$, using the reconstruction algorithm designed by Mallat and Zhong [39].

4.4 Experimental Results

Our first two enhancement tests apply to 1-D signals. The original signals are shown in Fig. 4.10(a) and (b). Fig. 4.10(a) is a synthetic signal having three different types of edges (at abscissa 50, 100 and 200, respectively), as in Fig. 4.8(b). We are interested to see whether our enhancement approach can adapt to these edges and enhance them properly. Fig. 4.10(b) is a real-world signal, which is row 200 of the texture-rich image *Mandrill* (see Fig. 4.18(a)). Fig. 4.10(c)(d) give the first-level approximation signals, which are our blurry inputs. The enhanced signals are shown in Fig. 4.10(e)(f). We used a 4-level Mallat-Zhong wavelet decomposition. Comparing Fig. 4.10(e) with (a) and (c), it is clear that the three different-shape edges are properly enhanced. Examining Fig. 4.10(b),(d) and (f) of the natural signal, we can see many details show up in our enhanced results (f), which are lost in the blurred input (d). Because our goal is the extrapolation and inference of high-frequency structure, it is insightful to examine the changes before and after enhancement in the Fourier frequency domain. Fig. 4.11 plots the Fourier spectra of the signals in Fig. 4.10. It is very clear our approach augments the frequency components missing in the input signals.

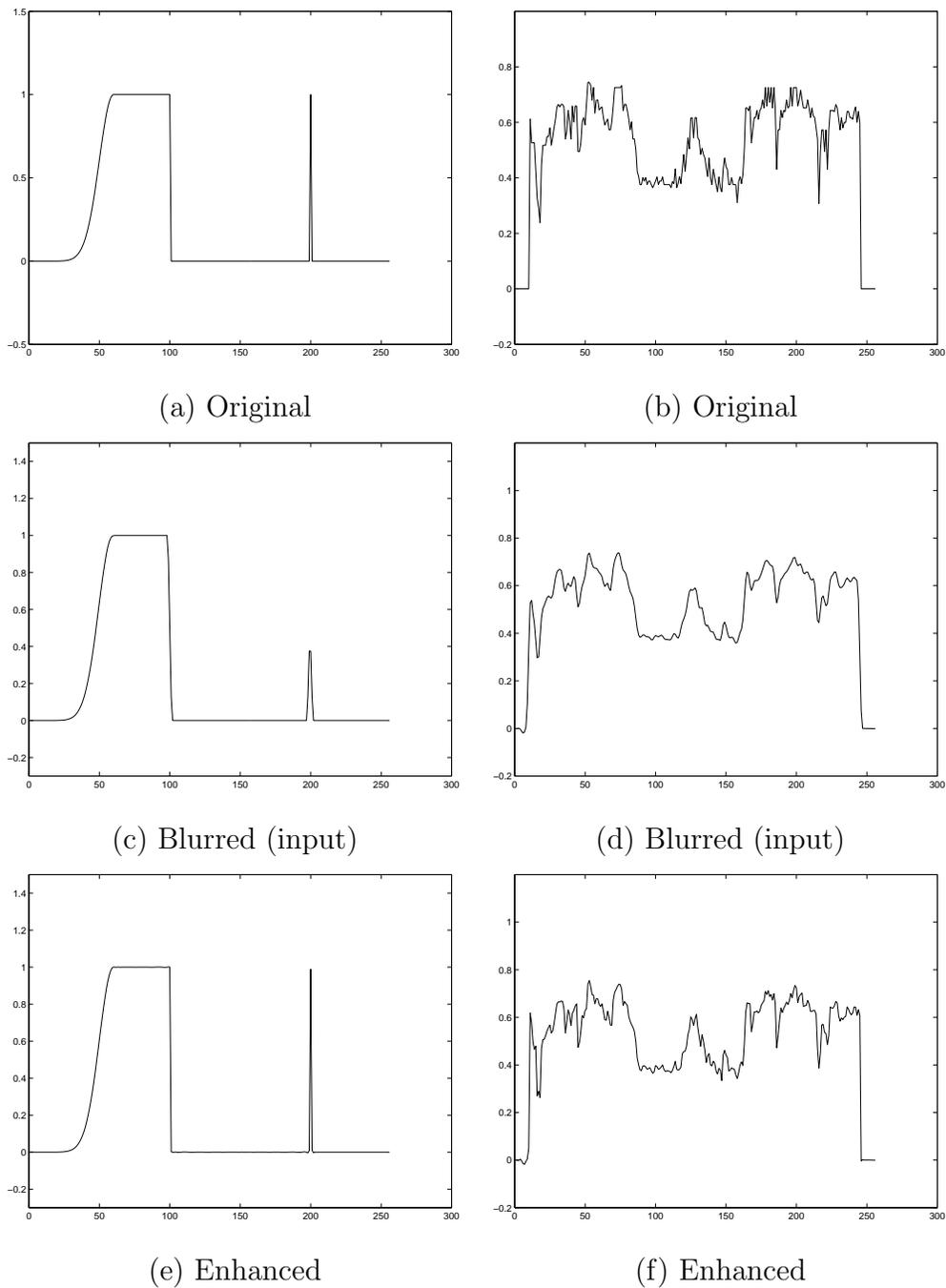


Figure 4.10: Enhancement results of two 1-D signals: a synthetic one and a natural signal. The three different-shape edges (at abscissa 50, 100 and 200) of the synthetic signal are properly enhanced in (e). For the real-world signal, the enhancement (f) shows many details which are missing in the blurry input (d).

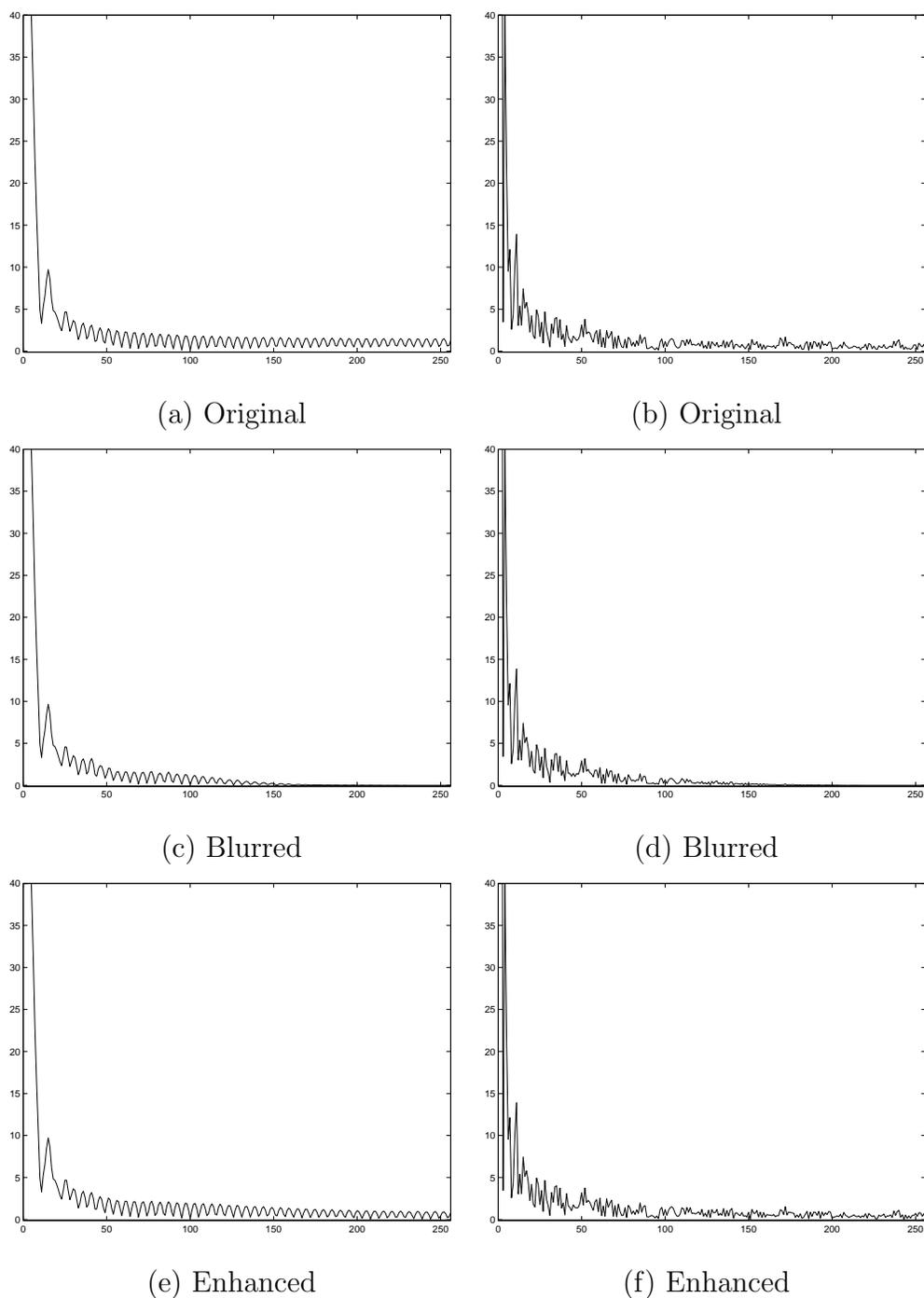


Figure 4.11: Examining enhancement results in the Fourier domain. High-frequency components, which are lost in (c) and (d), have been properly extrapolated ((e) and (f)).

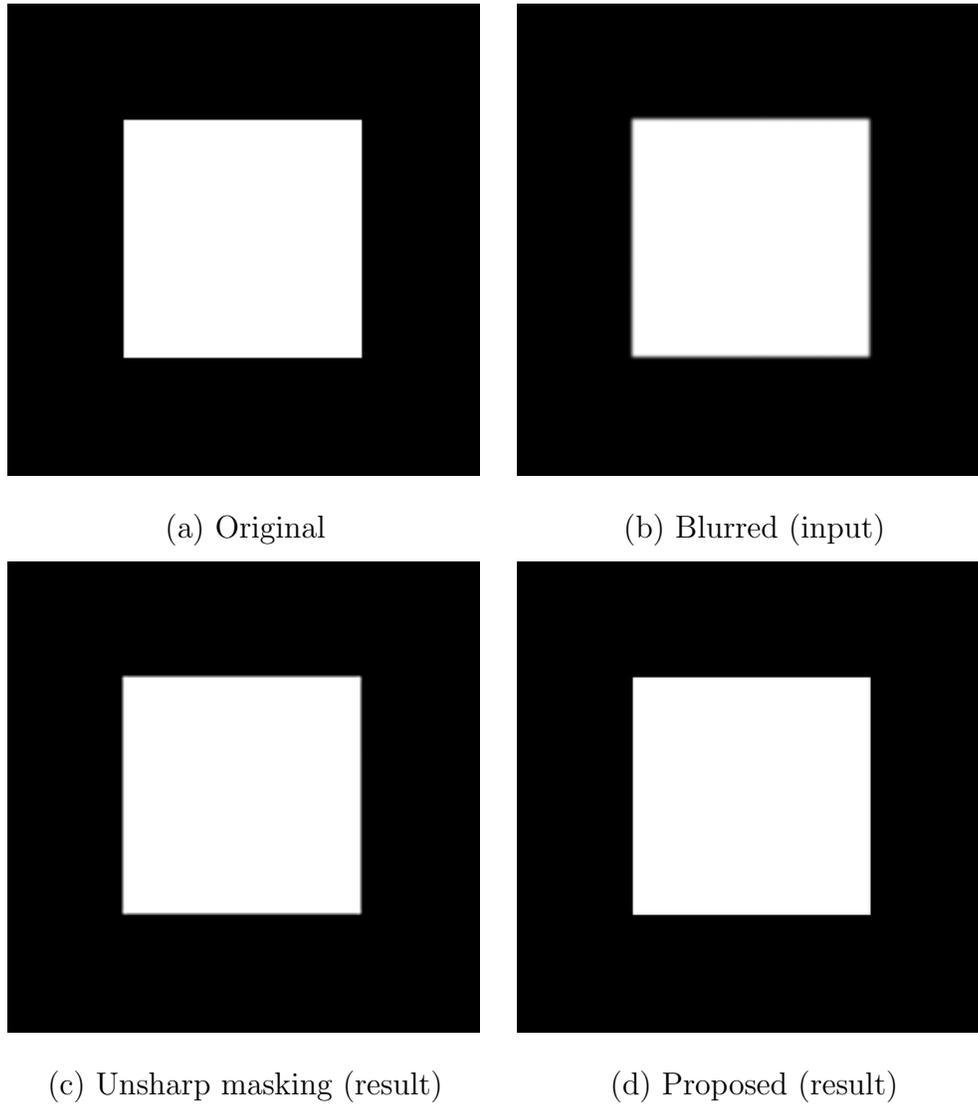


Figure 4.12: Enhancement results of the synthetic image *Square*. Visually, the proposed approach gives sharper result (d) than the unsharp masking (c). Also, (d) has less Gibbs-like oscillation artifact.

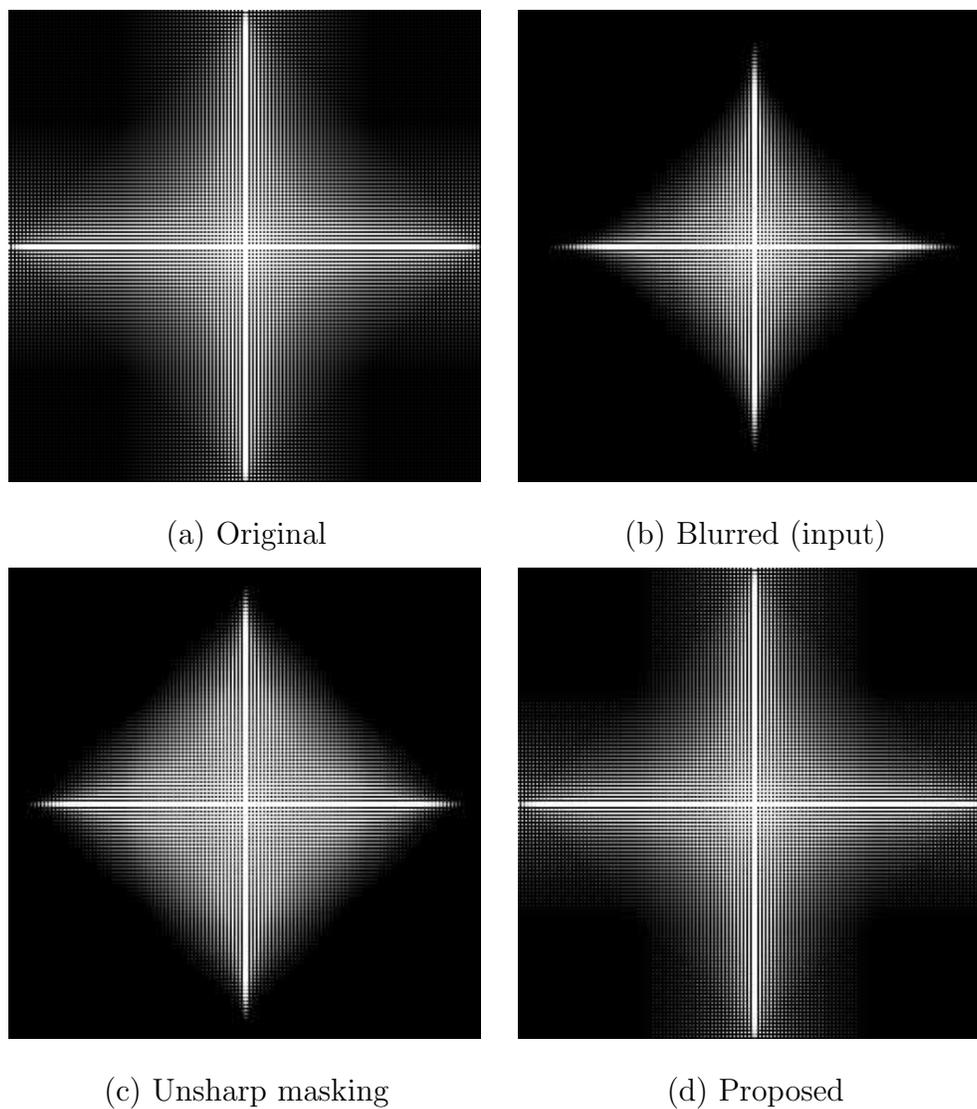


Figure 4.13: Spectra of synthetic image *Square* before and after enhancement. The missing high frequency components are properly “recovered” by the proposed approach (d). By comparison, unsharp masking (c) can only enhance the weak but existing high frequency components.



(a) Original



(b) Blurred (input)



(c) Unsharp masking



(d) Proposed

Figure 4.14: Enhancement results of image *Lena*.

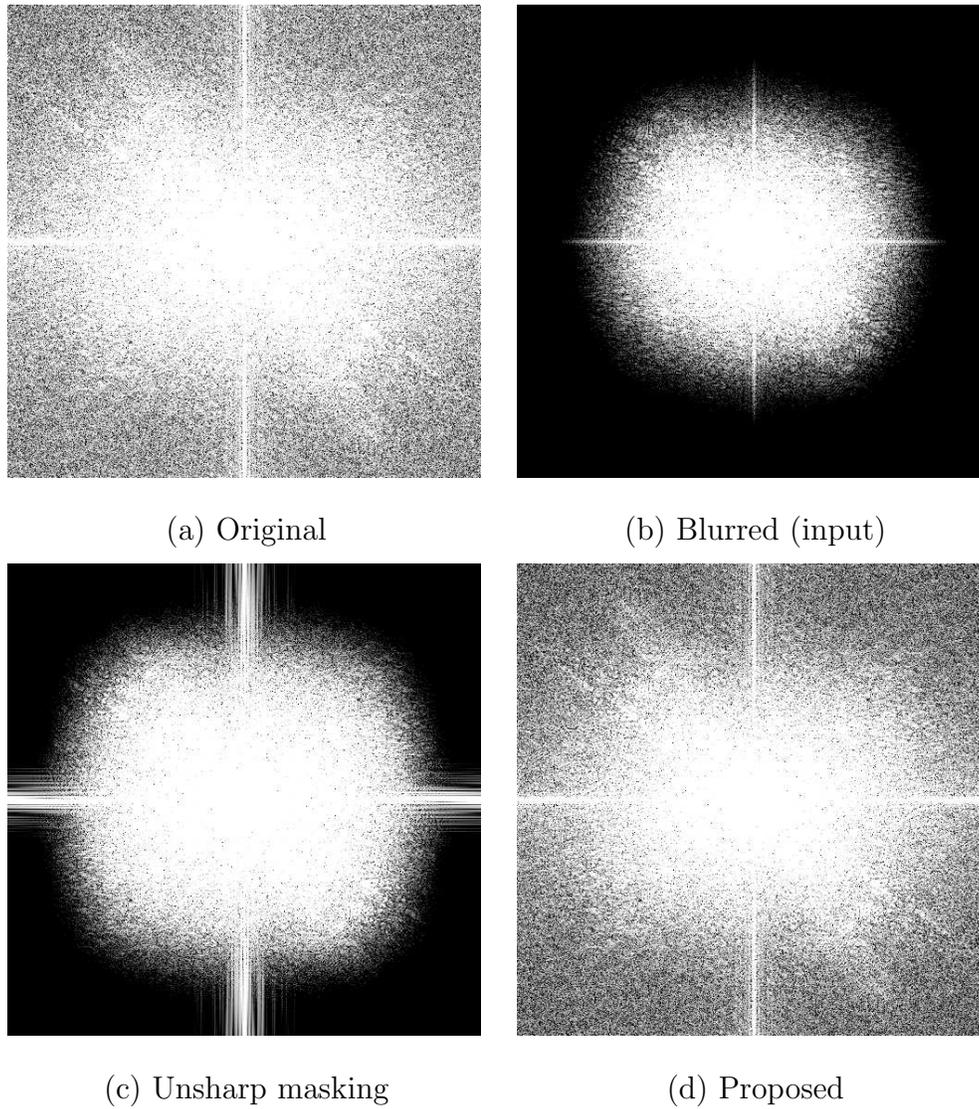


Figure 4.15: Spectra of image *Lena* before and after enhancement. In (d), the proposed method can expose much more missing high frequency components than the unsharp masking.



(a) Original



(b) Blurred (input)



(c) Unsharp masking



(d) Proposed

Figure 4.16: Enhancement results of image *House*

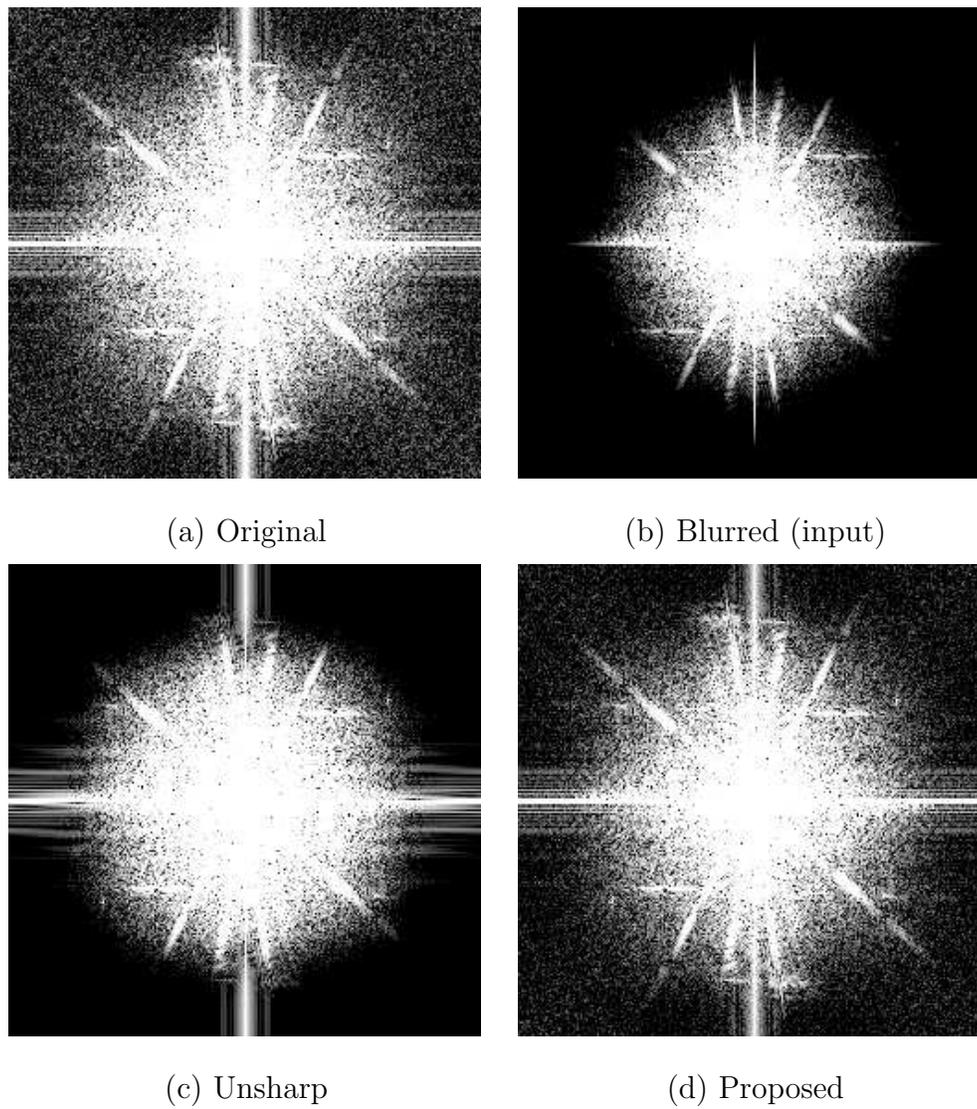
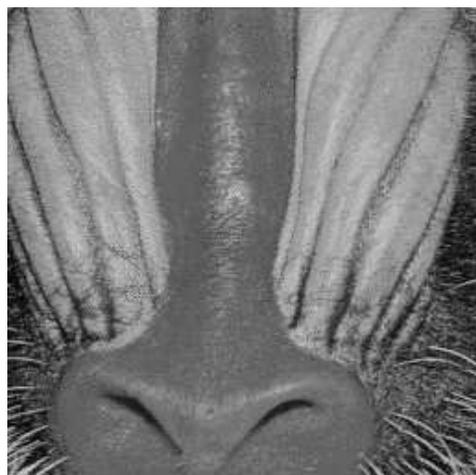


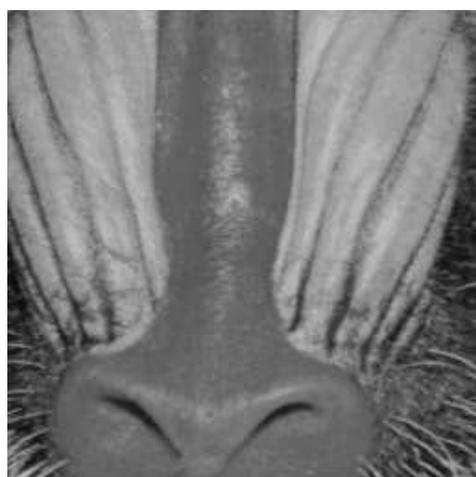
Figure 4.17: Spectra of image *House* before and after enhancement.



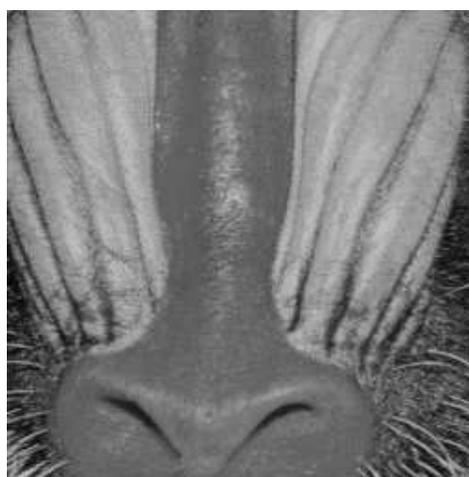
(a) Original



(b) Blurred (input)

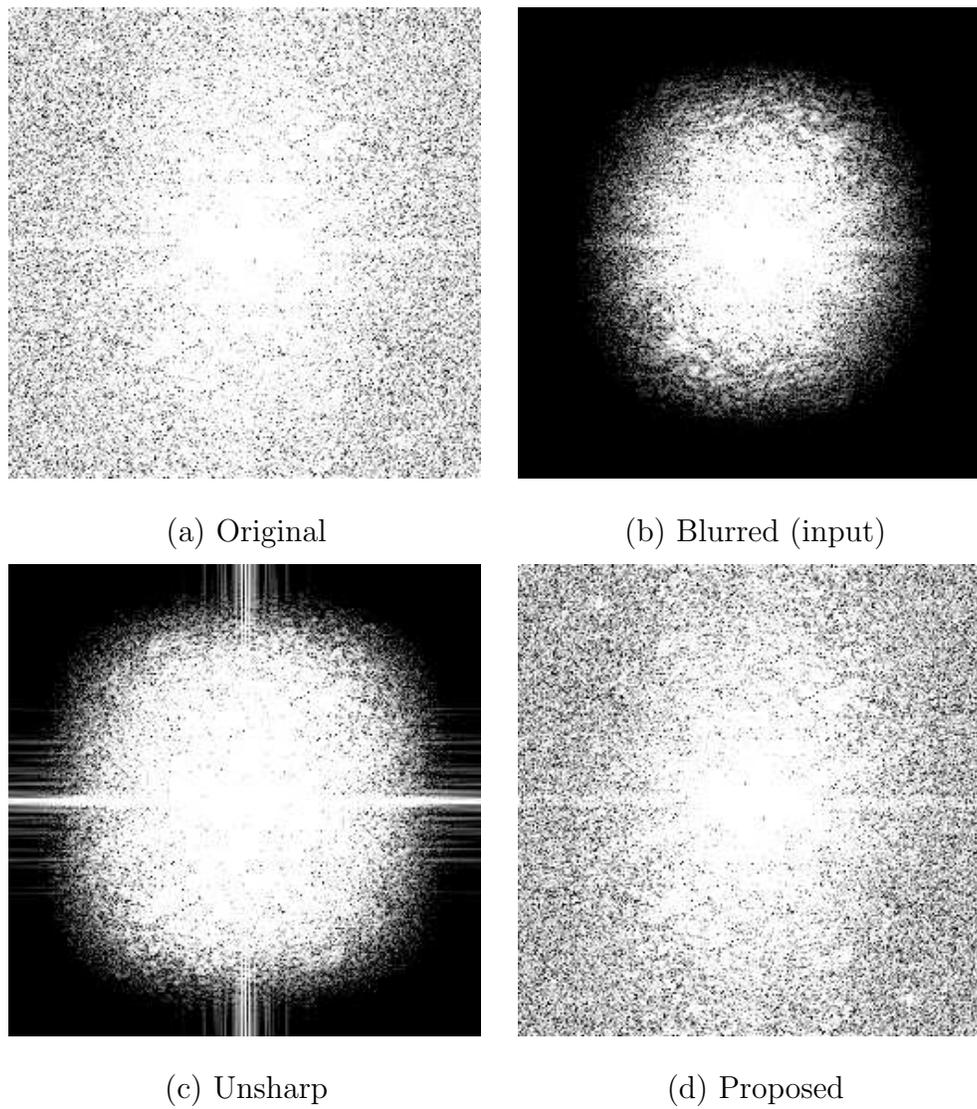


(c) Unsharp



(d) Proposed

Figure 4.18: Enhancement results of image *Mandrill*

Figure 4.19: Spectra of image *Mandrill*

We now show several image enhancement results using the proposed approach, which we compare with the traditional unsharp filters (using the Matlab *fspecial* function), where we will assess the enhanced sharpness aesthetically, by eye and frequency augmentation. For each image tested, we manually adjust the parameters of unsharp filters to give optimal results aesthetically.

We begin with the synthetic image *Square*, shown in Fig. 4.12. Our proposed enhancement, Fig. 4.12(d), looks sharper and more pleasing to our eyes than standard enhancement. The differences are, however, much more evident in the frequency domain, where there is a much greater augmentation and filling-in of high frequency details (see Fig. 4.13). For additional tests, the reader is referred to enhancements of *House*, *Lena* and *Mandrill*, with corresponding results shown in Figs. 4.16 to 4.19. In all these experiments we observe sharper results from the proposed method than from unsharp masking. The high-frequency components have also been properly augmented.

4.5 Discussion

We have shown good enhancement results of our proposed approach. Next, we compare our approach with several other enhancement methods in the literature.

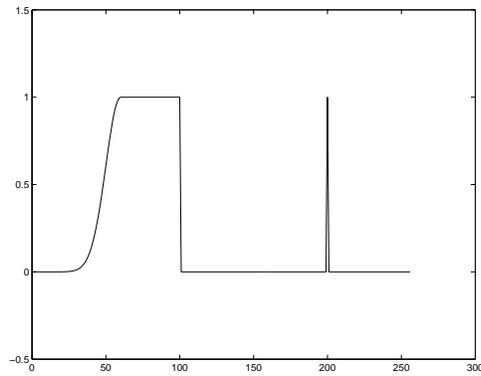
First we compare with the well-used unsharp filter [47]. Theoretically, if we know the degradation model and the model is invertible, then the unsharp masking filter can exactly recover the original signal. However, usually the degradation model is not known and we have to try different parameters of unsharp filters and choose the one yielding the best result. It should be noted that because the unsharp filter is a linear and shift-invariant filter it can not create any new frequency components. On the other hand, our proposed

approach works on a quite different idea. It tries to exploit the close correlation of wavelet coefficients across scales (but at essentially the same location) to estimate the detail signal at finest scale from those at coarser scales. It is non-linear and thus has the potential to create new high frequency component. These differences are clearly shown in the above examples.

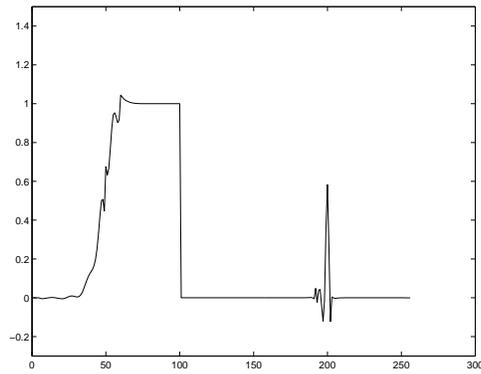
Our approach was mainly motivated by the idea of Greenspan *et al.* [53], who also performed image enhancement in a multiscale framework. Their approach was based on the observation that the *ideal* step signal is almost invariant (in the sense of phase, position and magnitude) in its multiscale decomposition. However, we observed two potential problems with their method:

1. They used the Marr-Hildreth edge detector [64] to find edges. It is well-known that Marr-Hildreth edge detector is sensitive to noise because it uses the zero-crossings of *second-order* derivatives to locate edges.
2. They used a single edge model, the ideal step edge, to estimate parameters. However, a natural image always consists of many different types of edges. Obviously, one *single* edge model can not represent all of them well. For example, Fig. 4.20(a) shows a signal with three different types of edges. The enhancement results by Greenspan's approach and ours are shown in Fig. 4.20(b) and (c), respectively. It is clear that due to using wrong parameters, in Fig. 4.20(b) the edge at abscissa 50 is over-sharpened while the edge at abscissa 200 is not enhanced enough. For sharp comparison, our approach can well adapt to various edges and properly enhance them.

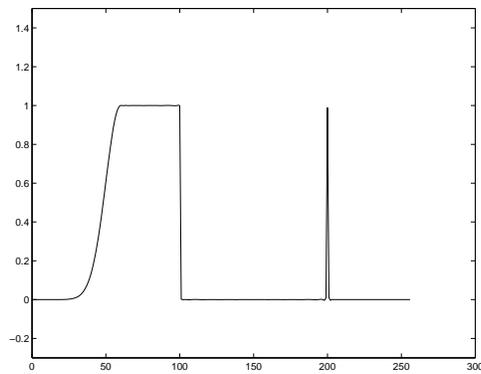
More recent work on the multiscale image enhancement has used the wavelet-based hidden Markov tree (HMT) model [55], [56]. The wavelet-based HMT was intended to capture



(a) Original



(b) Greenspan's approach



(c) Proposed approach

Figure 4.20: Compare Greenspan's approach [53] with our proposed one. Note that Greenspan's method does not discriminate the three different types of edges in (a), thus gives poor enhancement results (b). By comparison, our approach (c) has much better results.

the inter- and intra-scale dependencies of wavelet coefficients [57]. It has been used for image denoising and texture classification [23], [58]. However, to minimize the complexity of models and learning processes, so far only a very small number of hidden states (usually less than 4) are assumed and a conditional independence model is used. These assumptions significantly limit the representation power of the HMT models. The performance of the wavelet HMT model for image enhancement is still under study but probably more sophisticated models need to be developed to relax the above constraints. Another issue with existing HMT-based enhancement approaches [55], [56] is that the authors assumed that the variances of wavelet coefficients decay exponentially from scale to scale. However, based on (4.7) we see that exponential decay applies only to edges with $\sigma = 0$ (i.e., singular edges, (4.8)). When $\sigma \neq 0$ the variance change across scales is not a simple exponential function of scale and the relation described by (4.7) should be used.

The main drawback of the proposed approach is its high complexity, since it uses the iterative Mallat-Zhong reconstruction algorithm. Future work has to address two aspects: (1) a reduction in complexity, to avoid the iterative reconstruction process by exploiting the inter-scale dependency of all of the coefficients, not just those at the maxima points; and (2) to develop the proposed approach to enhancement with upsampling, performing image interpolation from its decimated versions.

Chapter 5

Wavelet Statistics

In this chapter we study the statistics of the wavelet transform. This is crucial in many applications, such as image denoising, interpolation, segmentation and compression. In general, via some transformation, we hope to find better and/or more convenient statistical models than those of the original images. For example, the well-known Fourier transform completely de-correlates all wide-sense stationary processes. Thus the *optimal* Wiener filter [24] can be very efficiently ($O(N \log_2 N)$) implemented by simply processing each Fourier coefficient *independently*. In general, the optimal linear transform perfectly decorrelating a process is the Karhunen-Loeve transform (KLT). Unfortunately, the KLT is expensive to compute ($O(N^3)$) since it needs to compute eigenvalues and eigenvectors of large matrices. The wavelet transform has been observed as a good approximation to KLT for many typical statistical processes [65],[66],[12], and it can be implemented very efficiently: $O(N)$, even faster than the fast Fourier transform (FFT) [3].

As stated in Chapter 2, researchers have already established many powerful wavelet statistical models, such as the hidden markov tree (HMT) [22], multiscale MRF [67], Gaussian

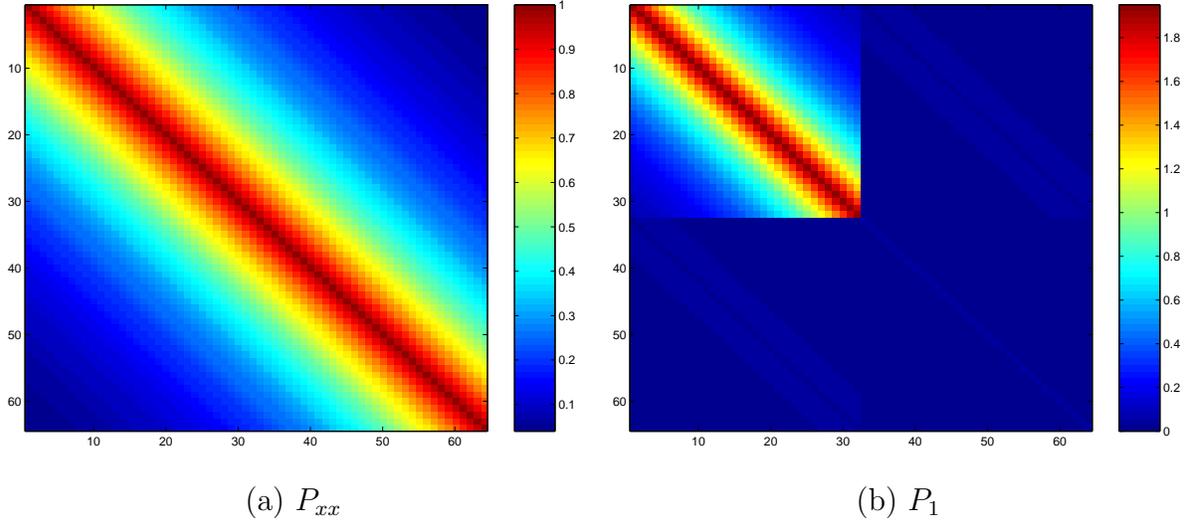


Figure 5.1: Autocovariance of (a) the 1-D Gaussian AR process and (b) its one-level Haar transform.

scale mixture (GSM) model [19] *etc.* However, these models were mainly inspired by the empirical observations of properties of the wavelet coefficients (e.g., low correlation, clustering, persistence). It should be noted that the statistical analysis of the wavelet coefficients is often hampered by the lack of explicit closed-form expressions of wavelet basis functions. Realizing such difficulties and also hoping to do more formal analysis, in this chapter we limit our study to understanding the statistics of first-order 1-D Gaussian auto-regressive (AR) models and separable 2-D Gaussian AR models, in the context of orthogonal wavelet transforms. AR models have been widely used in statistical signal processing [24],[68],[69], so our study is also meaningful in practice. Based on our analysis results of this chapter, we will design a wavelet-based *efficient and effective* Kalman filter (KF) for image denoising. This KF will be generalized for real-world video denoising in Chapter 7.

5.1 1-D Gaussian AR Process

The 1-D Gaussian AR process we are interested in is

$$x(n+1) = \alpha x(n) + w(n+1), \quad |\alpha| < 1 \quad (5.1)$$

where $w(n+1) \in N(0, \sigma_w^2)$ is an *i.i.d.* noise process and independent of $x(m)$, $m=1,2,\dots,n$.

Without loss of generality, we assume the variance of $x(n)$, σ_x^2 , to be 1. Thus, $\sigma_w^2 = 1 - \alpha^2$.

Assuming that (5.1) is in steady-state, the element (i, j) of the covariance matrix for the process is

$$P_{xx}(i, j) = \alpha^{|i-j|} \quad (5.2)$$

Fig. 5.1(a) displays the covariance matrix, as given in (5.2), for $\alpha = 0.95$ and $N = 64$ samples.

Consider taking an L -level wavelet transform of a set of samples ($\mathbf{x} \triangleq \{x(n), n=1,2,\dots,N\}$) of the process. We denote wavelet coefficients at level l , where $l = 1$ means the finest level and $l = L$ means the coarsest level, as $\mathbf{d}_l = \{d_l(n), n = 1, 2, \dots, N/2^l\}$, and the related scaling coefficients as $\mathbf{a}_l = \{a_l(n), n = 1, 2, \dots, N/2^l\}$. The low- and high-pass analysis filters are $h_0(n)$ and $h_1(n)$, respectively. We can show that the correlations among the wavelet coefficients decay at a rate much faster than the original process. For example, let us first consider the auto-covariance of \mathbf{d}_1 [70]:

$$\begin{aligned} P_{\mathbf{d}_1 \mathbf{d}_1}(n_1, n_2) &= E \left\{ \left[\sum_i h_1(2n_1 - i)x(i) \right] \left[\sum_j h_1(2n_2 - j)x(j) \right] \right\} \\ &= \sum_i \sum_j h_1(i)\alpha^{|2(n_1-n_2)-i+j|} h_1(j) \end{aligned} \quad (5.3)$$

where $E\{\cdot\}$ is the expectation function. An upper bound for the above covariance is

obtained by the summation of the absolute value of the term in (5.3):

$$P_{\mathbf{d}_1 \mathbf{d}_1}(n_1, n_2) \leq \sum_{i,j} |h_1(i)| \cdot |\alpha|^{2(n_1-n_2)-i+j} \cdot |h_1(j)| \quad (5.4)$$

Since $|\alpha| < 1$, we have the bounds

$$|\alpha|^{2(n_1-n_2)-i+j} \leq \begin{cases} |\alpha|^{2|n_1-n_2|-S+1} & , \quad 2|n_1 - n_2| \geq S - 1 \\ 1 & , \quad 2|n_1 - n_2| < S - 1 \end{cases} \quad (5.5)$$

for any n_1 and n_2 , where S is the support of the wavelet filter h_0 or h_1 . Since the right-hand side of (5.5) is no longer a function of i and j , we move it out of the summation in (5.4) to obtain the bound

$$P_{\mathbf{d}_1 \mathbf{d}_1}(n_1, n_2) \leq \begin{cases} \|h_1\|_{l_1}^2 \cdot |\alpha|^{2|n_1-n_2|-S+1} & , \quad 2|n_1 - n_2| \geq S - 1 \\ \|h_1\|_{l_1}^2 & , \quad 2|n_1 - n_2| < S - 1 \end{cases} \quad (5.6)$$

where $\|h\|_{l_1}$ represents the l_1 norm¹ [71], and, for the orthogonal wavelet transform, $\|h_0(n)\|_{l_1} = \|h_1(n)\|_{l_1}$. It is easy to show that $P_{\mathbf{a}_1 \mathbf{a}_1}(n_1, n_2)$ and $P_{\mathbf{a}_1 \mathbf{d}_1}(n_1, n_2)$ are also bounded by (5.6).

It is clear from (5.6) that the correlations of the wavelet coefficients of the 1-D Gaussian AR process decay much faster than the original process. Specifically, if the covariance of the original Gaussian process decays as $O(|\alpha|^{|\tau|})$, (where $\tau = n_1 - n_2$), then from (5.6) the covariances that result from a one-level transform ($P_{\mathbf{a}_1 \mathbf{a}_1}(n_1, n_2)$, $P_{\mathbf{a}_1 \mathbf{d}_1}(n_1, n_2)$ and $P_{\mathbf{d}_1 \mathbf{d}_1}(n_1, n_2)$) decay as $O(|\alpha|^{2|\tau|})$. By extension, for a m -level decomposition the covariances decay as $O(|\alpha|^{2^m |\tau|})$, where we use the same reasoning (5.3) to (5.6) and note that a m -level decomposition has a 2^m downsampling). So far, we limit the discussion to *intra*-scale covariance, however it is trivial to show that the *inter*-scale covariances of wavelet

¹The l_1 norm of a sequence is the sum of the absolute value of each element of the sequence.

coefficients also decay much faster than the original process. For example, consider a 2-level wavelet decomposition with wavelet coefficients $\mathbf{d}_1 = \{d_1(n), n = 1, 2, \dots, N/2^1\}$, $\mathbf{d}_2 = \{d_l(n), n = 1, 2, \dots, N/2^2\}$ and scaling coefficients $\mathbf{a}_2 = \{a_2(n), n = 1, 2, \dots, N/2^2\}$. Then, the cross-correlation between \mathbf{d}_1 and \mathbf{d}_2 is:

$$\begin{aligned} P_{\mathbf{d}_1\mathbf{d}_2}(n_1, n_2) &= E \left\{ \left[\sum_i h_1(2n_1 - i)x(i) \right] \left[\sum_j h_1^{(2)}(4n_2 - j)x(j) \right] \right\} \\ &= \sum_i \sum_j h_1(i)\alpha^{|2(n_1-n_2)-2n_2-i+j|} h_1^{(2)}(j) \end{aligned} \quad (5.7)$$

where $h_1^{(2)}(n) = h_1(n) * h_1(n)$ is the high-pass filter for level 2. Comparing (5.7) with (5.3), it is clear that $P_{\mathbf{d}_1\mathbf{d}_2}(n_1, n_2)$ decays at the same rate as $P_{\mathbf{d}_1\mathbf{d}_1}(n_1, n_2)$ (i.e., $O(|\alpha|^{2|\tau|})$).

Let us now consider taking a one-level Haar transform (i.e., $h_0(n) = [\sqrt{2}/2, \sqrt{2}/2]$, $h_1(n) = [-\sqrt{2}/2, \sqrt{2}/2]$) of \mathbf{x} . Using (5.6) we know that the covariances that result from a one-level Haar transform of the samples of the first-order Gauss AR process are upper bounded by:

$$\begin{cases} 2|\alpha|^{2|\tau|-1} & , \tau \neq 0 \\ 2 & , \tau = 0 \end{cases} \quad (5.8)$$

Specifically, from (5.3) the covariances are

$$P_{\mathbf{a}_1\mathbf{a}_1}(\tau) = \begin{cases} (\frac{1}{2} + \alpha + \frac{1}{2}\alpha^2)\alpha^{2|\tau|-1} & , \tau \neq 0 \\ 1 + \alpha & , \tau = 0 \end{cases} \quad (5.9)$$

$$P_{\mathbf{d}_1\mathbf{d}_1}(\tau) = \begin{cases} (-\frac{1}{2} + \alpha - \frac{1}{2}\alpha^2)\alpha^{2|\tau|-1} & , \tau \neq 0 \\ 1 - \alpha & , \tau = 0 \end{cases} \quad (5.10)$$

$$P_{\mathbf{a}_1\mathbf{d}_1}(\tau) = \begin{cases} -\frac{1}{2}\text{sgn}(\tau)(1 - \alpha^2)\alpha^{2|\tau|-1} & , \tau \neq 0 \\ 0 & , \tau = 0 \end{cases} \quad (5.11)$$

where

$$\text{sgn}(\tau) = \begin{cases} 1 & \tau \geq 0 \\ -1 & \tau < 0 \end{cases} \quad (5.12)$$

Putting \mathbf{a}_1 and \mathbf{d}_1 into a vector,

$$\kappa_1 \triangleq \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{d}_1 \end{bmatrix} \quad (5.13)$$

we call κ_1 the wavelet coefficient vector of level 1. Then the covariance matrix of κ_1 is

$$P_1 \triangleq E[\kappa_1 \kappa_1^T] = \begin{bmatrix} E[\mathbf{a}_1 \mathbf{a}_1^T] & E[\mathbf{a}_1 \mathbf{d}_1^T] \\ E[\mathbf{d}_1 \mathbf{a}_1^T] & E[\mathbf{d}_1 \mathbf{d}_1^T] \end{bmatrix} \quad (5.14)$$

which can be evaluated from (5.9)-(5.11). Fig. 5.1(b) displays P_1 for $\alpha = 0.95$. Note that the coarse scale information is displayed in the upper-left of Fig. 5.1(b). We can clearly see how the scaling coefficients \mathbf{a}_1 represent most energy of wavelet coefficients. The correlation structure can be interpreted more easily if P_1 is normalized by its variances, obtaining the correlation coefficients:

$$\Lambda_{\mathbf{a}_1 \mathbf{a}_1}(\tau) = \begin{cases} \frac{1+\alpha}{2} \alpha^{2|\tau|-1} & , \tau \neq 0 \\ 1 & , \tau = 0 \end{cases} \quad (5.15)$$

$$\Lambda_{\mathbf{d}_1 \mathbf{d}_1}(\tau) = \begin{cases} \frac{1-\alpha}{2} \alpha^{2|\tau|-1} & , \tau \neq 0 \\ 1 & , \tau = 0 \end{cases} \quad (5.16)$$

$$\Lambda_{\mathbf{a}_1 \mathbf{d}_1}(\tau) = \begin{cases} -\frac{1}{2} \text{sgn}(\tau) \sqrt{1-\alpha^2} \alpha^{2|\tau|-1} & , \tau \neq 0 \\ 1 & , \tau = 0 \end{cases} \quad (5.17)$$

The correlation-coefficient matrices are shown in Fig. 5.2. Several important points about this plot:

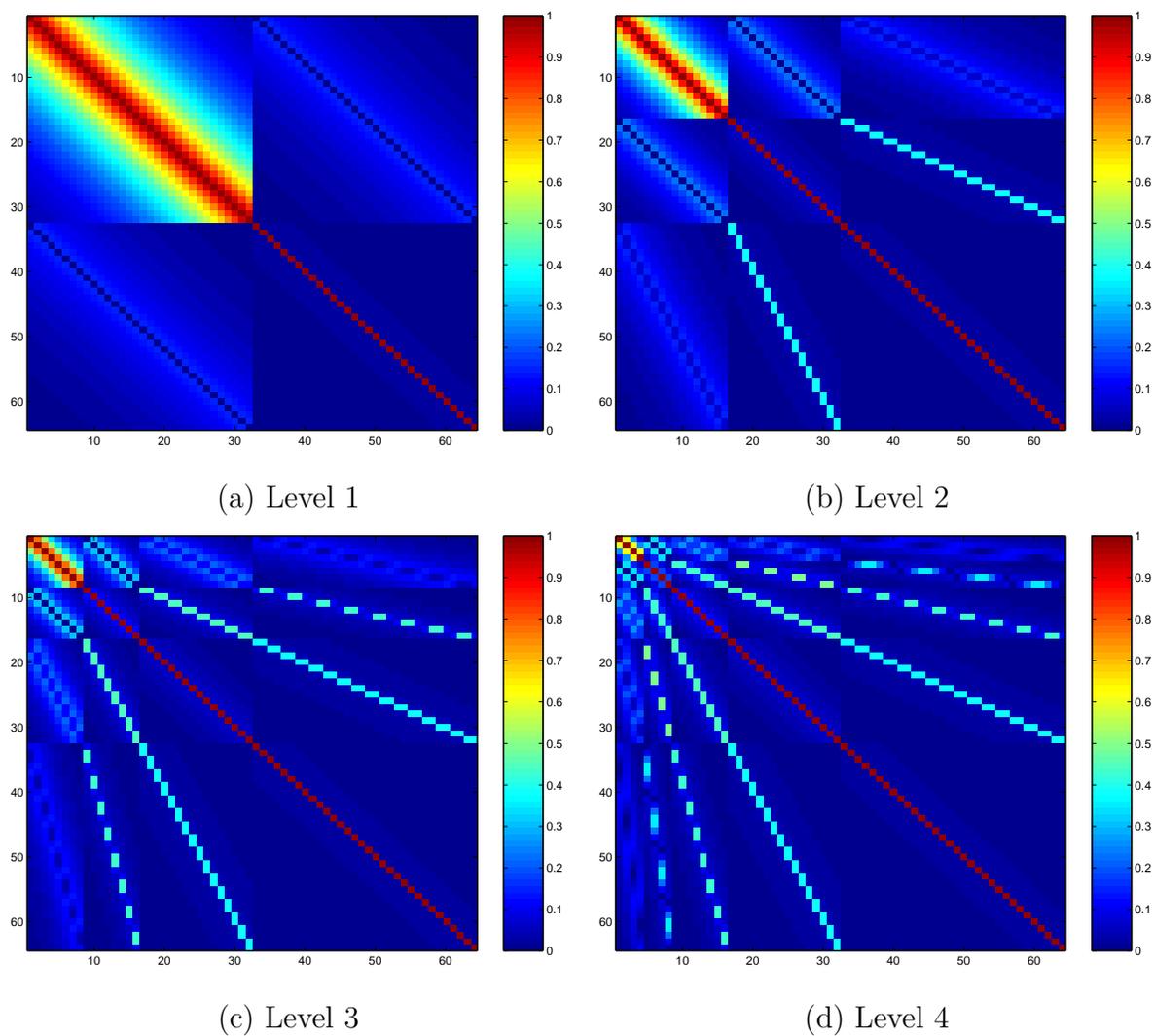
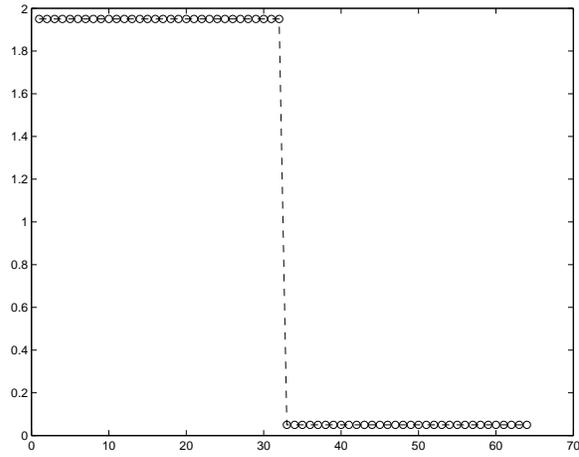
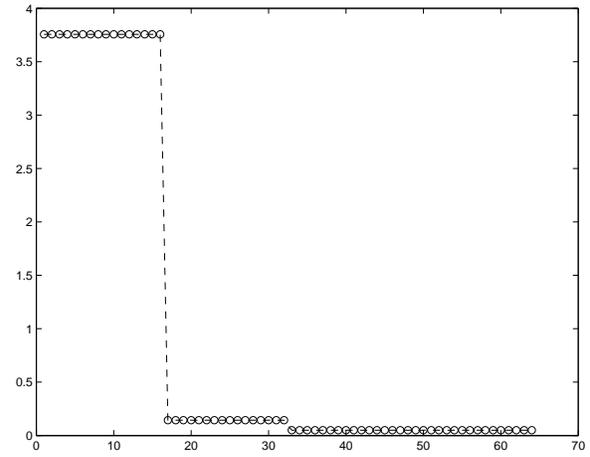


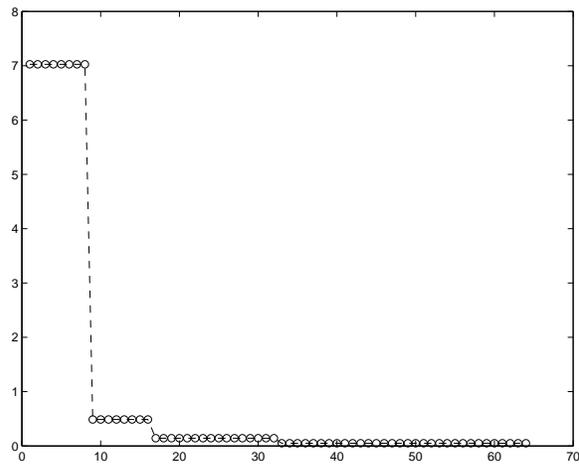
Figure 5.2: Correlation coefficient matrix of 4-level Haar transform of the Gaussian AR process $\alpha = 0.95$. Note the special structure of the remaining correlations.



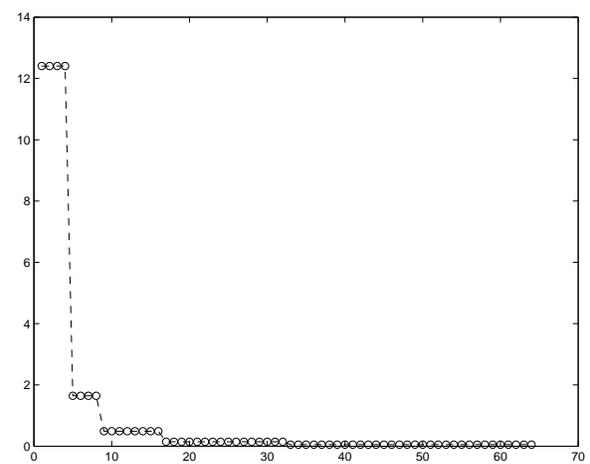
(a) Level 1



(b) Level 2



(c) Level 3



(d) Level 4

Figure 5.3: Variances of Haar transform of the Gaussian AR process $\alpha = 0.95$. Most energy concentrates in the approximation subband.

1. The correlation-coefficient matrix of the coarsest scale, $\Lambda_{\mathbf{a}_n, \mathbf{a}_n}^T$, has a dense structure, similar to that of the original process;
2. The correlation coefficients of the detail scales, $\Lambda_{\mathbf{d}_i, \mathbf{d}_i}^T$, concentrates on the diagonal line, and the corresponding matrix blocks are very sparse.

The sparse structure of $\Lambda_{\mathbf{d}_i, \mathbf{d}_i}$ means that most of the useful ‘information’ is contained in a small neighborhood. For example, from the estimation point of view, the MMSE estimate of the i th element, $\mathbf{d}_1(i)$, given the j th element is $\hat{\mathbf{d}}_1(i) = E\{\mathbf{d}_1(i)|\mathbf{d}_1(j)\}$.

The corresponding error variance is

$$\begin{aligned} E\{e^2(i)\} &= E(\mathbf{d}_1(i) - \hat{\mathbf{d}}_1(i))^2 \\ &= (1 - \lambda_{ij}^2)\sigma_{\mathbf{d}_1}^2(i) \end{aligned} \quad (5.18)$$

where λ_{ij} is the correlation coefficient between elements i, j , and $\sigma_{\mathbf{d}_1}^2(i)$ is the variance of element i . In (5.18) it is clear that the error variance ($E\{e^2(i)\}$) is completely determined by the correlation coefficient λ_{ij} : larger λ_{ij} yields smaller $E\{e^2(i)\}$. So λ_{ij} can be thought of as a measure of how much information element j has for element i . Interestingly, as we will show in the following denoising experiments the estimation error variance (MSE) from using only a small *neighborhood* information is very close to that of the optimal estimator (i.e., using *all* the wavelet coefficients).

3. The cross-correlation-coefficient matrix $\Lambda_{\mathbf{a}_1, \mathbf{d}_1}$ is also very sparse and concentrates on one band, corresponding to the elements with close spatial positions, but in neighboring levels.

Fig. 5.2 compares the correlation structure of the Haar transform for the 1- to 4-level cases. As we increase the number of scales we substantially increase the rate of decay.

However, we also introduce more undesired cross-correlation bands. These cross-correlation bands represent remaining information between subbands which needs to be considered when processing wavelet coefficients. There are two predominant approaches to handle this problem: one may design special wavelet filters to minimize cross correlation [70], however this turns out to be a difficult non-linear multi-variable optimization problem. Alternatively, one can estimate the cross-correlation from input data. Because these cross-correlation bands typically correspond to the same spatial locations in different subbands, tree-like models [57], [72] have been used to represent them.

Fig. 5.3 gives the variances of one to four-level Haar transform of the Gaussian AR process. As the number of scales increases more energies are compressed to the scaling coefficients. Combining the observations that the Haar transform can nearly decorrelate wavelet coefficients and compress energies we argue that it well approximates the action of the optimal Karhunen-Loeve transform for the first-order Gaussian AR process, an approximation that gets better as the number of scales increases. In fact, this observation is generally true for random processes which have a power spectrum concentrated at low frequencies (for example, $1/f$ processes [73]), since the wavelet transform effectively “zooms” in on low-frequency components by doing successive low-pass filtering and decimation. Besides the Haar wavelet, it is also interesting to see what happens with other wavelet transforms. We repeat the above experiments for the widely-used 4-tap (db2) and 8-tap (db4) Daubechies wavelets based on a periodic boundary extension.¹

Fig. 5.4 displays the correlation coefficients for one- to four-level Daubechies db2 transforms (we omit the plots for the correlation coefficients of the *db4* wavelet since they are

¹Periodic boundary extension means when wavelet filter runs out of finite-length data boundary, filtering is performed as if the data is periodically extended. This is one of the often used boundary extension modes.

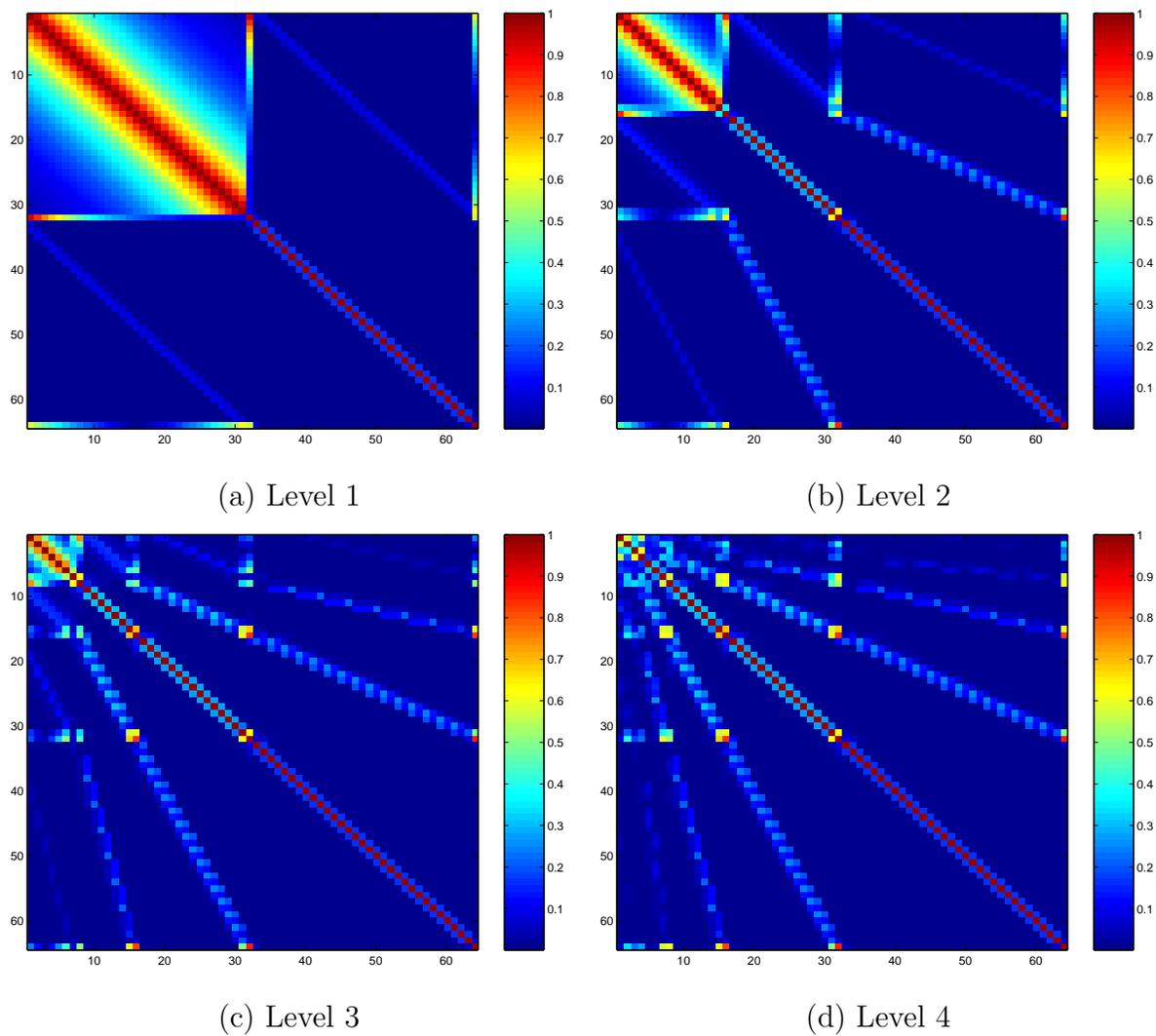


Figure 5.4: Correlation coefficient matrices of Daubechies *db2* transform of the first-order Gaussian AR process $\alpha = 0.95$. Note the strong correlation near the boundaries of each subband.

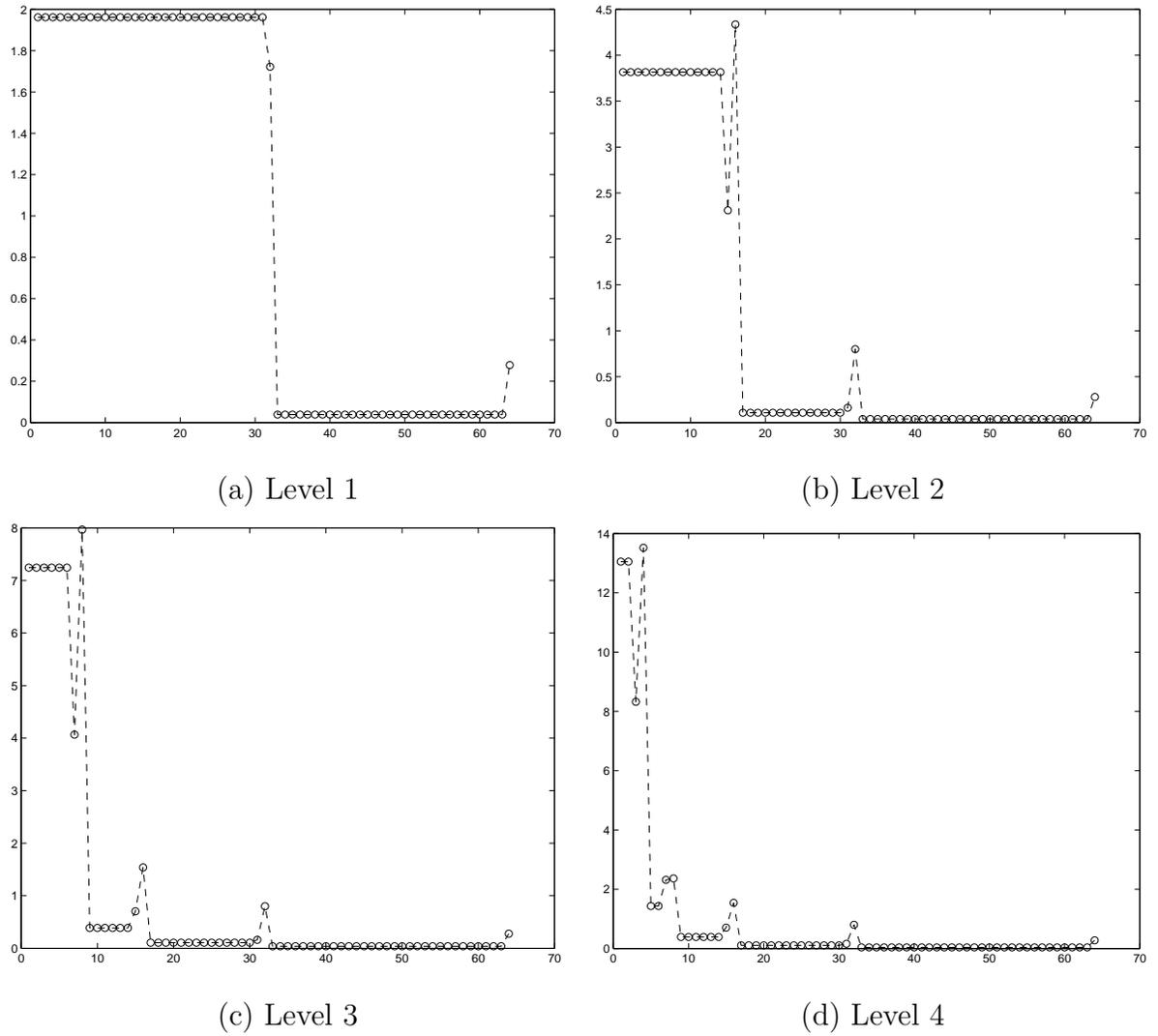
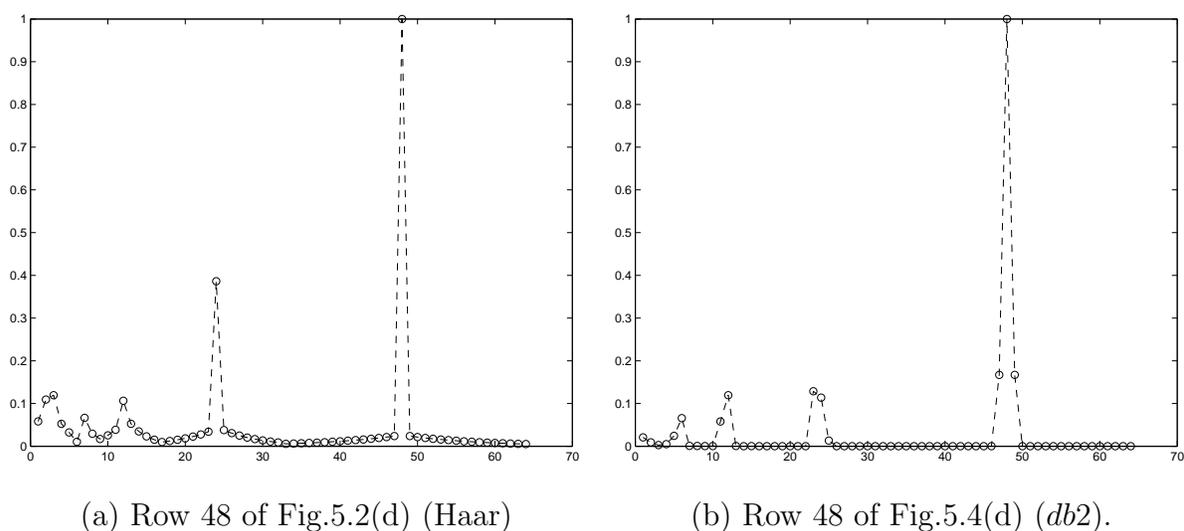


Figure 5.5: Variances of the *db2* transform of the Gaussian AR process $\alpha = 0.95$. Most of the energy of the coefficients is located in the low-frequency subband, however note the large energy at the boundary points.



(a) Row 48 of Fig.5.2(d) (Haar)

(b) Row 48 of Fig.5.4(d) (*db2*).

Figure 5.6: Comparing the correlation structure of Haar and *db2* wavelets using a profile of the correlation coefficients. We observe that *db2* has weaker inter-scale correlations (compare the magnitudes around abscissa 24) but stronger intra-scale correlations (compare the magnitudes at abscissa 47 or 49).

similar to those in Fig. 5.4). The corresponding variances are shown in Fig. 5.5. Comparing the correlation structures in Fig. 5.2 and Fig. 5.4 we can see at least two differences:

1. Roughly speaking, the inter-band correlations of the db2 wavelet are weaker than those of the Haar wavelet, whereas the intra-band correlations of the db2 are stronger. These can be seen most clearly from profiles of these figure, as shown in Fig. 5.6, which plot the correlation coefficients between coefficient 48 and all other coefficients. Note how the peak between abscissa 20 and 30 in Fig. 5.6(a) is much higher than that in Fig. 5.6(b). However, in Fig. 5.6(a), the correlation coefficient between intra-band elements 48 and 49 (also 48 and 47) is about 0.03 while it is about 0.2 in Fig. 5.6(b). This result means that for the Haar transform, information is scattered in several subbands, whereas db2 concentrates more information in one subband, a degree of concentration observed even more strongly for *db4*. Based on the measure of mutual information, Liu *et.al.* [74] had similar observations. They found that for wavelets with large vanishing moments (the vanishing moment of Haar and db4 wavelets is 1 and 4, respectively) the conditional mutual information for one coefficient from both its intra and inter-scale neighbors is very close to the conditional mutual information from just its intra-scale neighbors.
2. The db2 transform introduces boundary effects which do not exist with the Haar wavelet (compare Figs. 5.2 and 5.4). Boundary effects need to be handled properly because they can easily cause artifacts in the boundary areas.

From (5.3) we can see that after a one-level transform of the original stationary process $x(n)$, the detail signal \mathbf{d}_1 is still a stationary process. Similarly one can show that \mathbf{a}_1 is stationary and that \mathbf{a}_1 and \mathbf{d}_1 are jointly stationary. It is straightforward to generalize this

result to multi-level transforms. That is, for a L-level wavelet transform of stationary $x(n)$ with resulting coefficients $\{[\mathbf{a}_i \ \mathbf{d}_i]^T, i = 0, 1, \dots, L\}$, \mathbf{a}_i and \mathbf{d}_i are still stationary processes, and \mathbf{a}_i and \mathbf{d}_i are jointly stationary. However, coefficients at different levels are not jointly stationary. In general, we have

$$P_{\mathbf{ad}}(n_1, n_2) = P_{\mathbf{ad}}(n_1 - 2^{-j+i}n_2) \quad (5.19)$$

and similarly for $P_{\mathbf{aa}}$ and $P_{\mathbf{ad}}$. So if $i \neq j$ then $P_{\mathbf{ad}}(n_1, n_2)$ depends on specific values of n_1 and n_2 , not a function of just their difference $n_1 - n_2$. This loss of stationarity can also be seen visually by inspecting the off-diagonal matrix blocks in Fig. 5.2 and 5.4.

To summarize the above results, we have shown that, for first-order 1-D AR processes, the wavelet coefficients decay much faster than the original process, most of the energy in the wavelet domain is compressed to a few scaling coefficients, implying that the wavelet transform can be thought of as a good approximate to the optimal Karhunen-Loeve transform. We also showed that if the original process is stationary, each subband remains stationary, however the wavelet tree structure causes joint stationarity across levels to be lost.

Before concluding this section, let us look at two examples of *nonstationary* processes since most natural signals are not stationary. We slightly modify the above stationary process $x(n)$ to construct two new nonstationary processes $y(n)$ and $z(n)$:

$$y(n) = \begin{cases} x(n) & n < N \\ x(n) + \mu & n \geq N \end{cases} \quad (5.20)$$

and

$$z(n) = \begin{cases} x(n) & n < N \\ \sigma \cdot x(n) & n \geq N \end{cases} \quad (5.21)$$

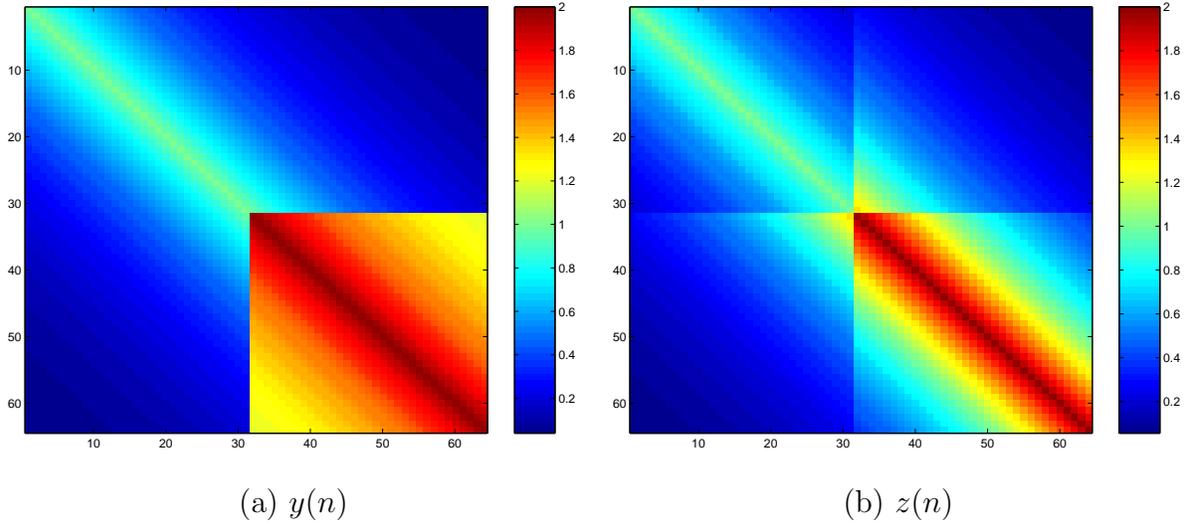


Figure 5.7: Correlation matrices of two nonstationary processes defined in (5.20) and (5.21), respectively. $y(n)$ has a change of mean value at the abscissa 32. $z(n)$ has a change of variance at the same location.

where μ and σ are parameters of mean and standard deviation, respectively, and N is specified. $y(n)$ can be seen as a model of an ideal step edge. Note $y(n)$ has non-zero mean values for $n > N - 1$. This is different from the previously discussed process $x(n)$, which is a zero-mean process. $z(n)$ is also different from $x(n)$ in that $x(n)$ has a constant variance whereas the variances of $z(n)$ change at $n = N$. Therefore, we intentionally generate two nonstationary processes. We want to see the statistics of their wavelet coefficients.

Fig. 5.7 plots the correlation matrices of the non-stationary processes $y(n)$ and $z(n)$. Fig. 5.8 and Fig. 5.9 show the correlation coefficients of one to four-level Haar transforms of $y(n)$ and $z(n)$, respectively, where $N = 32$, $\mu = 1$ and $\sigma = \sqrt{2}$. Comparing Figs. 5.8 and 5.9 with Fig. 5.2 we can see that the non-stationarities introduce strong correlations at locations where changes happen. Such correlations might cause so-called Gibbs-like

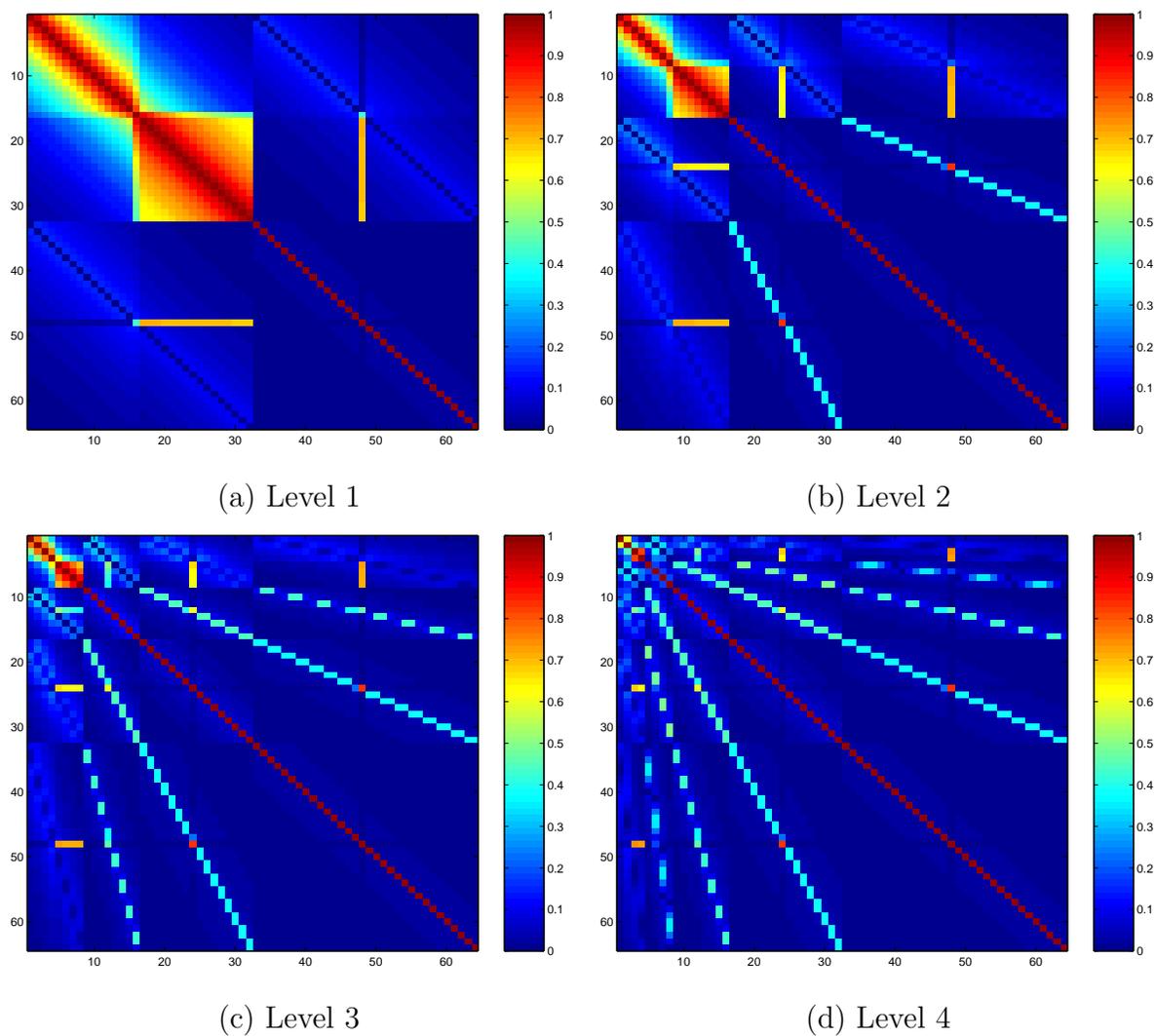


Figure 5.8: Correlation coefficient matrix of Haar transform of the non-stationary process $y(n)$ ($\alpha = 0.95$, $\mu = 1$). $y(n)$ models an ideal edge signal. Note the non-stationarity causes strong correlation near the change points.

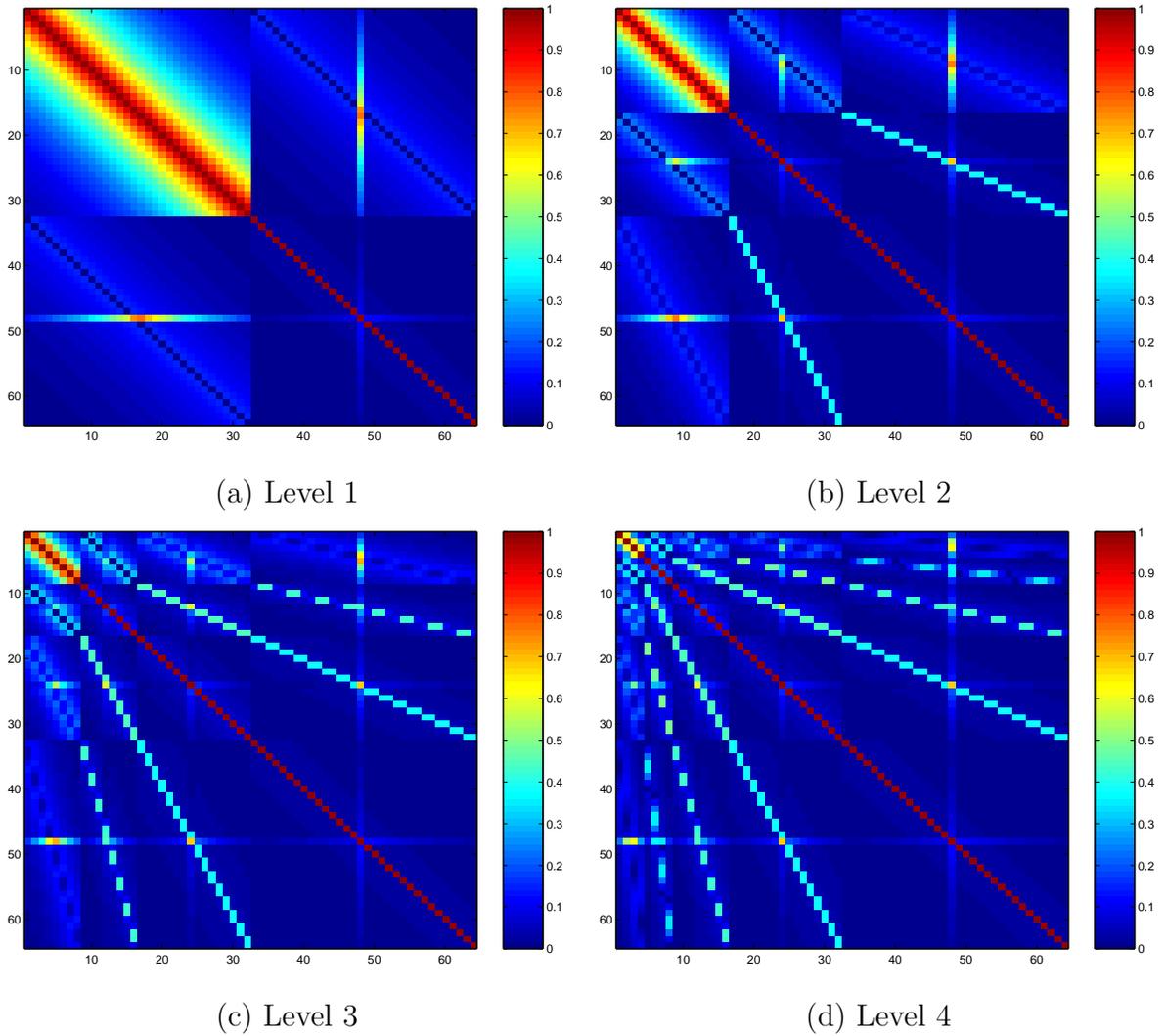


Figure 5.9: Correlation coefficient matrix of Haar transform of the non-stationary process $z(n)$ ($\alpha = 0.95$, $\sigma = \sqrt{2}$). Like Fig. 5.8, the non-stationarity causes strong correlation near the change points.

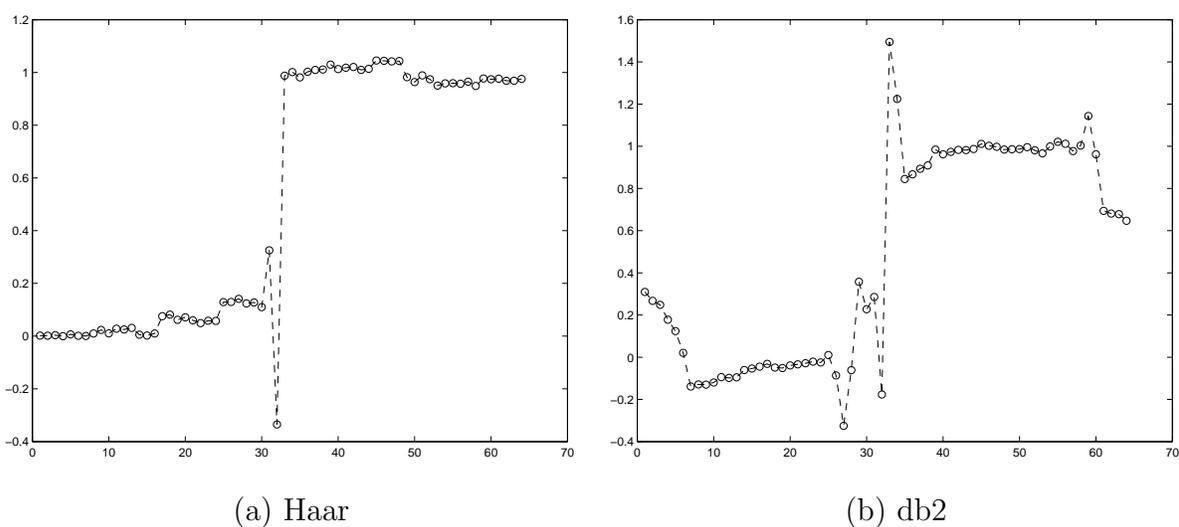


Figure 5.10: Synthesized $y(n)$ with the wavelet coefficients assumed *independent*. Note the Gibbs-like artifact around the discontinuity points due to neglecting remaining correlation in the four-level Haar and db2 transforms. In (b) we can also see artifacts at the boundary. This is due to neglecting the strong correlation at boundaries as we mentioned before.

artifacts if they are not properly taken care of. For example, if we neglect the remaining correlations and treat the wavelet coefficients as being independent of each other then the synthesized signals of $y(n)$ from the four-level Haar and db2 transforms look like those in Fig. 5.10. Note Gibbs-like artifacts occurs around the discontinuity locations (at abscissa 32). Also, the artifact with db2 is more spread than Haar. In real-world signals/images, there are many sharp changes like those in $y(n)$ or $z(n)$. So strong correlations exist in their wavelet coefficients, which can not be neglected. In Chapter 6, which considers image denoising, we discuss a new approach to handle this problem. In Fig. 5.10(b) we can also see the artifacts due to ignoring correlation around boundary (around abscissa 0 and 64).

We have shown that the wavelet transform can well approximate the optimal KLT for 1-D first-order stationary AR signals. In the next section we generalize this study to a practical 2-D process.

5.2 2-D Gaussian Process

We wish to study the correlation structures of a typical 2-D Gaussian process in the wavelet domain. The process we are interested in is the first-order non-symmetric half plane (NSHP) AR model (Fig. 5.11) [75],[69].

For the 2-D wavelet transform, we use the separable orthogonal wavelet transform. This type of 2-D wavelet transforms have been used widely due to its simplicity and effectiveness [9],[20],[76]. The separable 2-D wavelet bases are constructed by the Kronecker product

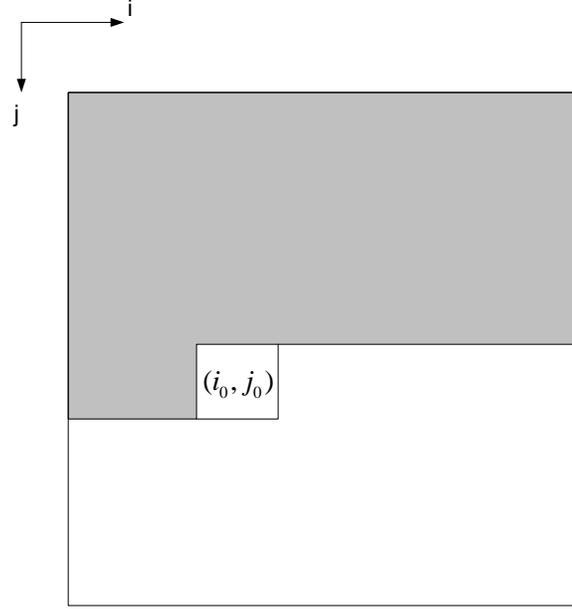


Figure 5.11: Non-symmetric half plane of the current pixel (i_0, j_0) is the shaded area. \times denotes the current pixel.

(tensor product) of a 1-D low-pass filter $h_0(n)$ and a high-pass filter $h_1(n)$:

$$\begin{aligned}
 h_a(n_1, n_2) &= h_0(n_1) \cdot h_0(n_2) \\
 h_h(n_1, n_2) &= h_0(n_1) \cdot h_1(n_2) \\
 h_v(n_1, n_2) &= h_1(n_1) \cdot h_0(n_2) \\
 h_d(n_1, n_2) &= h_1(n_1) \cdot h_1(n_2)
 \end{aligned} \tag{5.22}$$

The first-order NSHP AR model for image has been used by several authors [68][75][69][77]. This model can be described in the following form: for the pixel at spatial coordinate (i, j) we have the dynamic model

$$x(i, j) = \rho_v x(i-1, j) + \rho_h x(i, j-1) - \rho_h \rho_v x(i-1, j-1) + w(i, j), \quad |\rho_h| < 1, |\rho_v| < 1 \tag{5.23}$$

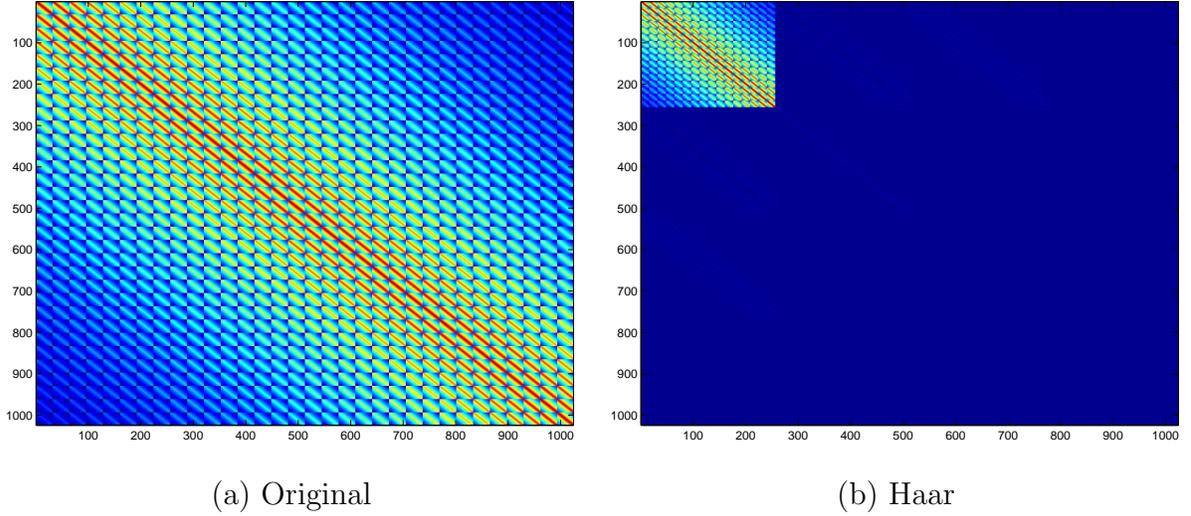


Figure 5.12: Covariances of the NSHP process and its one-level Haar transform (for a 32×32 image). Note that (b) is a much sparser matrix with large elements (in magnitude) concentrating in the top-left portion.

where $w(i, j) \in N(0, \sigma_w^2)$ is white Gaussian noise and where

$$w(i, j) \text{ is independent of } x(k, l) \quad k < i \text{ or } l < j. \quad (5.24)$$

For the steady state, if we set $\sigma_w^2 = (1 - \rho_h^2)(1 - \rho_v^2)$, then the variance of $x(i, j)$, σ_x^2 , is 1. One can verify that (5.23) defines a stationary process and that the covariance between $x(i_1, j_1)$ and $x(i_2, j_2)$ is

$$r(i_1, j_1, i_2, j_2) = r(\tau_1, \tau_2) = \rho_h^{|\tau_1|} \rho_v^{|\tau_2|} \quad (5.25)$$

where $\tau_1 = i_1 - i_2$ and $\tau_2 = j_1 - j_2$.

Let us denote the wavelet coefficients of a one-level 2-D separable orthogonal wavelet transform of $x(i, j)$ by $a_1(i, j)$, $h_1(i, j)$, $v_1(i, j)$ and $d_1(i, j)$. We can compute the covariance

between $d_1(i_1, j_1)$ and $d_1(i_2, j_2)$ as (note we use the separable 2-D transform):

$$\begin{aligned}
P_{d_1 d_1}(\tau_1, \tau_2) &= E \left\{ \sum_{n_1} h_d(i_1, 2j_1 - n_1) \sum_{m_1} h_d(2i_1 - m_1, j_1) x(m_1, n_1) \cdot \right. \\
&\quad \left. \sum_{n_2} h_d(i_2, 2j_2 - n_2) \sum_{m_2} h_d(2i_2 - m_2, j_2) x(m_2, n_2) \right\} \\
&= \sum_{m_1, m_2} h_d(m_1) h_d(m_2) \rho_h^{|2(i_1 - i_2) - m_1 + m_2|} \cdot \sum_{n_1, n_2} h_d(n_1) h_d(n_2) \rho_v^{|2(j_1 - j_2) - n_1 + n_2|} \\
&= \sum_{m_1, m_2} h_d(m_1) h_d(m_2) \rho_h^{|2\tau_1 - m_1 + m_2|} \cdot \sum_{n_1, n_2} h_d(n_1) h_d(n_2) \rho_v^{|2\tau_2 - n_1 + n_2|}
\end{aligned} \tag{5.26}$$

where $\tau_1 = i_1 - i_2$ and $\tau_2 = j_1 - j_2$. Following the derivation for the 1-D case (see (5.4), (5.5)), we can find a bound of $P_{d_1 d_1}(\tau_1, \tau_2)$ as

$$P_{\mathbf{d}_1 \mathbf{d}_1}(\tau_1, \tau_2) \leq \begin{cases} \|h_d\|_{l_1}^4 |\rho_h|^{2|\tau_1| - S + 1} |\rho_v|^{2|\tau_2| - S + 1} & (2|\tau_1| \geq S - 1), \quad (2|\tau_2| \geq S - 1) \\ \|h_d\|_{l_1}^4 |\rho_h|^{2|\tau_1| - S + 1} & (2|\tau_1| \geq S - 1), \quad (2|\tau_2| < S - 1) \\ \|h_d\|_{l_1}^4 |\rho_v|^{2|\tau_2| - S + 1} & (2|\tau_1| < S - 1), \quad (2|\tau_2| \geq S - 1) \\ \|h_d\|_{l_1}^4 & (2|\tau_1| < S - 1), \quad (2|\tau_2| < S - 1) \end{cases} \tag{5.27}$$

Two observations of (5.27):

1. Like 1-D case, the covariances of the wavelet coefficients decay at a much faster rate than the original process. For example, if the original covariance decays at $O(\tau_h)$ (or $O(\tau_v)$) then the wavelet coefficients decay at $O(\tau_h^2)$ (or $O(\tau_v^2)$).
2. Like the original process, $c_{d_1}(i, j)$ is still a stationary process.

It is trivial to show that the above observations also hold for the processes $a_1(i, j)$, $h_1(i, j)$, $v_1(i, j)$. In addition, all these processes are cross-stationary.

The above discussion was limited to one scale. However, following the same reasoning as the 1-D case, for multi-level decompositions one can show at least two more things: (1) the cross-covariance of 2-D wavelet coefficients at different levels also decays much faster than the original processes; (2) the wavelet coefficients at different levels are not cross-stationary (see (5.7)).

Specifically, let us take a one-level 2-D Haar transform of $x(i, j)$. From (5.27) we know that the covariance is bounded by

$$P_{\mathbf{d}_1 \mathbf{d}_1}(\tau_1, \tau_2) \leq \begin{cases} 4\rho_h^{2|\tau_1|-1} \rho_v^{2|\tau_2|-1} & (\tau_1 \neq 0), (\tau_2 \neq 0) \\ 4\rho_h^{2|\tau_1|-1} & (\tau_1 \neq 0), (\tau_2 = 0) \\ 4\rho_v^{2|\tau_2|-1} & (\tau_1 = 0), (\tau_2 \neq 0) \\ 4 & (\tau_1 = 0), (\tau_2 = 0) \end{cases} \quad (5.28)$$

The covariance of $\mathbf{d}_1(i, j)$ is

$$P_{\mathbf{d}_1 \mathbf{d}_1}(\tau_1, \tau_2) = \begin{cases} (-\frac{1}{2} + \rho_h - \frac{1}{2}\rho_h^2)(-\frac{1}{2} + \rho_v - \frac{1}{2}\rho_v^2)\rho_h^{2|\tau_1|-1} \rho_v^{2|\tau_2|-1} & (\tau_1 \neq 0), (\tau_2 \neq 0) \\ (-\frac{1}{2} + \rho_h - \frac{1}{2}\rho_h^2)(1 - \tau_v) & (\tau_1 \neq 0), (\tau_2 = 0) \\ (1 - \tau_h)(-\frac{1}{2} + \rho_v - \frac{1}{2}\rho_v^2) & (\tau_1 = 0), (\tau_2 \neq 0) \\ (1 - \tau_h)(1 - \tau_v) & (\tau_1 = 0), (\tau_2 = 0) \end{cases} \quad (5.29)$$

Putting all wavelet coefficients into a vector

$$\kappa_1 \triangleq \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{v}_1 \\ \mathbf{h}_1 \\ \mathbf{d}_1 \end{bmatrix} \quad (5.30)$$

the covariance matrix $P_1 = E(\kappa_1 \kappa_1^T)$ of the Haar transform (for a 32×32 $x(i, j)$) is shown

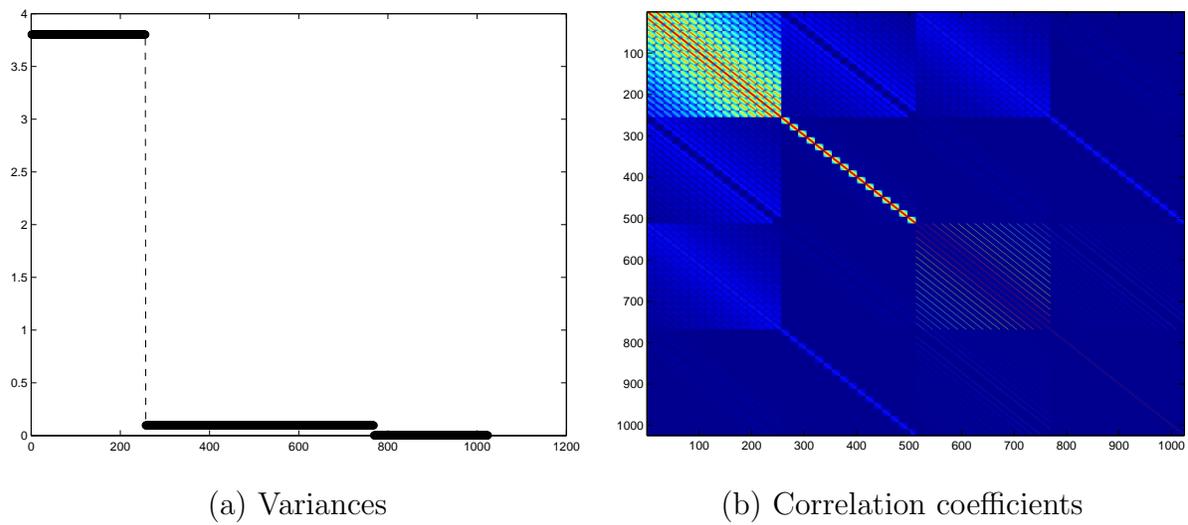


Figure 5.13: (a) Variances of the one-level 2-D Haar transform of the NSHP process. Note most energy concentrates in the low-frequency subband. (b) Correlation coefficients of the one-level 2-D Haar transform of the NSHP process. This is a very sparse matrix with large elements mainly on the main diagonal.

in Fig. 5.12(b). For comparison, Fig. 5.12(a) shows the covariance of the original process.

We have the following observations:

1. The energy of the wavelet coefficients (κ_1) concentrates in the coarse-scale component (i.e., \mathbf{a}_1). This is more obvious in the plot of variances of κ_1 (Fig. 5.13(a)). Gallager [78] found that for Gaussian random variables, information is proportional to the logarithm of the variance of the random variable. Following [78] we can say the information of the wavelet coefficients of the first-order NSHP Gaussian process concentrates in coarse scale.
2. Looking at Fig. 5.13(b) we can see that the correlation coefficient matrix of $E(\mathbf{a}_1\mathbf{a}_1^T)$ is a dense matrix, with a structure similar to the original process. However, the rest of the autocorrelation coefficient matrices (i.e., $E(\mathbf{v}_1\mathbf{v}_1^T)$, $E(\mathbf{h}_1\mathbf{h}_1^T)$ and $E(\mathbf{d}_1\mathbf{d}_1^T)$) have very sparse structures with large correlation coefficients (in magnitude) only in the neighborhood of the diagonal elements. This special structure reflects that the wavelet coefficients have localized intrascale correlations.
3. The cross-correlation coefficients (e.g., $E(\mathbf{a}_1\mathbf{v}_1^T)$, $E(\mathbf{v}_1\mathbf{h}_1^T)$ and $E(\mathbf{h}_1\mathbf{d}_1^T)$) have small magnitudes and are typically sparse. However, these coefficients are not zeros meaning there are remaining correlations between subbands on the same level.

For comparison, we further give the correlation coefficients of the *two-level* 2-D Haar transforms in Fig. 5.14(b). A five-level 2-D 4-tap Daubechies (*db2*) transform is also given in Fig. 5.16(a), where we show the boundary effects. For comparison, we show in Fig. 5.16(b) the five-level 2-D Haar transform, which is free of boundary effects.

Looking at Figs. 5.14 and 5.16 we have three new observations:

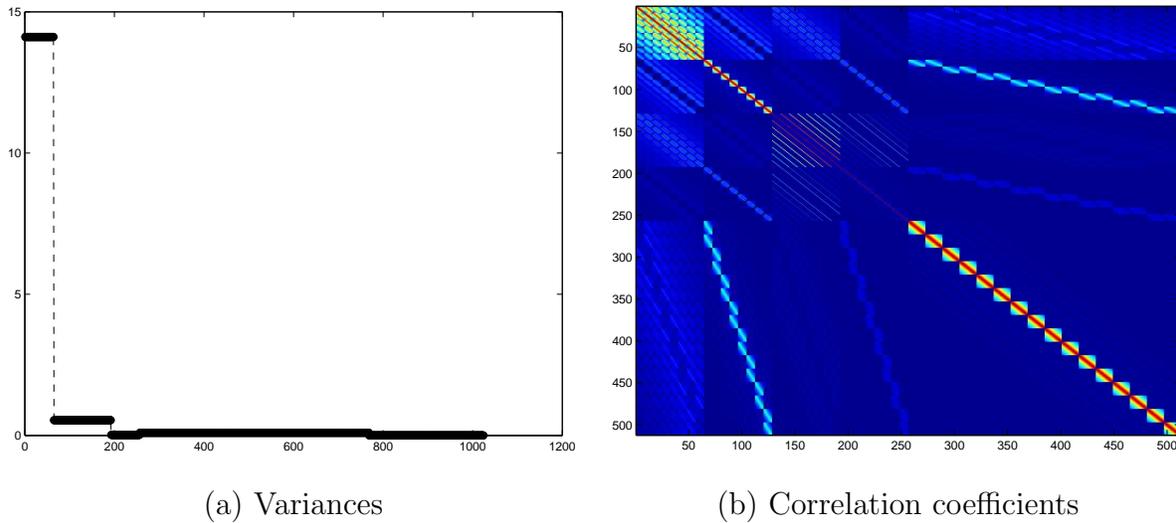


Figure 5.14: (a) Variances of the two-level Haar transform of the NSHP process; (b) Corresponding correlation coefficients. The ordering of wavelet coefficients is $[\mathbf{a}_2, \mathbf{v}_2, \mathbf{h}_2, \mathbf{d}_2, \mathbf{v}_1, \mathbf{h}_1, \mathbf{d}_1]^T$.

1. When increasing transform levels the correlation coefficient matrices get even sparser with more energy concentrating in the coarsest levels (see Fig. 5.14(a) and compare with Fig. 5.13(a)).
2. Besides the above localized intrascale correlations we can also see the strong parent-child (interscale) correlations. For 2-D wavelet transforms the parent-children relationship is shown in Fig. 5.15. It includes 5 coefficients (one parent and four children) corresponding to same spatial location, in two subsequent levels of same orientations.
3. In Fig. 5.16(a) boundary discontinuities cause strong correlations among wavelet coefficients.

We have studied the statistics of two typical Gaussian processes (first-order 1-D AR

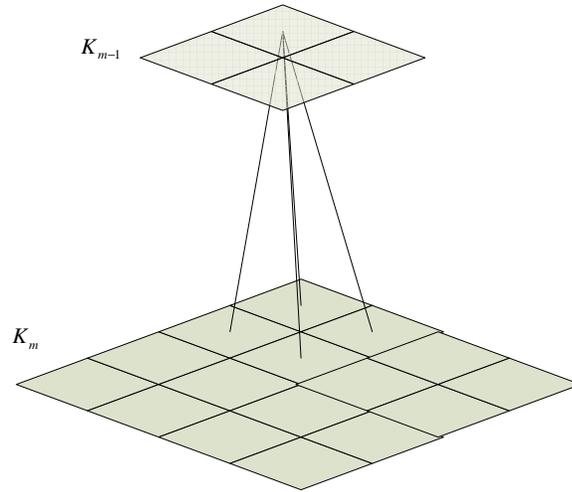


Figure 5.15: Parent-children relationship. One parent coefficient in level K_{m-1} has four children in level K_m .

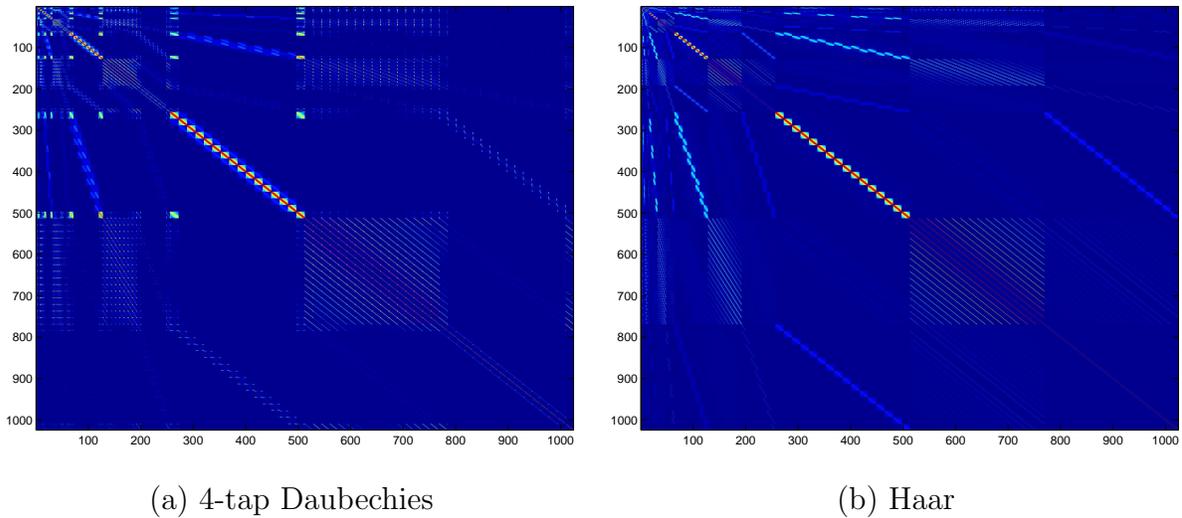


Figure 5.16: Correlation coefficients of 5-level (a) 4-tap Daubechies (*db2*) transform and (b) Haar transform. Note (a) has a few large elements (in magnitude) around boundaries of each subband, which does not exist in (b).

process and first-order 2-D NSHP process) in the wavelet domain. To summarize, wavelet transforms yield sparse correlation structures; concentrate most energy of the wavelet coefficients into the coarse scales; have remaining intrascale and interscale correlations with special structures (i.e., intrascale correlation is localized around diagonal elements; interscale correlation mainly exists among parent and children). These properties provide some promises for the wavelet-based signal/image processing. However, it is still not very clear how these properties of the wavelet transform can benefit subsequent processing and how well the wavelet-based processors perform. In the next section we exploit the above properties by designing a wavelet-based strip Kalman filter for image denoising. We will look at the advantages and disadvantages of wavelet-based processing.

5.3 Wavelet-based 2-D Strip Kalman Filtering

As shown in Chapter 2, the 2-D Kalman filter (KF) is usually implemented in an approximate form: the strip KF [26], to lower computational complexity. In this section we look at the wavelet-based strip KF and compare it with the spatial strip KF. It is difficult to discuss this problem in a general way. Rather, we focus our discussion on the NSHP image model (5.23). Firstly, we show the NSHP model can be characterized by a dynamic state equation, as required by the optimal KF. We then find optimal strip structures under certain complexity constraints. Spatial and wavelet-based strip KF's will be compared in terms of the relative increase of MSE for image denoising:

$$\delta_{MSE} = \frac{MSE_{strip} - MSE_{optimal}}{MSE_{optimal}} (\%) \quad (5.31)$$

where $MSE_{optimal}$ and MSE_{strip} denote the MSE of the optimal and strip KF's, respectively.

5.3.1 Dynamic State Equations of the NSHP Model and Strip Filter

To perform Kalman filtering, we need a dynamic state equation to describe the underlying process. In this subsection, we mainly derive the dynamic state equation for the NSHP image model.

Repeat (5.23):

$$x(i, j) = \rho_v x(i-1, j) + \rho_h x(i, j-1) - \rho_h \rho_v x(i-1, j-1) + w(i, j), \quad i, j = 1, 2, \dots, N$$

Equivalently, we have

$$x(i, j) - \rho_v x(i-1, j) = \rho_h x(i, j-1) - \rho_h \rho_v x(i-1, j-1) + w(i, j) \quad (5.32)$$

We then divide the 2-D problem into columns:

$$\begin{aligned} \mathbf{x}(j) &= [x(1, j), x(2, j), \dots, x(N, j)]^T \\ \mathbf{w}(j) &= [w(1, j), w(2, j), \dots, w(N, j)]^T \quad j = 1, 2, \dots, N \end{aligned} \quad (5.33)$$

which allows (5.32) to be written as

$$[0 \ 0 \dots \ -\rho_v \ 1 \ 0 \dots \ 0] \mathbf{x}(j) = \rho_h [0 \ 0 \dots \ -\rho_v \ 1 \ 0 \dots \ 0] \mathbf{x}(j-1) + [0 \ 0 \dots \ 0 \ 1 \ 0 \dots \ 0] \mathbf{w}(j) \quad (5.34)$$

A vector equation like (5.34) can be written for each value of $j = 2, 3, \dots, N$. However, some caution needs to be excised when $i = 1$. At the top border of the image, i.e., when $i = 1$, (5.32) reduces to the following:

$$x(1, j) = \rho_h x(1, j-1) + w(1, j), \quad j = 2, 3, \dots, N. \quad (5.35)$$

Combining (5.35) and (5.34), we get the following vector-matrix form:

$$A\mathbf{x}(j) = \rho_h A\mathbf{x}(j-1) + \mathbf{w}(j) \quad (5.36)$$

where A is the $N \times N$ matrix

$$\begin{pmatrix} 1 & 0 & & \dots & 0 \\ -\rho_v & 1 & 0 & & 0 \\ 0 & -\rho_v & 1 & 0 & \dots & 0 \\ \cdot & & & & & \cdot \\ \cdot & & & & & \cdot \\ \cdot & & & & & \cdot \\ 0 & \dots & & -\rho_v & 1 \end{pmatrix}$$

The matrix A , being a lower triangular Toeplitz matrix, is invertible. Pre-multiplying by A^{-1} on both sides of (5.36), we get

$$\mathbf{x}(j) = \rho_h \mathbf{x}(j-1) + A^{-1}\mathbf{w}(j) \quad (5.37)$$

which is the state-space equation of the first-order NSHP image models. It is causal and hence suitable for recursive linear estimation by Kalman filtering (KF).

Since we will perform image denoising, let the noisy measurements be given by

$$y(i, j) = x(i, j) + v(i, j), \quad i, j = 1, 2, \dots, N.$$

where $v(i, j) \sim N(0, \sigma_v^2)$, $i, j = 1, 2, \dots, N$, is an *i.i.d* process and independent of $x(i, j)$. If we arrange all the observations and noise in a column into corresponding vectors, forming $\mathbf{y}(j)$, $\mathbf{v}(j)$ consistent with (5.33) then the sequence of observed vectors can be expressed as

$$\mathbf{y}(j) = \mathbf{x}(j) + \mathbf{v}(j), \quad j = 1, 2, \dots, N, \quad (5.38)$$

which is our measurement equation.

To perform KF in the wavelet domain, we need analogous state and measurement equations like (5.37) and (5.38). It should be emphasized that we perform *1-D* wavelet transform on each image column sequentially, instead of a *2-D* wavelet transform on the whole image. This is quite different from the standard image denoising structure [9],[79], which uses a 2-D wavelet transform. The reason is twofold: (1) as shown below, the *1-D* wavelet coefficients still obey a dynamic model like (5.37), which allows Kalman filtering; (2) For our video denoising problem in Chapter 7, doing wavelet transform sequentially can minimize system latency. Using H to represent the wavelet transformation matrix, from (5.37) we have

$$\mathbf{x}_H(j) = \rho_h \mathbf{x}_H(j-1) + A_H^{-1} \mathbf{w}_H(j), \quad j = 1, 2, \dots, N, \quad (5.39)$$

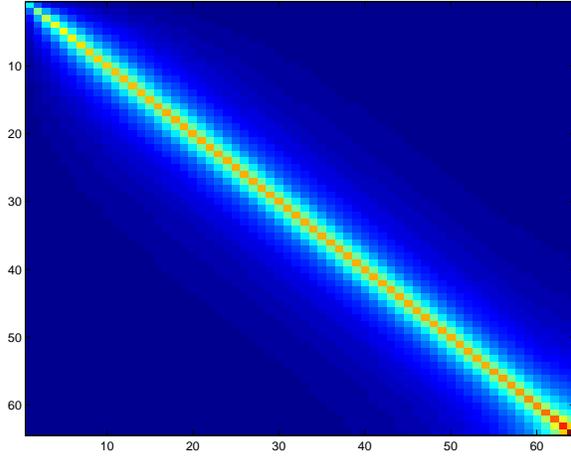
the state equation in the wavelet domain, where $\mathbf{x}_H = H\mathbf{x}$ and $\mathbf{w}_H = H\mathbf{w}$ are the wavelet coefficients of \mathbf{x} and \mathbf{w} respectively, and $A_H^{-1} = HA^{-1}H^T$. If the wavelet H is chosen to be orthonormal H , then \mathbf{w}_H is still an i.i.d. process like \mathbf{w} . Note that (5.39) is also an auto-regressive model like its spatial counterpart (see (5.37)), meaning that the optimal KF can be achieved in the wavelet domain.

Similarly, we have the wavelet domain measurement equation from (5.38):

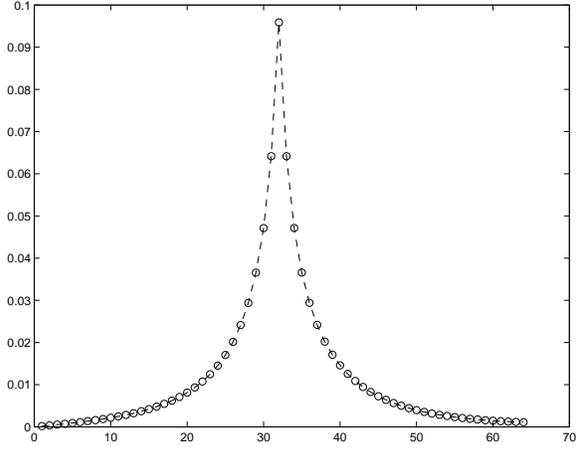
$$\mathbf{y}_H(j) = \mathbf{x}_H(j) + \mathbf{v}_H(j), \quad j = 1, 2, \dots, N. \quad (5.40)$$

where $\mathbf{y}_H = H\mathbf{y}$ and $\mathbf{v}_H = H\mathbf{v}$. As before, \mathbf{v}_H is i.i.d if H is orthonormal.

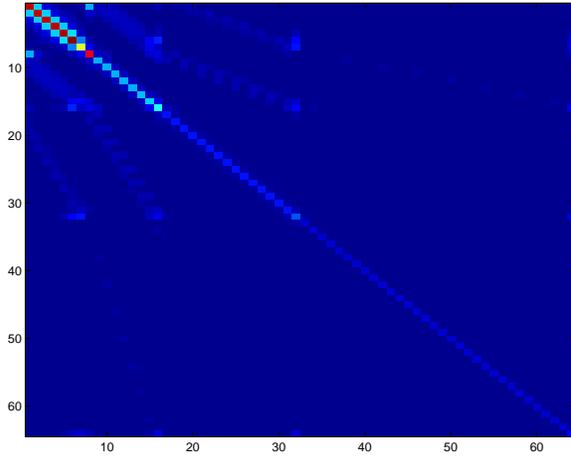
We have derived the state and measurement equations of the NSHP model ((5.37) and (5.38)). Thus, in principle, the *optimal* KF can be applied for estimation. However, in practice low-complexity *sub-optimal* KFs are often used, which we study here for the strip



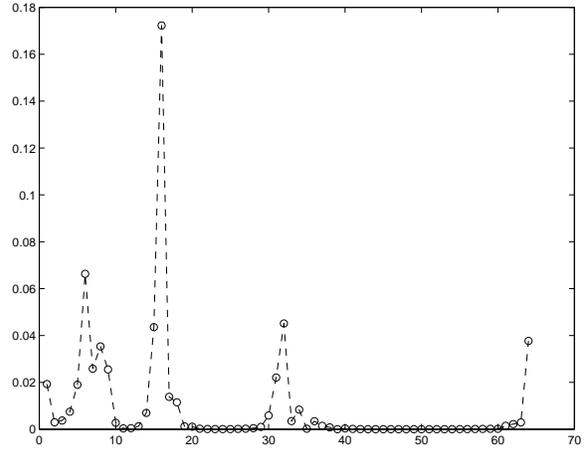
(a) Optimal spatial KF gain matrix



(b) A profile of (a)



(c) Optimal wavelet KF gain matrix



(d) A profile of (c)

Figure 5.17: (a) An example of the optimal *spatial* KF gain matrix ($\rho_h = \rho_v = 0.95$, $\sigma_v^2 = 1$). Note the large values are clustered around the main diagonal of the gain matrix. For the optimal *wavelet* KF (*db2* wavelet), the gain matrix is shown in (c). This matrix has a quite different structure from (a). However, both (a) and (c) are sparse, meaning the optimal KF can be well approximated by a strip KF.

KF. For the strip KF (see Chapter 2 for the background material of the strip KF), we also need to set up proper state and measurement equations. Assume we split the j th column into M strips, i.e., $\mathbf{x}(j) = \{\mathbf{x}_i(j), i = 1, 2, \dots, M\}$ and accordingly $\mathbf{y}(j) = \{\mathbf{y}_i(j), i = 1, 2, \dots, M\}$. Then, following [26] the state and measurement equations for $\mathbf{x}_i(j)$ are

$$\begin{aligned}\mathbf{x}_i(j) &= F_i \mathbf{x}_i(j-1) + G_i \mathbf{w}_i(j), \quad j = 1, 2, \dots, N. \\ \mathbf{y}_i(j) &= \mathbf{x}_i(j) + \mathbf{v}_i(j), \quad j = 1, 2, \dots, N.\end{aligned}\tag{5.41}$$

where

$$F_i = [E(\mathbf{x}_i(j)\mathbf{x}_i^T(j-1))] \cdot [E(\mathbf{x}_i(j)\mathbf{x}_i^T(j))]^{-1}$$

and

$$G_i G_i^T = E(\mathbf{x}_i(j)\mathbf{x}_i^T(j)) - F_i E(\mathbf{x}_i(j-1)\mathbf{x}_i^T(j-1)) F_i^T$$

For the wavelet-based strip KF, the state and measurement equations are formed similarly to (5.39) and (5.40).

5.3.2 Optimal Strip Structures

The 2-D KF with the state equation (5.37) or (5.39) has the complexity of $O(N \cdot N^3)$, which is still expensive to compute for large N . Woods *et.al.* [26] observed that the typical KF gains of the NSHP models, $K(j)$, only have significant values, in magnitude, around the main diagonal of the gain matrices, as illustrated in Fig. 5.17 for the steady-state $K(j)$ matrix corresponding to $\rho_h = \rho_v = 0.95$ and $\sigma_v^2 = 1$. So they proposed to do KF using only pixels in strips, thus the so-called strip KF. In this way, they considerably reduced the complexity of KF with little sacrifice (increase) of MSE . For example, if we choose

strip width $M = 16$ for image size $N^2 = 64^2$ then the complexity of the strip KF is only 1/16 that of the optimal KF. However, the relative increase of MSE , δ_{MSE} (for $\sigma_v^2 = 0.01$) is just 0.02%. Woods *et. al.* suggested to choose strip width M according to correlation lengths. In the previous sections, we have seen that a wavelet transform can greatly reduce the correlation length of an original process, so it is interesting to see how well the wavelet-based strip KF works and to compare it with its spatial counterpart. An example of the optimal wavelet KF gain matrix (with *db2* wavelet) is also given in Fig. 5.17. Like its spatial counterpart this sparse matrix can be well approximated by a strip filter.

To evaluate a strip KF, we need first to find the optimal strip structures in both the spatial and wavelet domains under some complexity constraint. This problem can be defined as

Assume the underlying image to have size $N \times N$. The strip design problem is to split the N elements of one column into M strips, each having length $l_i, i = 1, 2, \dots, M$, which minimizes the MSE:

$$\epsilon^2 = \sum_i^N e_i^2$$

where e_i^2 is the error variance of element i . The minimization is subject to a complexity constraint:

$$\mathcal{C} = \sum_i^M l_i^3 < T$$

where T is some preset complexity threshold. Note that T is with respect to the complexity of one column. The complexity of the whole image is clearly $N \cdot T$.

Thus, finding the optimal strip structures is a constrained discrete optimization problem, which is difficult to solve globally (due to the high dimensionality of data). We used a steepest-descent like approach to find sub-optimal structures. To limit getting stuck in the local minima, we start from several different initializations and choose the best one, which has yielded acceptable results. Our optimization approach has the following steps:

1. Specify the complexity threshold T ;
2. Initialize the strip structure by setting up N independent strips, each having the length of 1. Thus, the initial complexity is $\mathcal{C} = N \cdot 1^3 = N$.
3. Compute the current complexity \mathcal{C} . If $\mathcal{C} \geq T$, go to Step 5.
4. Increment the size of one of the N strips, choosing the one which gives the maximum decrease of MSE (ϵ^2). Go to Step 3.
5. Merge strips if this leads to a decrease in ϵ^2 and has the resulting complexity $\mathcal{C} \leq T$. If some merge happens, go to Step 4.
6. Stop.

We will see in the next section that the above simple approach shows interesting strip structures in the wavelet domain in addition to good estimation performance.

5.3.3 Spatial and Wavelet-based Strip KFs

Equipped with the state and measurement equations plus sub-optimal strip structures we are now able to compare the spatial and wavelet-based strip KFs according to δ_{MSE} (5.31) for an image denoising problem. The state and measurement equations are given in (5.41),

with $\rho_h = \rho_v = 0.95$ and various noise levels $\sigma_v^2 = 0.1 \sim 10$. The image size is $N^2 = 64^2$. For wavelet transforms we use three typical wavelet bases: Haar, 4-tap Daubechies wavelet (*db2*) and 8-tap Daubechies (*db4*). Three levels of decomposition were taken with the coefficient vector $\kappa_1 = [\mathbf{a}_3, \mathbf{d}_3, \mathbf{d}_2, \mathbf{d}_1]^T$, where \mathbf{a}_3 and \mathbf{d}_1 represent the coarsest and finest levels, respectively. Fig. 5.18 shows the sub-optimal strip structures found by the above procedure for fixed complexity $\mathcal{C} = 4 \times 16^3$ but variable noise $\sigma_v^2 = 0.1, 1, 10$. This complexity corresponds to splitting one image column into 4 strips, each having the length of 16. The following observations are clear:

1. Spatial and wavelet strips have very different structures. The spatial strips have almost equal strip lengths, whereas the lengths of the wavelet strips can be quite different.
2. Spatial strip structures do not change with σ_v^2 . However, the structure of wavelet strips depends heavily on σ_v^2 .
3. With the increase of σ_v^2 , the wavelet strips tend to emphasize on the low frequency subbands (e.g., $\mathbf{a}_3, \mathbf{d}_3$). To see this, compare Fig. 5.18(c) and (i). On the other hand, for small σ_v^2 both low-frequency and high-frequency subbands are included in strips. In many proposed denoising systems in literatures (e.g., [9], [79], [20], *etc.*), \mathbf{a}_3 is left untouched. However, our strip structures clearly show that when σ_v^2 is large \mathbf{a}_3 must be processed.
4. Comparing the strip structures of Haar and *db2* wavelets qualitatively, the former are more spread and has more interscale neighbor elements while the latter seems to prefer intrascale ones. We also compared the *db2* with *db4* wavelets and found the

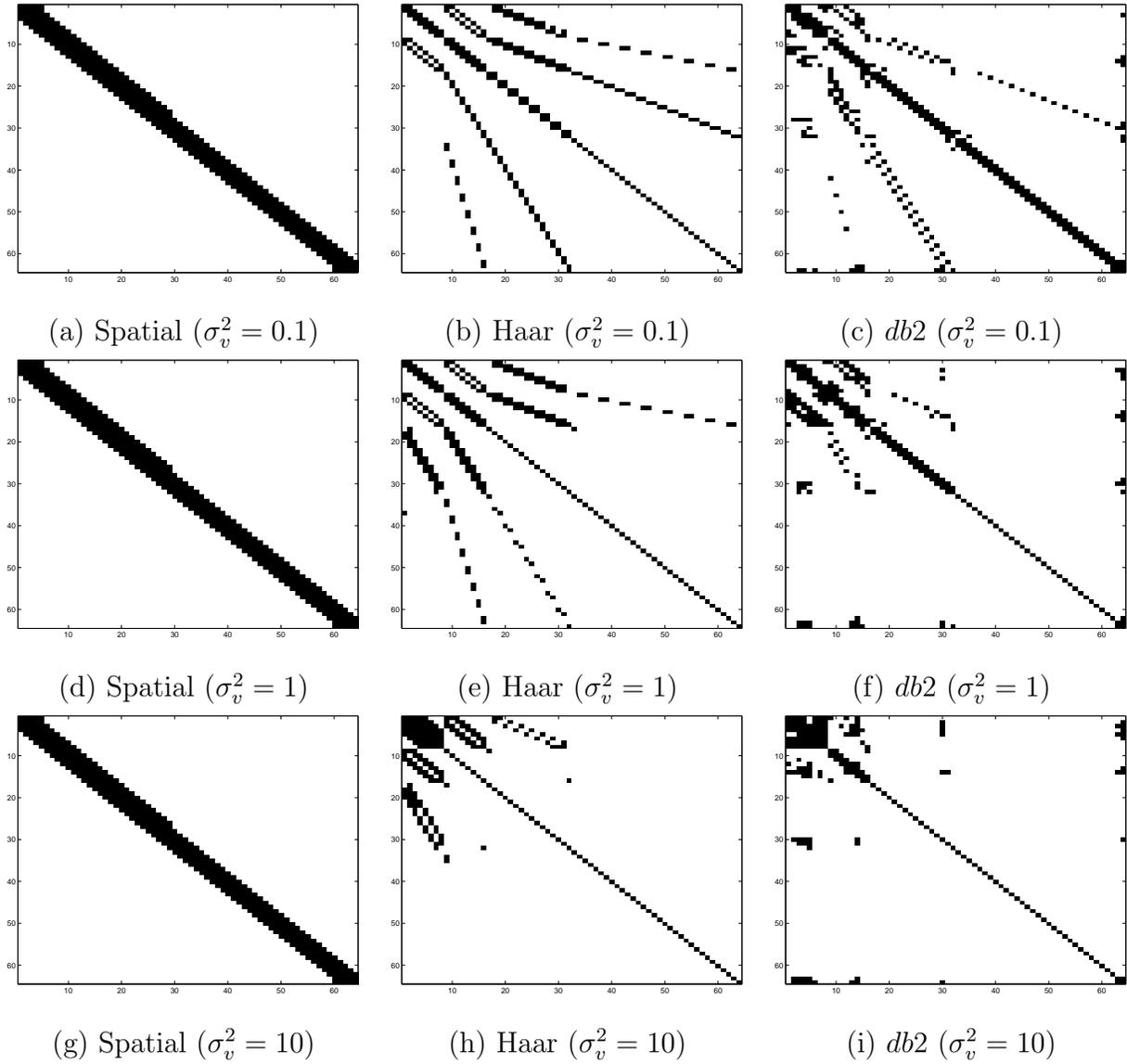


Figure 5.18: The spatial and wavelet (Haar and *db2*) strip structures for fixed complexity $\mathcal{C} = 4 \times 16^3$ and various noise variances $\sigma_v^2 = 0.1, 1, 10$. Each row forms a strip. Note the large differences between these two types of structures.

strip structure of *db4* has even more intrascale elements than *db2*. Recall we have mentioned that compared to the Haar wavelet the coefficients of *db2* wavelet has stronger intrascale but weaker interscale correlations (see Fig. 5.6). This is consistent with our current observation. We further conjecture that with the increase of wavelet vanishing moments, most information is represented by intrascale and near interscale neighbors instead of other coefficients. To prove this formally, some information measure needs to be defined. For example, in [74] the mutual information is used for this purpose. Here, our measure is δ_{MSE} (Eq. 5.31). So if our conjecture is correct then δ_{MSE} must be a very small number when wavelet strip filters use *only* a few intrascale and near interscale neighbor coefficients. This fact is shown in the denoising experiments (Tables 5.1 to 5.4).

5. The strips in Figs. 5.18(c),(f) and (i) consist of a few boundary elements. We already see in Fig. 5.4 that boundary elements have strong correlations. Artifacts occurs in boundary areas if these correlations are ignored. This will be demonstrated in the following denoising results. However, our strip design procedure takes care of this problem automatically.

We have seen that the spatial and wavelet strips have quite different structures. Furthermore the wavelet strips of various bases (e.g., Haar, *db1*, *db2*) look different. Besides the above qualitative comparisons, quantitative study is also necessary. For this, we use δ_{MSE} to measure the performance of the strip filters. Tables 5.1 to 5.4 list δ_{MSE} for different complexities \mathcal{C} and noise levels σ_v^2 . We have the following explanations for these tables:

1. When the strip sizes are 1, corresponding to the complexity of 64×1^3 , the wavelet strip filters are much better than the spatial ones for all the tested noise levels.

Domain	Complexity \mathcal{C}			
	64×1^3	8×8^3	4×16^3	2×32^3
Spatial	16.46	0.05	1.94e-04	1.39e-08
db_1	5.60	2.06	1.54	0.14
db_2	3.14	1.37	0.72	0.20
db_4	2.23	0.43	0.22	0.09

Table 5.1: δ_{MSE} (in %) for the optimal spatial and wavelet strip filters for $\sigma_v^2 = 0.01$. When the strip size is 1 (complexity 64×1^3), the wavelet strip filters perform much better than their spatial counterparts, due to the energy compaction property of wavelets. For larger strip sizes (higher complexities), the spatial strip filter performs best most likely because of the complicated, non-local structure of wavelet correlations, in contrast to the single structure in the spatial domain.

Domain	Complexity \mathcal{C}			
	64×1^3	8×8^3	4×16^3	2×32^3
Spatial	99.05	5.09	0.49	0.01
db_1	18.48	8.15	5.25	0.14
db_2	11.42	4.26	2.29	0.49
db_4	8.21	1.50	0.73	0.27

Table 5.2: Like Table 5.1, for $\sigma_v^2 = 0.1$. Compared with Table 5.1, we can see that the δ_{MSE} increases, as expected, however the spatial strip filters are much more sensitive than the wavelet strip ones.

Domain	Complexity \mathcal{C}			
	64×1^3	8×8^3	4×16^3	2×32^3
Spatial	209.60	47.61	19.04	4.82
db_1	27.33	4.00	1.53	0.24
db_2	25.46	2.50	0.78	0.16
db_4	21.09	1.22	0.40	0.12

Table 5.3: Increasing σ_v^2 to 1, the wavelet strip filters perform much better than the spatial one for all cases of complexities. In addition, the db_4 wavelet gives the best results.

Domain	Complexity \mathcal{C}			
	64×1^3	8×8^3	4×16^3	2×32^3
Spatial	91.57	55.48	38.59	22.10
db_1	33.64	1.15	0.56	0.20
db_2	33.70	0.81	0.25	0.04
db_4	32.34	0.51	0.16	0.02

Table 5.4: For $\sigma_v^2 = 10$, the wavelet strip filters perform much better than the spatial one for all cases of complexities.

Specifically, *db4* wavelet gives the best performance. Note in this case all elements are processed *independently*. For low-level noises (e.g., $\sigma_v^2 = 0.01$ and 0.1) assuming independence in the wavelet domain (*db4* wavelet) causes *MSE* to increase by only 2.23% and 8.21% (i.e., $\delta_{MSE} = 2.23\%$ and 8.21%), compared to the increase of 16.46% and 99.05% in the spatial domain. This underscores why many researchers assumed the simple independence model in the wavelet domain but still achieved impressive denoising results (e.g., [9],[35],[20],[21]). The advantage of the wavelet transform is mainly due to its energy compaction (de-correlation) property.

However, for high-level noises (e.g., $\sigma_v^2 = 1$ and 10) assuming independence in either spatial or wavelet domains may not be appropriate since the minimum relative increase of *MSE* is 21.09%.

2. For the low complexity $\mathcal{C} = 8 * 8^3$, the *db4* wavelet strip filters give good performance for the large tested noise-level range (i.e., $\sigma_v^2 = 0.01$ to 10). For example, in this noise range the *maximum* δ_{MSE} is 1.50%. For sharp comparison the maximum δ_{MSE} is 55.48% for the spatial strip filters of same complexity. This suggests most useful information for a wavelet coefficient is carried by a *few* other coefficients (strip elements). On the contrary information structure in the spatial domain becomes more spread with the increase of σ_v^2 .
3. For high-level noise cases we have observed two facts for the wavelet strip filters: (1) strip size should be at least larger than 1; (2) most strip elements are from the low-frequency subbands (e.g., \mathbf{a}_3 and \mathbf{d}_3) (see Fig. 5.18). These facts suggest the wavelet coefficients in low-frequency subbands must be also processed so as to effectively suppress noise. It should be noted most proposed denoising approaches

(e.g., [9], [35], [20], [21]) assumed independence models (i.e., strip size is 1) and neglected the remaining correlation (information) in wavelet coefficients. So those approaches should be only applicable to low-level noise cases. Recently, Portilla [19] *et.al.* tried to exploit the remaining inter- and intra-scale correlation. They compared with many other denoising methods and claimed the best denoising results (in the sense of MSE).

4. The spatial strip filter may be a little better than the wavelet strip filter, when σ_v^2 is small and strip size is large enough. However, the wavelet strip filter generally has *much* lower risk (MSE) when the strip size is small or when σ_v^2 is large.

5.3.4 Strip-based Denoising Experiments

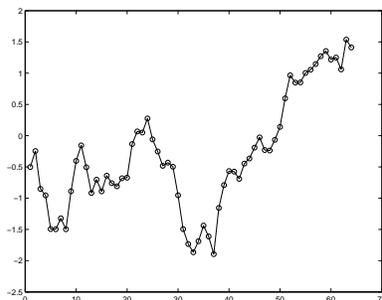
We have compared the spatial and wavelet strip filters in term of optimal strip structures and δ_{MSE} . In this subsection, we give the results of several 1-D signal and 2-D image denoising experiments for better understanding of the behaviors of the spatial and wavelet strip filters. The image model is still the above NSHP model (Eq. 5.37) with $\rho_h = \rho_v = 0.95$ and size $N^2 = 64^2$. For the tested 1-D signals, we use columns of the NSHP images. Rosenfeld [75] showed that such 1-D signals obey the first-order AR model (5.1) with $\alpha = \rho_v$. For the wavelet transform, we perform 3-level decompositions with the coefficient vector $\kappa_1 = [\mathbf{a}_3^T, \mathbf{d}_3^T, \mathbf{d}_2^T, \mathbf{d}_1^T]^T$. We use the *db4* wavelet since it gives the best overall performance for all of the noise levels σ_v^2 and complexities \mathcal{C} tested, as shown in Tables 5.1 through 5.4. Because the goal of the strip filter is to significantly reduce complexity we only test two complexities: 64×1^3 and 8×8^3 . The strips are designed by the optimization procedure discussed in subsection 5.3.2. We test three noise levels: $\sigma_v^2 = 0.01, 0.1, \text{ and } 10$.

In Figs. 5.19-5.21, we give the denoised 1-D signals from the strip filters. For comparison, we also show the results of the optimal filter, i.e., the one using all the 64 elements. We can see when σ_v^2 is very small (i.e., 0.01) even the simplest size 1 spatial strip filter looks similar to the optimal one. However, as σ_v^2 increases to 10 only the result of the size 8 wavelet strip filter is close to the optimum. It should be noted the well-used size 1 wavelet strip model seems only acceptable for small σ_v^2 cases (i.e., $\sigma_v^2 = 0.01$ and 0.1). Besides δ_{MSE} we can also use the confidence interval of the error $e_{optimal} = \hat{x}_{strip} - \hat{x}_{optimal}$ to measure the denoising performance of strip filters. The confidence interval $[-3\sqrt{E\{e_{optimal}^2\}}, 3\sqrt{E\{e_{optimal}^2\}}]$ of the strip filters for $\sigma_v^2 = 10$ is shown in Fig. 5.22, superimposed on the optimal estimate for comparison. Note that the wavelet strip filter with complexity 8×8^3 has much smaller error confidence interval than other strip filters. We also two more comments of the figure: (1) in Fig. 5.22(c) the error bars show approximate periodicity. This will be discussed in the next paragraph. (2) in Fig. 5.22(c) and (d) larger errors happen around boundaries. This is due to boundary discontinuities, which cause strong correlations of the corresponding wavelet coefficients. If thus correlations can not be completely modeled by the strip filters then larger errors show up in these areas. However, boundary errors in (d) are much smaller than those in (c), meaning the optimal strips designed by our algorithm in subsection 5.3.2 can take care of the boundary effects properly.

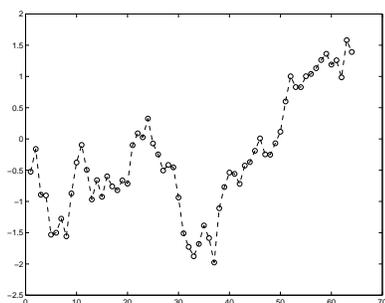
Now let us take a look at the variances of the denoising error,

$$\sigma_{e_x}^2 = \text{diag}(E[(\mathbf{x} - \hat{\mathbf{x}}_{strip})(\mathbf{x} - \hat{\mathbf{x}}_{strip})^T]), \quad (5.42)$$

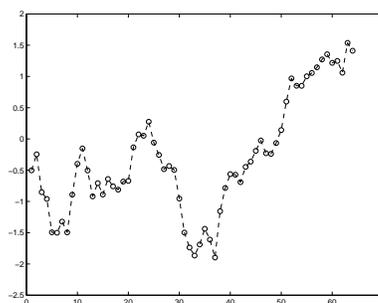
of the spatial and wavelet (*db4*) strip filters for $\sigma_v^2 = 10$ (Fig. 5.23), where $\text{diag}(A)$ means the vector consisting of the diagonal elements of the matrix A . It is interesting to see that Fig. 5.23(b) has a rough periodic structure, which does not exist in other parts of Fig. 5.23.



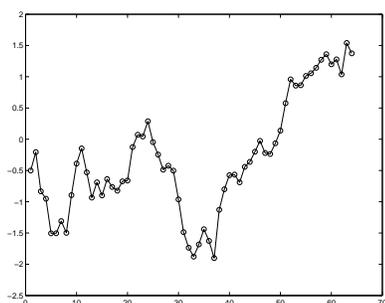
(a) Optimal estimate



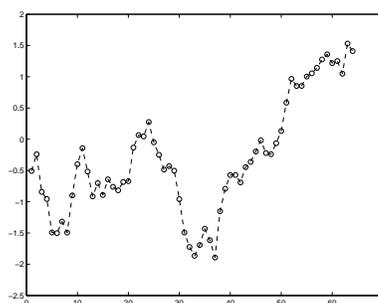
(b) Spatial, complexity 64×1^3



(c) Spatial, complexity 8×8^3

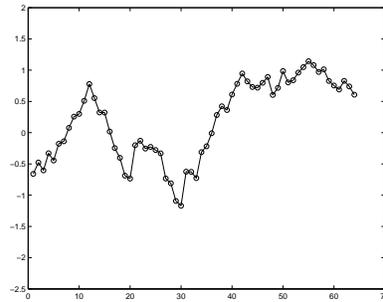


(d) Wavelet, complexity 64×1^3



(e) Wavelet, complexity 8×8^3

Figure 5.19: Compare the denoising results of the spatial and wavelet strip filters for $\sigma_v^2 = 0.01$. For comparison, optimally filtered signal is also given. Because the noise σ_v^2 is so small the differences in the results are unnoticeable.



(a) Optimal estimate

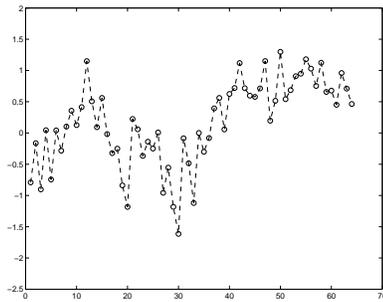
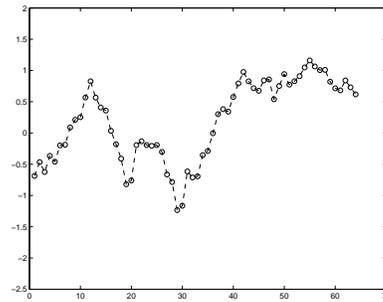
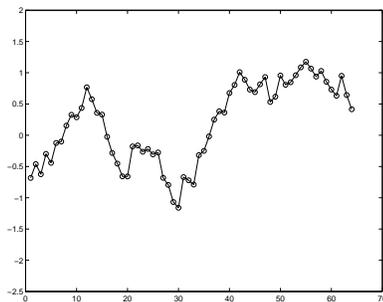
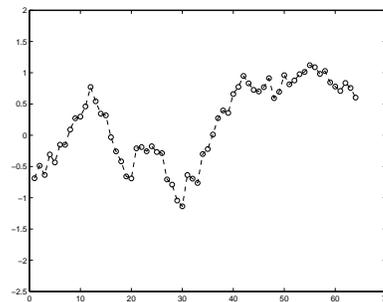
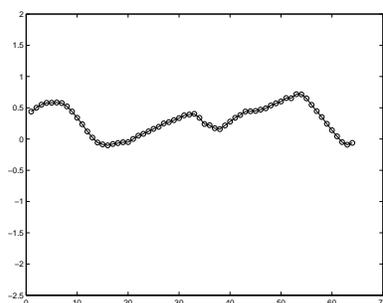
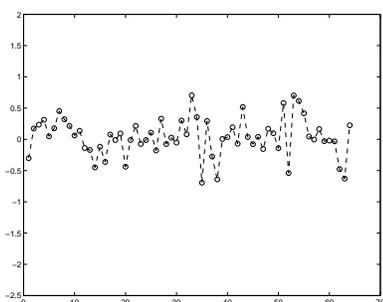
(b) Spatial, complexity 64×1^3 (c) Spatial, complexity 8×8^3 (d) Wavelet, complexity 64×1^3 (e) Wavelet, complexity 8×8^3

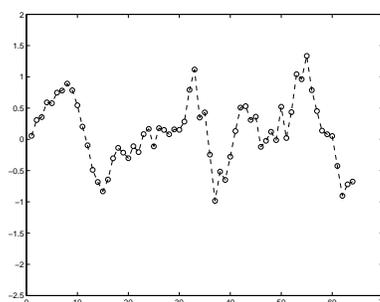
Figure 5.20: When $\sigma_v^2 = 0.1$, the length 1 spatial strip filter has significant differences from the optimal filter. However, the length 1 wavelet strip filter still does a good job.



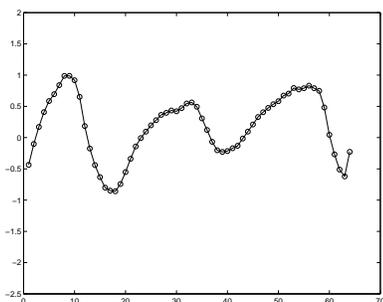
(a) Optimal estimate



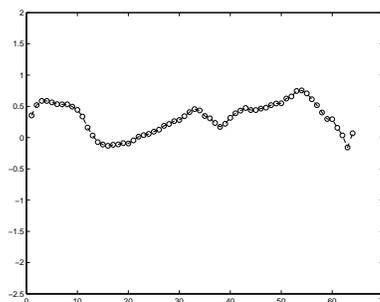
(b) Spatial, complexity 64×1^3



(c) Spatial, complexity 8×8^3



(d) Wavelet, complexity 64×1^3



(e) Wavelet, complexity 8×8^3

Figure 5.21: When σ_v^2 is increased to 10, the results of the two spatial filters are not particularly poor. The wavelet strip filter of size 1 has large deviations from the optimal filter, however, the size 8 wavelet strip filter gives credible results. The deviations of the strip filters to the optimal one can be measured using confidence intervals of the error $E[(\hat{\mathbf{x}}_{strip} - \hat{\mathbf{x}}_{optimal})^2]$, shown in Fig. 5.22.

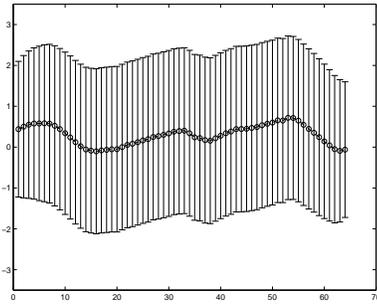
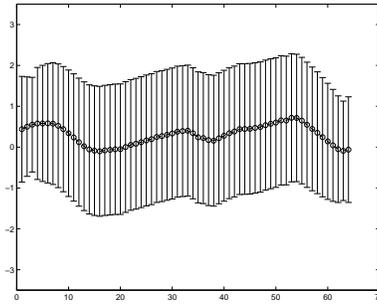
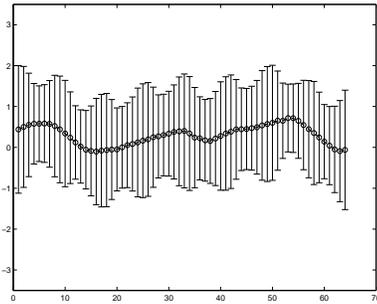
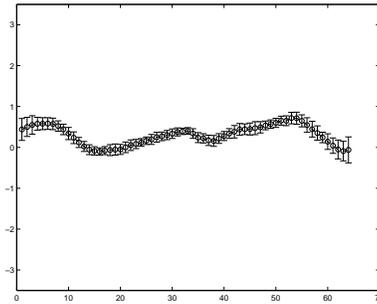
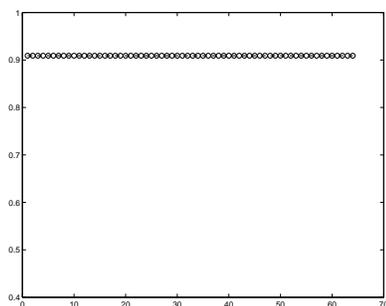
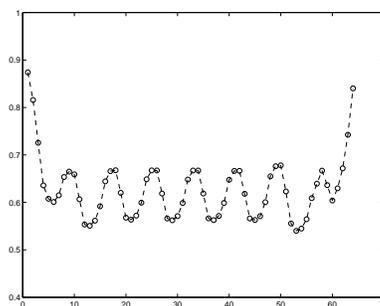
(a) Spatial strip with complexity 64×1^3 (b) Spatial strip with complexity 8×8^3 (c) Wavelet strip with complexity 64×1^3 (d) Wavelet strip with complexity 8×8^3

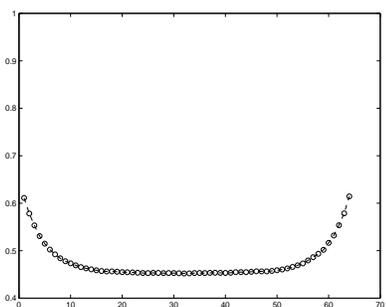
Figure 5.22: Plot of the error confidence intervals of 4 strip filters for $\sigma_v^2 = 10$. Note the wavelet strip filter with complexity 8×8^3 has the smallest interval, which is *much* smaller than the others.



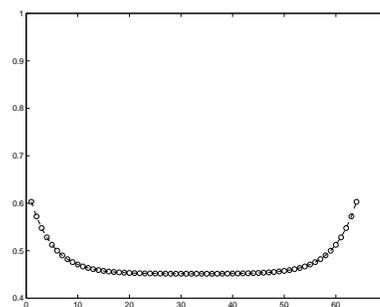
(a) Spatial error pattern ($\mathcal{C} = 64 \times 1^3$)



(b) Wavelet error pattern ($\mathcal{C} = 64 \times 1^3$)



(c) Wavelet error pattern ($\mathcal{C} = 8 \times 8^3$)



(d) Optimal error pattern ($\mathcal{C} = 64^3$)

Figure 5.23: The independence model in the wavelet domain yields near-periodic error pattern, except the near boundaries (see (b)). To suppress this type of periodic artifact, remaining correlation of the wavelet coefficients must be considered (see (c)).

Note for Fig. 5.23(b) the strip sizes are 1 (corresponding to complexity $\mathcal{C} = 64 \times 1^3$), which means the assumption of the independence model of wavelet coefficients. This is equivalent to assuming the covariance matrix in the wavelet domain (P_{DWT}) to be diagonal. Golden [70] showed that if P_{DWT} is diagonal, then the corresponding covariance matrix in the spatial domain, $P = H^T P_{DWT} H$, has a periodic diagonal, consistent with the behavior of Fig. 5.23(b). However, the wavelet coefficients of stationary processes are correlated, consistent with our study results of correlation structures in the wavelet domain, so to suppress this type of periodic artifact the remaining correlation of the wavelet coefficients must be considered in applications. Fig. 5.23(c) plots the error pattern of the wavelet strip filter with a larger complexity $\mathcal{C} = 8 \times 8^3$, which is almost free of this periodic phenomenon. Under the complexity $\mathcal{C} = 8 \times 8^3$, many strips have sizes larger than 1 (see Fig. 5.18), which means the correlation of wavelet coefficients is exploited in these strip filters. For comparison, we show in Fig. 5.23(d) the error pattern of the optimal estimator, which looks very similar to Fig. 5.23(c).

Next we compare the spatial and wavelet strip Kalman filters (KF) for image denoising. The original image model is the NSHP model (5.37). We perform three-level wavelet decomposition using the *db4* wavelet. Three noise levels ($\sigma_v^2 = 0.01, 0.1$ and 10) are tested. Note the state and measurement equations of the strip KF are in (5.41), instead of (5.37) and (5.38), which are for the optimal KF. Figs. 5.24 to 5.26 give the denoising results for $\sigma_v^2 = 0.01, 0.1$ and 10 , respectively, along with the noisy images. For comparison, we also show the denoising results of the optimal KF. Generally speaking, we have similar observations as those for the 1-D signal denoising. Visually, the wavelet strip KFs outperform their spatial counterparts. Especially, the wavelet strip KF of complexity 8×8^3 performs quite close to the optimal KF (see Table 5.5 for δ_{MSE}). It should be noted this complexity

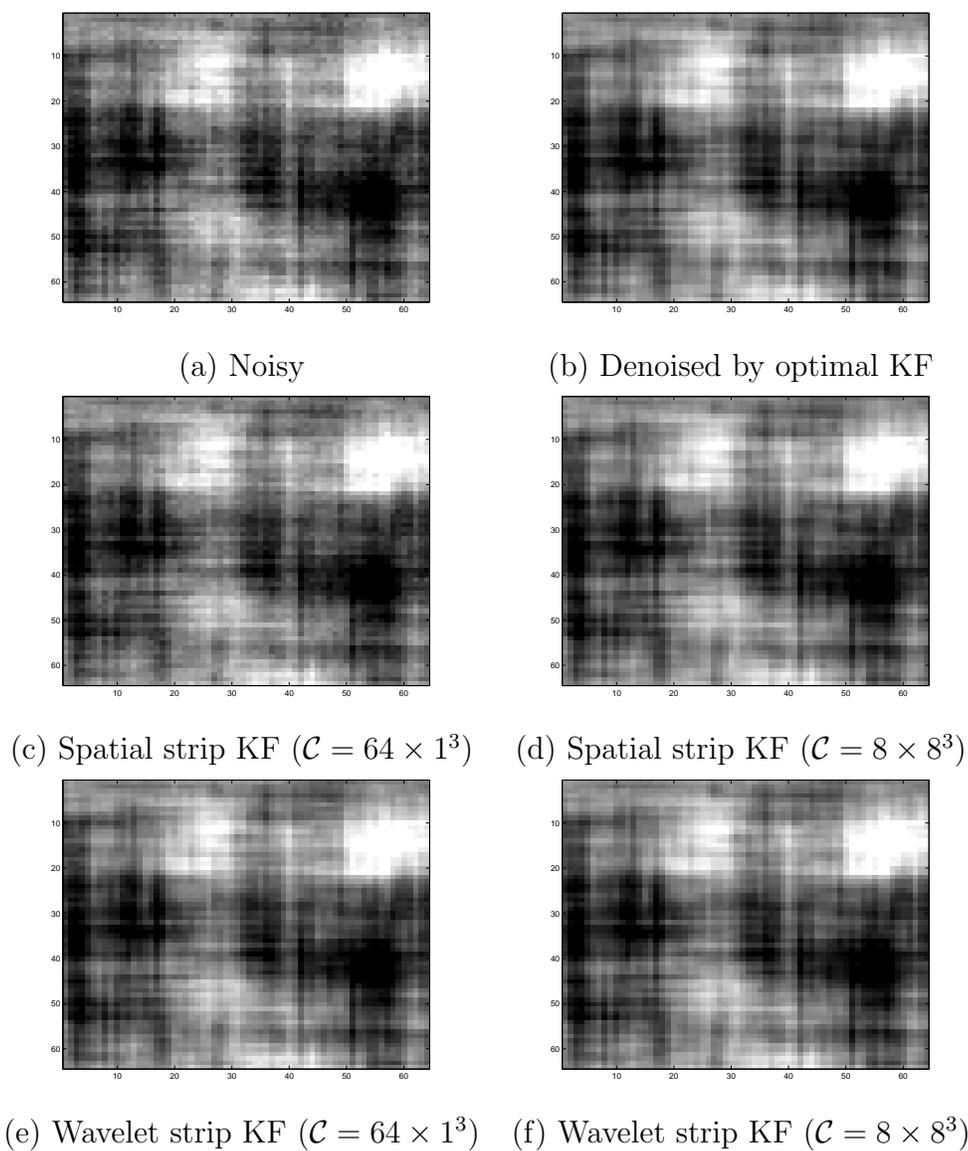
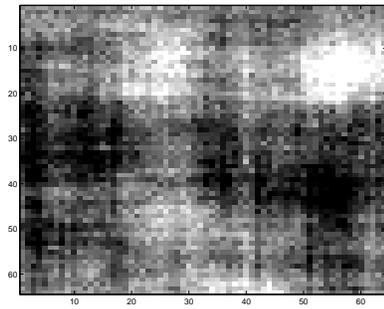
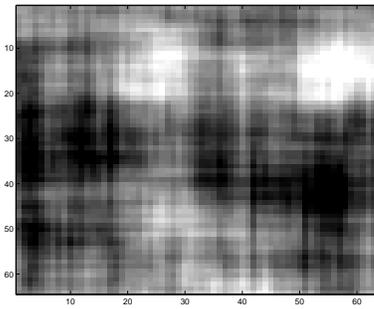


Figure 5.24: Image denoising by the spatial and wavelet strip KFs for low-noise level $\sigma_v^2 = 0.01$. Compared with the result of the optimal filter visually, it is clear the result of the size 1 (i.e., $\mathcal{C} = 64 \times 1^3$) spatial strip KF is much noisier than others. For comparison, the size 1 wavelet strip KF gives a much better result.



(a) Noisy



(b) Denoised by optimal KF

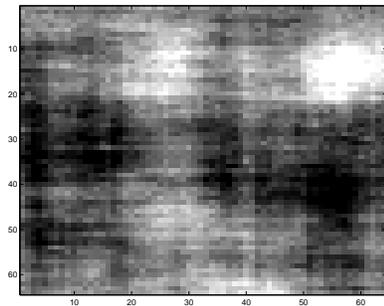
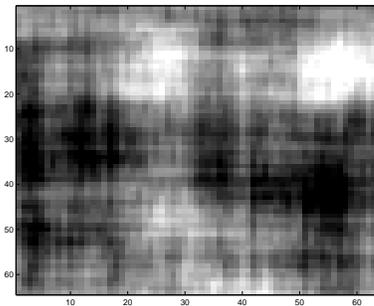
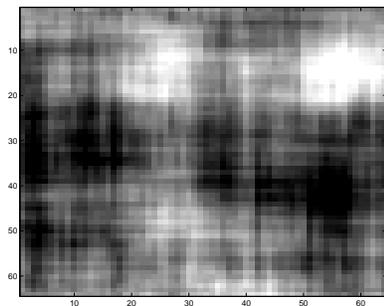
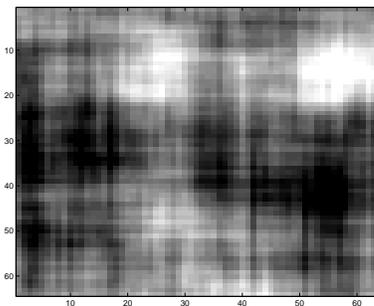
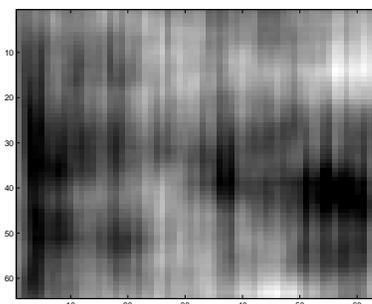
(c) Spatial strip KF ($\mathcal{C} = 64 \times 1^3$)(d) Spatial strip KF ($\mathcal{C} = 8 \times 8^3$)(e) Wavelet strip KF ($\mathcal{C} = 64 \times 1^3$)(f) Wavelet strip KF ($\mathcal{C} = 8 \times 8^3$)

Figure 5.25: Similar to Fig.5.24, but σ_v^2 is increased to 0.1. The size 1 spatial strip KF is unacceptable now. Interestingly, the size 1 wavelet strip KF looks even better than the spatial strip KF with complexity $\mathcal{C} = 8 \times 8^3$, though it has a much lower complexity. This advantage of wavelet strip KF comes from the strong decorrelation powers of the wavelet transforms as shown in the first part of the chapter.



(a) Noisy



(b) Denoised by optimal KF

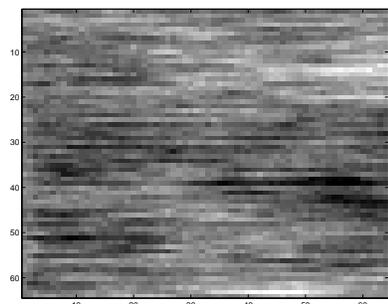
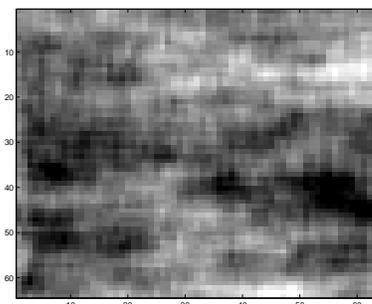
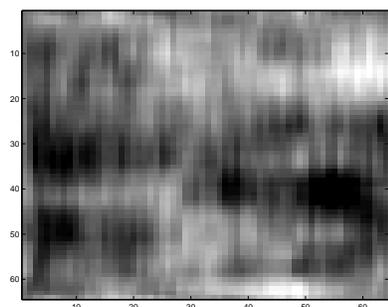
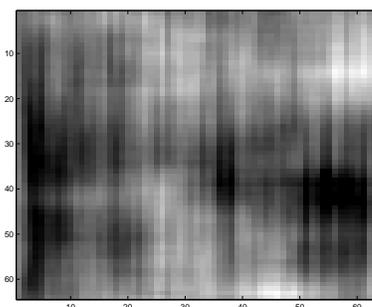
(c) Spatial strip KF ($\mathcal{C} = 64 \times 1^3$)(d) Spatial strip KF ($\mathcal{C} = 8 \times 8^3$)(e) Wavelet strip KF ($\mathcal{C} = 64 \times 1^3$)(f) Wavelet strip KF ($\mathcal{C} = 8 \times 8^3$)

Figure 5.26: If we further increase σ_v^2 to 10, then *only* the result of the wavelet strip KF with $\mathcal{C} = 8 \times 8^3$ looks close to the optimal one. This suggests the wavelet transform can not completely de-correlate the NSHP model and for large σ_v^2 , remaining correlation of wavelet coefficients must be considered even for visual evaluation.

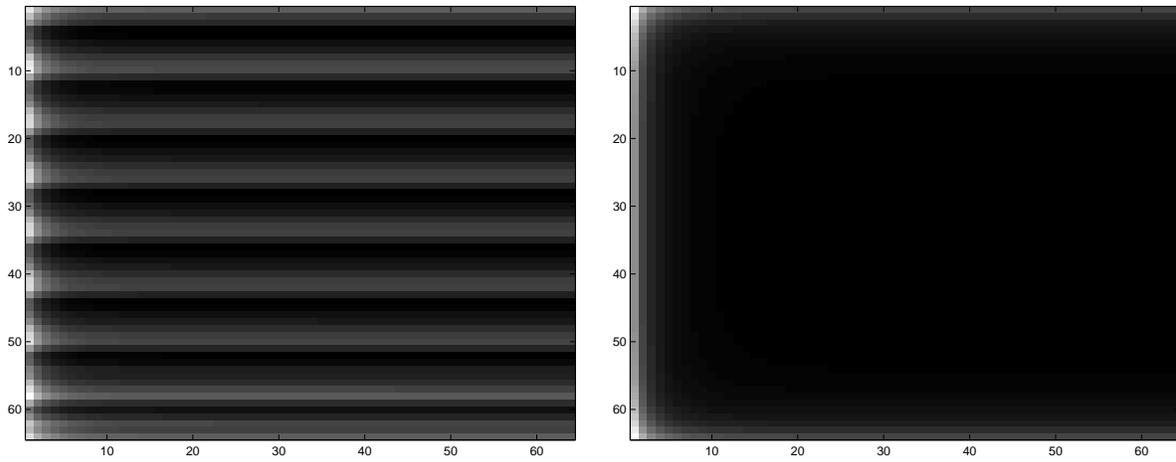
Strip KF	σ_v^2		
	0.01	0.1	10
Spatial ($\mathcal{C} = 64 \times 1^3$)	57.53	110	97.29
Spatial ($\mathcal{C} = 8 \times 8^3$)	3.35	18.11	32.68
Wavelet ($\mathcal{C} = 64 \times 1^3$)	4.81	5.85	14.08
Wavelet ($\mathcal{C} = 8 \times 8^3$)	0.94	0.79	0.03

Table 5.5: δ_{MSE} (in percentage) of the spatial and wavelet strip KFs for various noise levels $\sigma_v^2 = 0.01, 0.1$ and 10 . Clearly, the wavelet strip KFs outperform their spatial counterparts. Especially, the wavelet strip KF with complexity $\mathcal{C} = 8 \times 8^3$ performs very closely to the optimal filter both visually and with respect to δ_{MSE} . When σ_v^2 is relatively small, the size 1 wavelet strip KF is also attractive, considering its simplicity.

is just $1/64$ that of the optimal one.

The error variances of the size 1 wavelet strip filter (*db4*, $\sigma_v^2 = 10$) is shown in Fig. 5.27(a). Note the intrinsic periodic pattern with this strip model. We also display the error variances of the optimal KF for comparison. Note this type of periodic artifact is often more obvious for larger σ_v^2 .

Due to the impressive performance of the wavelet-based filters shown above, in the next two chapters we will perform wavelet-based real-world image and video denoising as applications of these processors.



(a) Error pattern of wavelet strip KF

(b) Error pattern of the optimal KF

Figure 5.27: (a) Periodic error pattern of the wavelet strip KF with size 1. (b) The optimal KF is free from such a pattern.

Chapter 6

Wavelet-based Image Denoising

The last decade has seen a good deal of effort in exploiting the power of wavelet transforms for solving practical problems. These include estimation [79, 80, 81], detection and classification [82], compression [10], prediction and filtering [7] and synthesis [66]. Most noticeably, the wavelet transform has been adopted in modern image and video compression standards (e.g., JPEG-2000 and MPEG-4 [83, 84]). In this chapter, we are interested in the application of wavelets to real-world image denoising problems.

Images are frequently corrupted by noise in the acquisition and transmission phases. The goal of denoising is to remove the noise, both for aesthetic and compression reasons, while retaining as much as possible the important signal features. Traditionally, denoising is achieved in the spatial domain by approaches such as Wiener filtering, which is the optimal estimator in the sense of mean squared error (MSE) for stationary Gaussian processes. However, the requirement of stationarity and an accurate statistical model of the underlying process lead to a poor performance on natural images, which fails these assumptions. As a result, attempts to apply the Wiener filter to natural images tend to lead to blurred edges.

In practice, adaptive methods [85, 86] are mostly used. Adaptive methods are good in that they are fast and can effectively suppress noise for many *natural* images. More importantly, their adaptivity allows them to work for non-stationary processes. The main problem with such methods is their assumption that natural images are independent random processes, which usually is not true.

Since the last decade, the wavelet transform has been studied extensively to suppress noise in natural images [9, 19, 76, 81, 87] because of its effectiveness and simplicity. It is now well-known that wavelet transforms with some regularity have strong decorrelation abilities (see Section 5.2), and can well represent an image with just a few large coefficients. Therefore, it is far more reasonable to assume that the wavelet coefficients are *independent* than the original spatial domain pixels. This explains why impressive denoising results have been achieved by simply thresholding or shrinking each wavelet coefficient *independently* [9, 19, 76]. Indeed, this kind of approach has much better results than the traditional methods [85, 86], both subjectively and objectively.

A quick review of the literature on the wavelet-based image denoising approaches is worthwhile. The pioneering work of Donoho *et.al.* [35] inspired the research on wavelet-based denoising. They tried to constrain the smoothness of denoised signals and to find the asymptotically optimal minimax regression estimator, which turned out to be a simple thresholding process but outperformed many previous denoising methods (e.g., [85, 88]). This kind of idea was later developed by Starck *et.al.* [15] for image denoising. Donoho's work was mainly based on deterministic signal analysis. Because noise is naturally a stochastic process, many researchers adopted the probabilistic models and Bayesian inference approaches to achieve denoising in the wavelet domain. The representative models include the non-Gaussian independence model [9], hidden Markov tree model (HMT) [22],

Gaussian scale mixture model (GSM) [19] and Markov field model [76]. A brief description of these models is given in Chapter 2. We can summarize these ideas for denoising as follows:

1. Non-Gaussian independence models, typically generalized Gaussian distribution (GGD) and Gaussian mixture models, were used to characterize the distribution of wavelet coefficients [9, 81, 89, 90]. Although the resulting denoising equations are relatively simple, they were often effective in terms of MSE and impressive denoising results have been obtained. The main problems are the Gibbs-like artifacts around edge areas due to ignoring strong remaining correlations in these areas.
2. Gaussian models exploiting inter- and/or intra-scale correlations were used by [91, 92, 93, 94]. For image denoising, a significant decrease of MSE (relative to the above independence model) was reported when noise variances are relatively high. However, these models ignore the non-Gaussian nature of wavelet coefficients and due to limited amount of data available for estimating the needed parameters, artifacts can often be seen due to parameter estimation errors.
3. Non-Gaussian models exploiting inter- and/or intra-scale correlations were proposed by, for example [19, 20, 21, 57]. These models include the well-known hidden Markov tree (HMT) models [57, 58, 95], Markov field models [20, 21, 96, 97] and scale mixture Gaussian (GSM) models [19]. These models are sophisticated and have very powerful representation strengths. They can be used not only for image denoising, but also for segmentation [23], detection [82], enhancement [55]. However, for image denoising, the formulations using these models usually do not result in closed-form solutions and high-complexity numerical iterative methods have to be used. Another problem

with these models is none of them can represent edge areas very well and thus have unavoidable artifacts in these areas.

In this chapter, we propose a new wavelet-based image denoising approach. We do not intend to propose any new models of wavelet coefficients, as there are already many powerful ones. Instead, noting that the most annoying artifacts often exist around edges, and that it is difficult to accurately model (or estimate) the statistics in such areas, we propose a preprocessing step to remove edges (i.e., mean value differences) prior to taking the wavelet transform. We expect this can reduce the remaining correlation of the wavelet coefficients. We showed in Chapter 5 (see Fig. 5.8) that the mean value difference can cause strong correlation of wavelet coefficients. If such remaining correlation is not taken care of properly, they can produce severe Gibbs-like artifacts (see Fig. 5.10). Another issue we are interested in is to compare the performance of the real-valued and the complex-valued wavelets for image denoising. Kingsbury [32] suggested that the complex wavelet has good potential for denoising applications, which we will test experimentally.

6.1 Problem formulation

The image denoising problem can be modeled as follows: given the noisy measurements for spatial indices i, j

$$y(i, j) = x(i, j) + v(i, j), \quad i, j = 1, 2, \dots, N \quad (6.1)$$

we need to estimate the noise-free image $x(i, j)$, where N is the size of the square image, and where the additive noise $v(i, j)$ is assumed to be independent and identically distributed (*iid*) as Gaussian (i.e., $v(i, j) \sim N(0, \sigma_v^2)$), and independent of $x(i, j)$. Following [68, 69, 85,

86] we also assume $x(i, j)$ to be Gaussian: $x(i, j) \sim N(\mu_x(i, j), \sigma_x^2(i, j))$. Note that $x(i, j)$ constitutes a non-stationary process.

Defining a vector $\mathbf{x} = \{x(i, j), i, j = 1, 2, \dots, N\}$ by stacking the image $x(i, j)$ column-by-column (lexicographically) and similarly for $v(i, j)$ and $y(i, j)$ we can rewrite (6.1) as

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \quad (6.2)$$

where $\mathbf{x} \sim N(\mathbf{u}_x, P_x)$ and $\mathbf{v} \sim N(\mathbf{u}_v, \sigma_v^2 I)$. Since \mathbf{x} and \mathbf{v} are jointly Gaussian, so is $\mathbf{y} \sim N(\mathbf{u}_y, P_y)$. Thus, the optimal estimate of \mathbf{x} is a linear function of \mathbf{y} :

$$\hat{\mathbf{x}} = P_{xy} P_y^{-1} (\mathbf{y} - \mathbf{u}_y) + \mathbf{u}_x \quad (6.3)$$

where $P_{xy} = E\{\mathbf{x}\mathbf{y}^T\}$ is the cross-correlation matrix of \mathbf{x} and \mathbf{y} . Although (6.3) clearly shows the mathematical relationship between the estimate $\hat{\mathbf{x}}$ and measurement \mathbf{y} , in image denoising, it is seldom used directly for at least two reasons:

1. P_y^{-1} needs to be evaluated in (6.3). However, P_y usually has a very large size. For example, for a moderate $N = 256$, P_y is of the size 65536×65536 . To exactly compute the inverse of such a large matrix is very difficult, if ever possible, unless P_y has some special form (e.g., diagonal), not common in practice.
2. $P_{xy} = P_x$ is needed in (6.3). However, for real-world images P_x is very difficult to model and estimate due to the large variability of natural images.

So, finding $\hat{\mathbf{x}}$ directly from (6.3) is a very difficult (if ever possible) task.

In practice many researchers use approximate models of P_x . These models must satisfy at least the following conditions:

1. They should have good representation power. That is, they can model the underlying problems reasonably well.
2. It should be relatively easier to estimate model parameters.
3. The resulting problem should be computable in practice.

The usual models include causal recursive models [68, 69] and non-causal Markov random fields (MRF) [20, 21]. The main problem with the causal recursive model is its poor ability to represent non-causal processes. However, the non-causal MRF model is usually very computationally intensive.

So far we have been trying to handle the estimation problem in the original spatial domain. However, it may simplify in a transformed domain. As \mathbf{x} and \mathbf{v} are Gaussian the optimal whitening transform is linear; denoting by H an *invertible* linear transform and applying it to (6.2) we have

$$H\mathbf{y} = H\mathbf{x} + H\mathbf{v} \quad (6.4)$$

Denoting $\mathbf{y}_H \equiv H\mathbf{y}$, $\mathbf{x}_H \equiv H\mathbf{x}$ and $\mathbf{v}_H \equiv H\mathbf{v}$ (6.4) is rewritten as

$$\mathbf{y}_H = \mathbf{x}_H + \mathbf{v}_H \quad (6.5)$$

(6.5) is the measurement equation in the transform domain. Since H is invertible we can estimate \mathbf{x} by estimating \mathbf{x}_H first and then taking inverse transformation, i.e.

$$\hat{\mathbf{x}} = H^{-1}\hat{\mathbf{x}}_H \quad (6.6)$$

where

$$\hat{\mathbf{x}}_H = P_{x_H} P_{y_H}^{-1}(\mathbf{y}_H - \mathbf{u}_{x_H}) + \mathbf{u}_{x_H} \quad (6.7)$$

and $P_{x_H} = HP_xH^T$, $P_{y_H} = HP_yH^T$ and $\mathbf{u}_{x_H} = H\mathbf{u}_x$.

Although H can be any invertible transform, ideally we hope H can diagonalize P_{x_H} because this makes the solution of (6.7) very simple. This ideal diagonalizer is the well-known Karhonen-Loeve (KL) transform [25]. Unfortunately, the KL transform needs the eigenvectors of P_x , which are very costly for computation. Interestingly, the wavelet transforms, as a universal unconditional basis, are well-known to be a good approximation to the KL transform [3, 35]. Donoho *et.al.* [35, 98, 99] showed that the asymptotical optimal minimax regression estimator of \mathbf{x}_H is to shrink each wavelet coefficient *independently*. They proposed hard-thresholding (6.8) and soft-thresholding methods (6.9) with the universal threshold $T = \sigma_v\sqrt{2\log N}$, where N is the data length.

$$\eta_T(x) = \begin{cases} x, & |x| > T \\ 0, & otherwise \end{cases} \quad (6.8)$$

$$\theta_T(x) = \text{sgn}(x) \cdot \max(|x| - T, 0) \quad (6.9)$$

Mallat [3], Chang *et. al.* [9] and Chipman *et. al.* [81] also assumed the independence model in the wavelet domain. They used generalized Gaussian distribution (GGD) model for the wavelet coefficients. In Chapter 5 we showed that in the sense of MSE the independence model in the wavelet domain has the performance close to that of the true model for small σ_v^2 (this is often the case in practice). Therefore for simplicity, it is often acceptable to treat the wavelet coefficients as being independent to each other. We call this model the *independence model*.

However, the wavelet transform is not equivalent to the KL transform and many authors (e.g., [10, 19, 91, 93]) have noted that the remaining correlation among wavelet coefficients are useful for denoising and compression. In particular, the most *significant* correlations

are mostly focused in the inter- and intra-scale neighborhoods. Note that the neighborhood may include coefficients from other subbands (e.g., the parents in Fig. 5.15), as well as from the same subband. We call a model which exploits neighborhood information as a *correlation model*.

So, following others, we will use two common models, independent and correlated, in the wavelet domain for our denoising task:

Independence model: We model a wavelet coefficient at location (i, j) of level l , $x_H^{(l)}(i, j)$, as $x_H^{(l)}(i, j) \sim N(\mu_{x_H}^{(l)}(i, j), (\sigma_{x_H}^{(l)}(i, j))^2)$. Later, for notational simplicity we drop the superscript l if no confusion happens. Since P_{x_H} is treated as diagonal we can rewrite (6.7) in a much simpler form

$$\hat{x}_H(i, j) = \frac{\sigma_{x_H}^2(i, j)}{\sigma_{x_H}^2(i, j) + \sigma_{v_H}^2(i, j)}(y_H(i, j) - \mu_{x_H}(i, j)) + \mu_{x_H}(i, j) \quad (6.10)$$

We will call (6.10) the wavelet-based Lee filter [88].

From (6.10) it is clear that to find $\hat{x}_H(i, j)$ we only need to know $\sigma_{x_H}^2(i, j)$, $\mu_{x_H}(i, j)$ and $\sigma_{v_H}^2(i, j)$. For real-world images these parameters have to be estimated from measurement data. However, due to the non-stationarity of natural images, how to *accurately* estimate $\sigma_{x_H}^2(i, j)$ and $\mu_{x_H}(i, j)$ is still an open problem. In the next subsection we propose a clustering-based adaptive weighted averaging (AWA) method to estimate these parameters. Specifically, we set $\mu_{x_H}(i, j) = 0$ by introducing a mean-value removal preprocessing step before taking wavelet transform. That is, we estimate mean values in the spatial domain, instead of in the wavelet domain, and then subtract them from measurements. This is because we observed that much information of mean values is lost in the wavelet domain around edge areas. Accordingly, by modifying (6.4), we get our final measurement equation in the wavelet

domain:

$$H(\mathbf{y} - \mu_{\mathbf{x}}) = H(\mathbf{x} - \mu_{\mathbf{x}}) + H\mathbf{v} \quad (6.11)$$

The spatial mean vector $\mu_{\mathbf{x}}$ is estimated by AWA. Then the estimation equation is (6.10) with $\mu_{x_H}(i, j) = 0$.

To estimate the variance of the noise, $\sigma_{v_H}^2(i, j)$, we use the sample variance of \mathbf{y}_H in several hand-selected *smooth* patches as its estimate. Donoho [98] proposed to use the following function to estimate $\sigma_{x_H}^2(i, j)$

$$\sigma_{v_H} = \text{median}(|y_H(i, j)|)/0.6745, \quad y_H(i, j) \in \{\text{finest subband}\} \quad (6.12)$$

We found experimentally the above two methods for estimating σ_{v_H} have very close results.

Correlation model: We use a neighborhood of coefficients drawn from two subbands at adjacent scales, thus taking advantage of the strong statistical coupling observed through scales in multiscale representations. Specifically, we experimentally choose a 3×3 region surrounding the reference coefficient, together with the coefficient at the same location and orientation at the next coarser scale (the *parent*). Such a neighborhood structure is also used in [19]. The related estimation equation is:

$$\hat{\mathbf{x}}_H = P_{x_H}(P_{x_H} + P_{v_H})^{-1}\mathbf{y}_H \quad (6.13)$$

The signal covariance matrix P_{x_H} can be computed from the observation sample covariance matrix P_{y_H} :

$$P_{x_H} = P_{y_H} - P_{v_H} \quad (6.14)$$



Figure 6.1: A wavelet-based denoising result using the independence model of the wavelet coefficients. Note the Gibbs-like artifacts around the body of the cameraman.

We force P_{x_H} to be positive semidefinite by performing an eigenvalue decomposition and setting any possible negative eigenvalues (nonexisting or negligible, in most cases) to zero.

Owning the models of wavelet coefficients and related estimation equations, we should next address the parameter estimation problem.

6.2 Adaptive weighted averaging (AWA)

This is the key part of this chapter. To find \hat{x}_H we need to first estimate the required parameters (i.e., μ and σ^2). Almost all previous work (e.g., [9, 76, 89]) simply assumed that wavelet coefficients have zero mean values, i.e., $\mu_{x_H}(i, j) = 0$. We have observed that

this assumption often causes Gibbs-like (or ringing) artifacts around image edges, as seen in Fig. 6.1, so we propose a preprocessing step to remove the mean in the spatial domain, prior to taking the wavelet transform. We can then assume zero mean values for wavelet coefficients. However, this requires us to correctly estimate mean values.

Besides mean values, we also need to estimate variances. There are mainly three approaches for estimating $\sigma_{x_H}^2(i, j)$:

1. LoPresto *et.al.* [89] use local sample variance.
2. Chang *et.al.* [9] first group coefficients with similar *context* together, and then estimate variances by averaging the group members.
3. Based on the assumption that $\sigma_{x_H}^2(i, j)$ are random variables and change slowly, Mihcak *et.al.*[76] tried to estimate variances according to a MAP formulation.

Note that all the above methods work on local data in a window of size S . Finding the optimal S is not easy: when the window size is too small the estimates will be sensitive to noise; however, a large window adapts slowly to changing characteristics. We propose a new nonlinear approach (AWA) to handle this dilemma. Our idea is to use relatively *large* size windows and cluster the window elements into groups. Then only the elements within the *same* group as the reference element are used in estimation. Since AWA is a general approach for mean and variance estimation (not specifically for use in the wavelet domain), for notational clarity we will discuss its principles in the spatial domain.

A. Principle of Adaptive Weighted Averaging

Given measurements $\{y(i, j), i, j = 1, 2, \dots, N\}$, it is common to use the sample mean and

sample variance in a chosen window of size $(2r + 1) \times (2r + 1)$ as the estimates of mean and variance at (i, j) :

$$\hat{\mu}_x(i, j) = \frac{1}{(2r + 1)^2} \sum_{p=i-r}^{i+r} \sum_{q=j-r}^{j+r} y(p, q) \quad (6.15)$$

$$\hat{\sigma}_x^2(i, j) = \frac{1}{(2r + 1)^2} \sum_{p=i-r}^{i+r} \sum_{q=j-r}^{j+r} [y(p, q) - \hat{\mu}_x(i, j)]^2 - \sigma_v^2(i, j) \quad (6.16)$$

It is well-known that (6.15)-(6.16) tend to blur estimates around edge areas [85, 89]. For more accurate estimates we propose to modify (6.15)-(6.16) by introducing pixel-dependent weights $w(i, j)$ to each window element [100, 101]:

$$\begin{aligned} \hat{\mu}_x(i, j) &= \sum_{p=i-r}^{i+r} \sum_{q=j-r}^{j+r} w(i, j, p, q) \cdot y(p, q) \\ \hat{\sigma}_x^2(i, j) &= \sum_{p=i-r}^{i+r} \sum_{q=j-r}^{j+r} w(i, j, p, q) \cdot [y(p, q) - \hat{\mu}_x(i, j)]^2 - \sigma_v^2(i, j) \end{aligned} \quad (6.17)$$

The weights $w(i, j)$ play the role of selection functions. The basic idea is to select the window elements of the same cluster as the reference pixel $x(i, j)$ and use only these elements in estimating $\hat{\mu}_x(i, j)$ and $\hat{\sigma}_x(i, j)$. We use a distance-based clustering to find $w(i, j)$:

$$w(i, j, p, q) = \frac{K(i, j)}{1 + a(\max[\epsilon^2, (y(i, j) - y(p, q))^2])} \quad (6.18)$$

where $K(i, j)$ is the normalization constant:

$$K(i, j) = \left\{ \sum_{p, q} \frac{1}{1 + a(\max[\epsilon^2, (y(i, j) - y(p, q))^2])} \right\}^{-1} \quad (6.19)$$

The quantities a and ϵ are the parameters of the weight function. We choose $\epsilon^2 = 4\sigma_v^2$, i.e., the expected value of the square of the difference between two element values that differ

PSNR (dB) (Original)	30.0	25.2	22.2	20.5
AWA (Synthetic)	45.8	40.7	37.9	35.9
Traditional (Synthetic)	34.9	32.6	30.4	29.0
AWA (Cameraman)	33.4	30.1	28.0	26.7
Traditional (Cameraman)	26.4	25.8	25.0	24.5
AWA (Lena)	33.9	31.7	30.3	29.6
Traditional (Lena)	29.5	28.6	27.8	27.2

Table 6.1: Comparison of PSNR’s of the traditional and AWA-based spatial Lee filters. Our AWA-based approach outperforms the traditional one by at least $2dB$ for the noise range tested. Visually, the AWA-based filter also gives much better results (see Fig. 6.2-6.5).

due to the presence of noise only. It is easy to see that when the differences among the element values in a window are due to the presence of noise only, then AWA reduces to the sample mean and sample variances of (6.15)-(6.16). We choose a such that $a\epsilon^2 \gg 1$ to exclude outliers from the weight function $w(\cdot)$. Given $w(\cdot)$ we estimate both local mean and local variance *adaptively* using (6.17). It might be interesting to note that the AWA estimator functions like an amplitude-limited averager, which is closely related to the well-studied Sigma filter [88, 102].

B. Comparison of AWA with traditional methods

To show the superiority of AWA to the traditional method (i.e., using sample mean / variance) for estimating the local mean / variance and for subsequent image denoising, we test these two methods on both synthetic and real-world images. The local mean and vari-

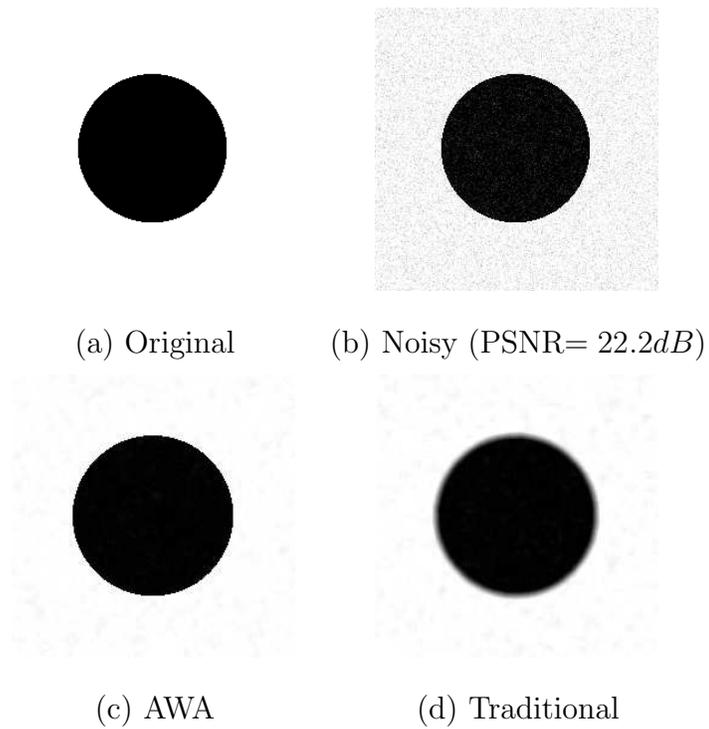


Figure 6.2: Local mean estimation using AWA (part (c)) and the traditional sample mean (part (d)). Clearly, in (d) the true mean values are blurred. On the contrary, our AWA estimator does not have this problem and gives much better estimates.

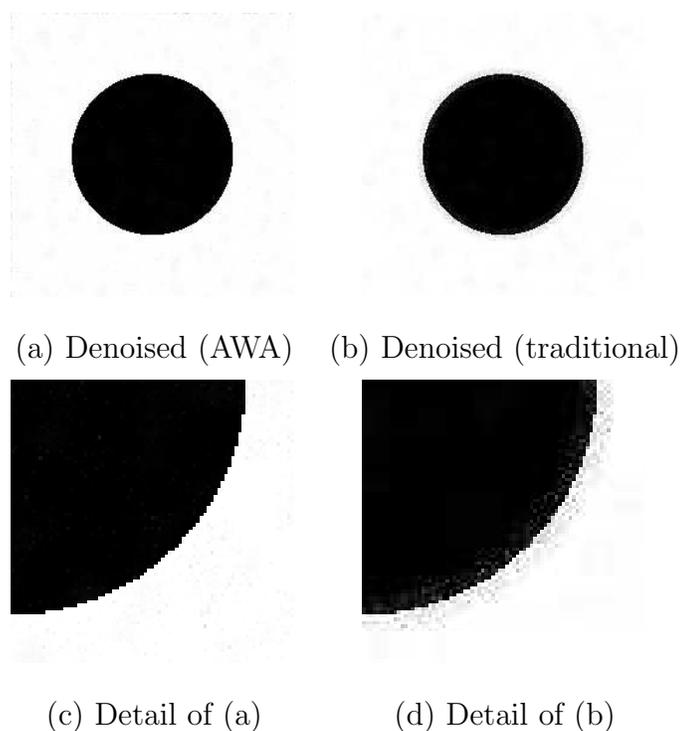


Figure 6.3: Comparison of the AWA-based and traditional *spatial* Lee filters for denoising the synthetic image *Circle* (Fig. 6.2(a),(b)). Due to using poorly estimated parameters (μ_x and σ_x^2) the traditional approach has very noisy edges (see part (d)). However, the proposed AWA-based spatial Lee filter gives a much cleaner result (see part (c)).

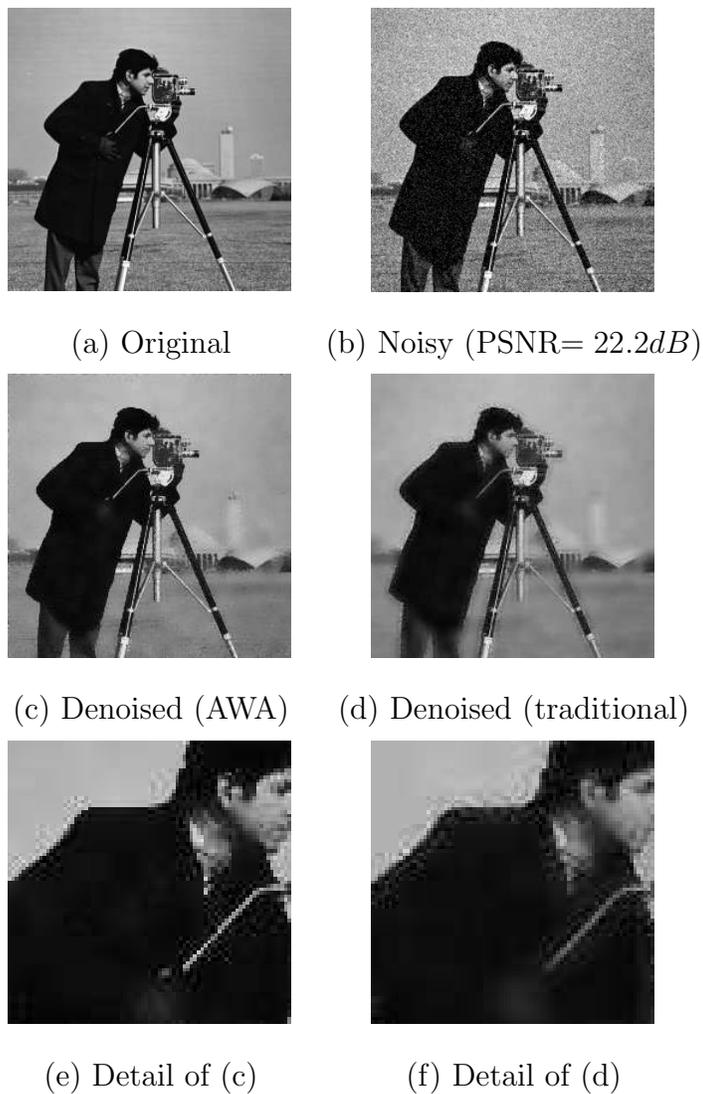


Figure 6.4: Comparison of the AWA-based and traditional *spatial* Lee filter for the image *Cameraman* (PSNR= 22.2dB). Like Fig. 6.3, the traditional filter tends to blur images and gives very noisy edges. On the contrary, our AWA-based filter has much sharper and cleaner results.

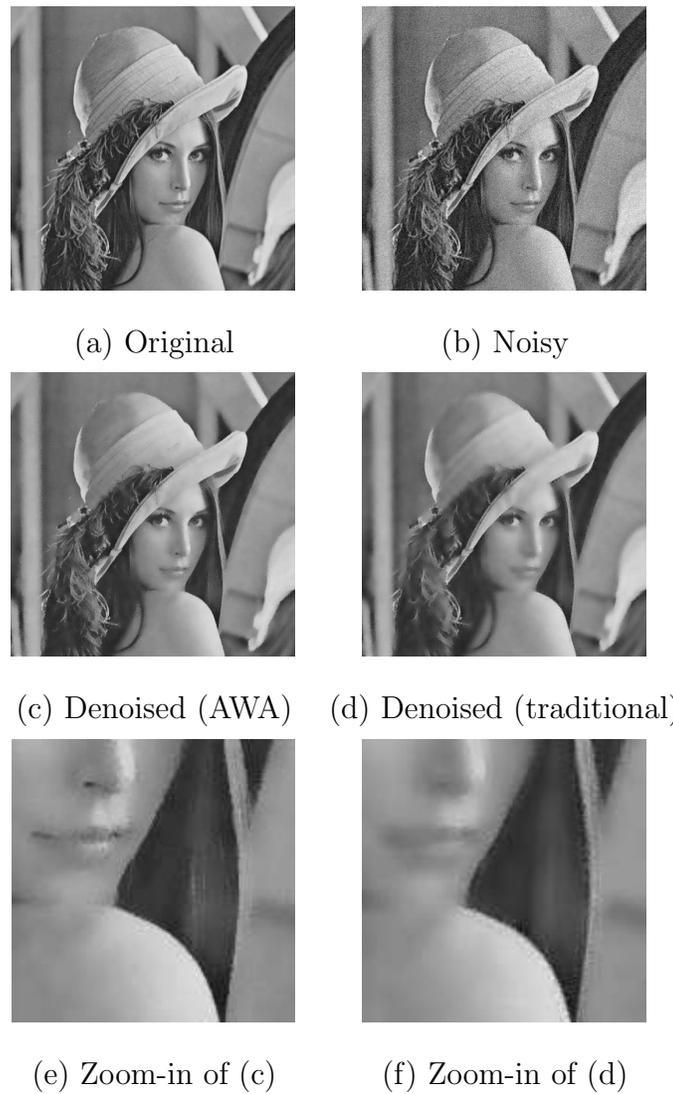


Figure 6.5: Similar test as Fig. 6.4 but for the image *Lena*.

ance are computed by (6.15), (6.16) and (6.17), respectively. The Lee filter [86], which is a pixel-based adaptive linear filter, is used for denoising, because it is well-used by researchers [85],[86] and it only needs local mean / variance as parameters. We use the Matlab function *wiener2*(\cdot) to implement the traditional Lee filter. The spatial Lee filter is given by

$$\hat{x}(i, j) = \frac{\hat{\sigma}_x^2(i, j)}{\hat{\sigma}_x^2(i, j) + \sigma_v^2(i, j)} \cdot (y(i, j) - \hat{\mu}_x(i, j)) + \hat{\mu}_x(i, j) \quad (6.20)$$

where $\hat{\mu}_x(i, j)$ and $\hat{\sigma}_x^2(i, j)$ are estimated local mean and local variance at (i, j) . $y(i, j)$ is the noisy measurement (i.e., $y(i, j) = x(i, j) + v(i, j)$). We assume that the noise variance $\sigma_v^2(i, j)$ is known. Recall that in this experiment we use two different methods (i.e., *AWA* and *sample mean / variance*) to estimate $\hat{\mu}_x(i, j)$ and $\hat{\sigma}_x^2(i, j)$, and we want to compare their denoising results. Fig. 6.2 shows the results of a synthetic image *Circle*. It is clear that Fig. 6.2(d) blurs the true mean in the vicinity of edges, while Fig. 6.2(c) gives a much sharper estimate. Lee filtering results are shown in Fig. 6.3. Note the poor estimates of local mean and variance from the traditional method (Fig. 6.2(d)) causes noisy edges (Fig. 6.3(b),(d)). However, AWA has a much cleaner denoising result (Fig. 6.3(a),(c)). In our second experiment we use two real-world images: *Cameraman* and *Lena*. The results are given in Fig. 6.4 and Fig. 6.5. It is clear that AWA also greatly outperforms the traditional method in these results. In Table 6.1 we compare peak SNR (PSNR) of the above denoising results. In all cases AWA beats the traditional method by more than $2dB$.

6.3 Wavelet-based image denoising system

Fig. 6.6 shows the block diagram of our wavelet-based denoising system. It mainly consists of six parts:

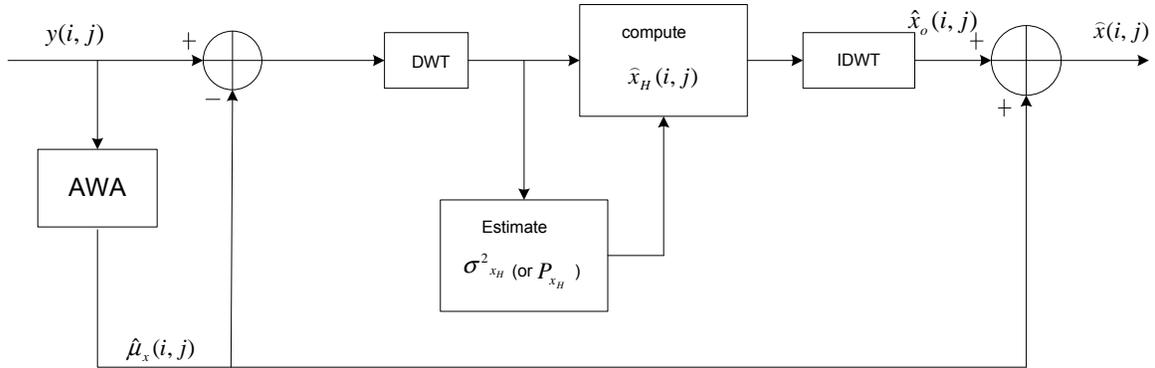


Figure 6.6: Block diagram of the proposed image denoising system

Preprocessing: Use AWA to estimate and then remove local mean $\hat{\mu}_x(i, j)$ (see (6.11)).

Wavelet transform (WT): Compute the wavelet decomposition of the mean-removed noisy image.

A large number of wavelet transforms have been developed (e.g., orthogonal / bi-orthogonal, decimated / overcomplete, real-valued / complex-valued). We will focus on three common wavelets in our image denoising experiments:

Orthogonal: Daubechies' symmelet with eight vanishing moments [11]. We note that this type of wavelet is sensitive to shifts.

Overcomplete: Overcomplete wavelets have many different implementations [34], [37]. We will use the *shift-averaging* scheme by Coifman and Donoho [38] to perform denoising in an overcomplete representation, which is explained below. Let $\text{Shift}_{k,l}[g]$ denote the operation of circularly shifting the input image g by k indices in the vertical direction and l indices in the horizontal, and let $\text{Unshift}_{k,l}[g]$ be a similar operation but in the opposite direction. Also, let

$\text{Denoise}[g]$ denote the operation of taking the DWT of the input image g , shrink it in some way, then transform it back to the spatial domain. Then Coifman's *shift-averaging* yields an output which is the average of the shrunken copies over all possible shifts:

$$\hat{f} = (1/N^2) \sum_{k,l=0}^{N-1} \text{Unshift}_{k,l}[\text{Denoise}[\text{Shift}_{k,l}[g]]]. \quad (6.21)$$

It is well-known that denoising in the overcomplete expansion can significantly ameliorate many unpleasant phenomena associated with the decimated wavelets, such as Gibbs-like ringing around edges and specks in smooth regions.

Complex-valued: We have showed in Chapter 3 that the complex wavelet transforms can be nearly shift-invariant, which is well-known to be important for denoising. So we are interested to see how well the complex wavelet transforms perform for real-world image denoising. The dual-tree complex wavelet designed by Kingsbury [103] will be used.

Estimate model parameters: For the *independence model* use AWA (6.17) to estimate $\sigma_{x_H}^2(i, j)$ and set $\mu_{x_H}(i, j) = 0$.

For the *correlation model* use measurements $y_H(i, j)$ to estimate \hat{P}_{y_H} . then compute \hat{P}_{x_H} by

$$\begin{aligned} \hat{P}_{x_H} &= \hat{P}_{y_H} - \hat{P}_{v_H} \\ &= \hat{P}_{y_H} - \sigma_v^2 \cdot I \end{aligned}$$

where I is the identity matrix. If \hat{P}_{x_H} is negative definite we make it positive semidefinite using eigenvalue decomposition.



(a) Original

(b) Noisy ($PSNR = 22.2dB$)Figure 6.7: The original and noisy images of the image *Cameraman*.

Compute $\hat{x}_H(i, j)$: For the *independence model* use (6.10) to compute $\hat{x}_H(i, j)$.

For the *correlation model* use (6.13) to compute $\hat{x}_H(i, j)$.

Inverse wavelet transform (IWT): Inverse transform $\hat{x}_H(i, j)$ back to the spatial domain to get $\hat{x}_o(i, j)$.

Add the removed mean: Add back the removed mean value to $\hat{x}_o(i, j)$:

$$\hat{x}(i, j) = \hat{x}_o(i, j) + \hat{\mu}_x(i, j) \quad (6.22)$$

$\hat{x}(i, j)$ is our final denoising result.

6.4 Experimental results

The standard images *Synthetic*, *Cameraman*, *Lena* and *Barbara* are used as test images. Independent identically-distributed (*i.i.d.*) Gaussian noise of different noise levels is added

AWA	Wavelet	Model	PSNR			
			30.0	25.2	22.2	20.5
N	db4	indep.	33.1	29.2	27.7	26.2
Y	db4	indep.	34.1	29.8	28.1	26.4
N	db4	corr.	33.5	29.5	27.9	26.4
Y	db4	corr.	34.3	30.1	28.3	26.6
N	OV	indep.	33.7	29.8	28.5	26.5
Y	OV	indep.	34.3	30.3	28.7	26.8
N	OV	corr.	33.9	30.1	28.7	26.8
Y	OV	corr.	34.6	30.6	28.8	26.8
N	CWT	indep.	33.6	29.7	28.5	26.4
Y	CWT	indep.	34.2	30.2	28.7	26.5
N	CWT	corr.	33.8	30.0	28.7	26.6
Y	CWT	corr.	34.3	30.4	28.8	26.7

Table 6.2: Compare the performance of 12 approaches (with/without AWA) for denoising image *Camerman*. We found AWA outperforms the standard approaches by about 0.6dB to 1dB for high PSNR cases.

PSNR (<i>dB</i>) (Input)	30.0	25.2	22.2	20.5
AWA+CWT+corr.	36.1	33.8	32.4	31.3
Portilla [19]	35.6	33.7	32.5	31.69
Chang+OV [9]	35.2	33.4	32.1	31.1
Mihcak [76]	34.8	32.4	31.0	29.9

Table 6.3: PSNR results of different methods for *Lena* image and various input PSNRs. *AWA + CWT + corr.* means the proposed combined method of AWA, complex wavelets and the correlation model. *Chang + OV* denotes Chang’s method with the overcomplete wavelet transform. Our proposed approach is better than or comparable to that by Portilla *et al.*, who claimed the best wavelet-based denoising performance so far. The proposed approach is consistently better than the other two well-referenced methods.

PSNR (<i>dB</i>) (Input)	30.0	25.2	22.2	20.5
AWA+CWT+corr.	34.9	32.1	30.2	28.7
Portilla [19]	34.3	31.9	30.3	29.1
Chang+OV [9]	33.6	30.8	29.5	28.3
Mihcak [76]	33.7	30.2	28.7	27.5

Table 6.4: PSNR results of different methods for the *Barbara* image and various input PSNRs. *AWA + CWT + corr.* means the combined method of AWA, complex wavelet and the correlation model. *Chang + OV* denotes Chang’s method with the overcomplete wavelet transform. We have the same observations as in Table 6.3.



(a) N-AWA+ortho+indep.



(b) AWA+ortho+indep.



(c) N-AWA+CWT+indep.



(d) AWA+CWT+indep.

Figure 6.8: Denoising results with and without AWA for the image *Cameraman*. The corresponding original and noisy images are shown in Fig. 6.7. Clearly, the proposed AWA-based approach can significantly suppress the Gibbs-like artifacts around the contour of the cameraman (compare (a) and (b)). Although the shift-invariant complex wavelet can also reduce these artifacts (compare (a) and (c)), combining AWA and the complex wavelet transform together can furthermore greatly improve performance (compare (c) and (d)).

and four levels of decomposition are used. For the orthogonal wavelet transform, the wavelet employed is Daubechies' symmlet with eight vanishing moments [11], which was also used in [9, 35]. For the complex wavelet transform we use Kingsbury's dual-tree complex wavelet [103].

We first compare the various models with/without AWA preprocessing for image denoising. The results of the image *Cameraman* are listed in Table 6.2. The results of other test images (*Synthetic*, *Cameraman*, *Barbara*) reflect similar behavior. We have the following observations from Table 6.2:

1. AWA-based approaches clearly outperform their counterparts without AWA (the difference is about 0.6dB to 1dB in PSNR) for relatively large *PSNRs* (e.g., 30.0dB and 25.22dB). For small *PSNRs* (i.e., 22.2dB and 20.5dB) AWA results are also better but with smaller differences. Fig. 6.8 shows the denoising results of the image *Cameraman*. It is clear that the traditional methods (non-AWA) (see Fig. 6.8(a) and (c)) suffer from serious Gibbs-like artifacts (such as the contour area of the cameraman), which have, however, been considerably suppressed by AWA (see Fig. 6.8(b) and (d)).
2. It is well-known that the shift-invariance property of a wavelet transform (e.g., complex wavelet and overcomplete wavelet) helps to reduce artifacts significantly. This explains why Fig. 6.8(c) has less obvious ringing (around the body of cameraman) than (a). However, the AWA-based orthogonal filter performs much better than the traditional complex wavelet (compare Fig. 6.8(c) and (b)). Indeed, almost no Gibbs-like artifacts show up in Fig. 6.8(b). This suggests the proposed AWA method is very useful for suppressing annoying Gibbs-like artifacts.



(a) N-AWA+CWT+indep.



(b) AWA+CWT+indep.



(c) N-AWA+OV+indep.



(d) AWA+OV+indep.

Figure 6.9: Compare the denoising results of complex and overcomplete wavelet transforms. Subjectively, these two transforms have very similar results. The quantitative results in Table 6.2 support the same conclusion.

3. Correlation information becomes increasingly more important with the increase of noise levels; This is consistent with our results in Ch. 5 where we showed that strip filters with size larger than 1 is needed for near optimal filtering when noise variance (σ_v^2) is large.
4. Due to its property of near shift-invariance and good directional selectivity, the dual-tree complex wavelet yields similar denoising result with the overcomplete wavelet; however, the former one has much lower complexity, as shown in Ch. 3. The denoising results of the complex and overcomplete wavelet transforms with/without AWA are shown in Fig. 6.9.

Next, we compare the proposed approach with several state-of-the-art denoising methods in the literature [9, 19, 76]. The PSNR results of images *Lena* and *Barbara* are shown in Tables 6.3 and 6.4, respectively. We choose these two test images because they are the only common data used by the above references.

Portilla [19] *et.al.* used a Gaussian scale mixture (GSM) model to represent the joint distribution of wavelet coefficients. The parameter *scale* is modeled by Jeffrey's noninformative prior function [104]. To the best of our knowledge, their results are the best (in the sense of PSNR) among the published ones so far. The second method we compare is by Chang *et.al.* [9]. They used a generalized Gaussian distribution (GGD) model and a *pixel*-adaptive soft-thresholding approach. A doubly stochastic process was used by Mihcak *et.al.* [76], and their scheme is also pixel-adaptive but instead of thresholding they performed shrinking due to using a Gaussian marginal model. From our 12 tested approaches in Table 6.3 we choose the one which combines the techniques of AWA, complex wavelet transform and correlation model.

From Tables 6.3 and 6.4 we can see that when the input PSNR is $30dB$ our proposed approach has the highest output PSNR, which is about $0.5dB$ higher than Portilla's results [19]. For PSNR= $25.2dB$ and $22.2dB$ the proposed approach is comparable to Portilla's scheme. For the worst noise case (PSNR= $20.5dB$) we tested our PSNR is $0.4dB$ lower than that by Portilla. Portilla's better PSNR probably comes from two facts: (1) For strong noise case, due to large classification error AWA can not find the proper weights it needs (see (6.18)); (2) Portilla *et.al.* used a prior model for its parameter *scale* (the Jeffrey's noninformative function). This model properly described the non-Gaussian characteristics of the variances of wavelet coefficients and allow a maximum a posterior (MAP) estimation of parameters. For comparison, we did not assume any a prior knowledge of variances and used an adaptive maximum-likelihood (ML) approach (AWA) to estimate parameters. Theoretically, MAP gives better estimates than ML. However, The MAP estimator of Portilla *et.al.* does not have a closed-form solution and thus *iterative* numerical approach has to be used, which is computationally expensive. On the contrary, our ML estimator is much cheaper to compute. For all these experiments we had better results than those of Chang and Mihcak.

To summarize, the proposed approach achieves one of the best results for low to moderate noise levels. For high noise applications our PSNR is about $0.4dB$ lower than the best result in literatures. However, if both performance and efficiency are considered the proposed method is still a competitive candidate for the wavelet-based image denoising.

Chapter 7

Wavelet-based Video Denoising

With the maturity of digital video capturing devices and broadband transmission networks, numerous applications have been emerging. These include teleconferencing, remote surveillance, multimedia services and digital television, to name a few. However, the video signal is almost always corrupted by noise from capturing devices or during transmission due to random thermal or other electronic noises. Usually, noise reduction can considerably improve visual quality and benefit the subsequent processing tasks, such as video compression.

There are many *spatial* domain video denoising approaches (e.g., [69, 101, 105, 106]), which can roughly be divided into two classes:

Temporal-only: A temporal-only approach utilizes only the temporal correlations, neglecting spatial information. Since video signals are closely correlated along motion trajectories, motion estimation/compensation is naturally employed. In those cases where the motion estimation is not accurate, motion detection is used to avoid blurring. These techniques can preserve spatial details well, but the resulting images

usually still contain removable noise since spatial correlations are neglected.

Spatio-temporal: More sophisticated methods exploit both spatial and temporal correlations, such as simple adaptive weighted local averaging [101], 3-D order-statistic algorithms [106], 3-D Kalman filtering [69] and 3-D Markov random models [105].

Although a third alternative could be to do 2-D spatial denoising for each video frame separately, utilizing the vast number of existing image denoising techniques, almost no work in this direction showed success mainly because 2-D denoising tends to blur spatial details.

Although there have been many papers addressing the applications of wavelet transforms to image denoising, comparatively few have addressed wavelet-based *video* denoising. Roosmalen *et.al.* [107] proposed video denoising by thresholding the coefficients of a specific 3-D wavelet representation and Selesnick *et.al.* [108] found an *efficient* 3-D orientation-selective wavelet transform, 3-D *complex* wavelet transforms, which avoided the time-consuming motion estimation process. The main drawbacks of the 3-D wavelet transforms include long time latency and an inability to adapt to fast motions.

In most video processing applications a long latency is unacceptable, so recursive approaches are widely employed. Pizurica *et al.* [2] proposed sequential 2-D spatial and 1-D temporal denoising, in which they first do sophisticated wavelet-based image denoising for each frame and then recursive *temporal* averaging. However, 2-D spatial filtering tends to introduce artifacts and to remove *weak* details with noise.

In this chapter, we propose wavelet-domain recursive video denoising. We use a 2-D spatial wavelet, and filter recursively to preserve low latency. But unlike [2], our temporal filtering works on the wavelet coefficients themselves (instead of on the spatial pixels).

In such a way we minimize spatial blurring to preserve the *weak* spatial details in still areas. The key to processing directly in the wavelet domain is an efficient shift-invariant transform, allowing spatial motion to be meaningfully reflected in the wavelet coefficients. We study several schemes for robust wavelet-domain motion estimation, and then proceed to motion-compensated temporal filtering in the *wavelet domain* using an adaptive Kalman filter.

7.1 Problem formulation

In standard wavelet-based image denoising [9],[19],[76] the 2-D wavelet transform is used to get a compact representation. Thus it would seem natural to select 3-D wavelets for video denoising [107, 108]. However, there are compelling reasons to choose a 2-D spatial wavelet transform with recursive temporal filtering for video denoising:

1. There is a clear asymmetry between the spatial and temporal axes, in terms of correlation and resolution. A recursive approach is naturally suited to this asymmetry, whereas a 3-D wavelet is not.
2. Recursive filtering can significantly reduce time delay and memory requirements.
3. For autoregressive models the optimal estimator can be achieved recursively.
4. Motion information can be efficiently exploited with recursive filtering.

A. Problem formulation

The video denoising problem can be modeled as follows: given video measurements y with spatial indices i, j and temporal index k

$$y(i, j, k) = x(i, j, k) + v(i, j, k), \quad i, j = 1, 2, \dots, N, \quad k = 1, 2, \dots, M \quad (7.1)$$

corrupted by i.i.d Gaussian noise v , we need to estimate the true image sequence x . Define $\mathbf{x}(k)$, $\mathbf{y}(k)$ and $\mathbf{v}(k)$ to be the column-stacked images at time k , then (7.1) becomes

$$\mathbf{y}(k) = \mathbf{x}(k) + \mathbf{v}(k), \quad k = 1, 2, \dots, M \quad (7.2)$$

We propose to do denoising in the wavelet domain. Let H be the 2-D wavelet transform matrix, then (7.2) is transformed as

$$H\mathbf{y}(k) = H\mathbf{x}(k) + H\mathbf{v}(k) \quad (7.3)$$

Denoting wavelet coefficient vector as $\mathbf{y}_H(k)$, (7.3) is rewritten as

$$\mathbf{y}_H(k) = \mathbf{x}_H(k) + \mathbf{v}_H(k) \quad (7.4)$$

Since we seek a recursive temporal filter we assert an autoregressive form for the signal model

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{w}(k+1) \quad (7.5)$$

thus

$$\mathbf{x}_H(k+1) = A_H \cdot \mathbf{x}_H(k) + B_H \cdot \mathbf{w}_H(k+1) \quad (7.6)$$

where $A_H \equiv HAH^{-1}$, $B_H \equiv HBH^{-1}$. It should be noted the wavelet domain state model still has an autoregressive form. Therefore, *optimal* filtering can be achieved in a recursive way in the wavelet domain.

B. An example: recursive image filtering in the spatial and wavelet domains

As a quick proof of principle, we can denoise 2-D images using a recursive 1-D wavelet procedure, as analogous to denoising 3-D video using 2-D wavelets. We do not propose this as a superior approach to image denoising, rather as a measure of promise in the video case. We use an autoregressive image model and apply a 1-D wavelet transform to each column, followed by recursive filtering column by column, as in Chapter 5. We assess estimator performance in the sense of relative increase of MSE:

$$\delta_{MSE} = \frac{MSE - MSE_{optimal}}{MSE_{optimal}} \quad (7.7)$$

where $MSE_{optimal}$ is the MSE of the optimal recursive estimator.

We use a common image model

$$x(i, j) = \rho_v x(i-1, j) + \rho_h x(i, j-1) - \rho_v \rho_h x(i-1, j-1) + w(i, j), \quad \rho_h = \rho_v = 0.95 \quad (7.8)$$

which is a causal MRF model and can be converted to a vector autoregressive model [75]. The optimal recursive filtering requires the joint processing of entire columns. As this would be completely impractical in the video case, for reasons of computational complexity we recursively filter the wavelet coefficients *independently*, ignoring inter-coefficient relationships. As shown in Table 7.1, scalar processing in the wavelet domain leads to only very modest increase in MSE relative to optimum, whereas this is not at all the case in the spatial domain. It should be noted that the wavelet-based scalar processor is comparable to the optimal filter when the $PSNR > 10dB$, a condition satisfied by most practical applications.

SNR(dB)	10.0	0	-10.0
δ_{MSE} (spatial)	99.1%	209.6%	91.6%
δ_{MSE} (wavelet)	8.2%	21.1%	32.3%

Table 7.1: Percentage increase in δ_{MSE} , the estimation error relative to the optimal estimator. Estimating wavelet coefficients independently introduces only small errors when the SNR exceeds $10dB$ (about 8%). However, doing so in the spatial domain leads to large increases in error.

7.2 The Denoising System

The success of the above 1-D recursive wavelet image denoising motivates the extension to the video case: 2-D recursive wavelet video denoising. The block diagram of the proposed video denoising system is illustrated in Fig. 7.1. There are three crucial aspects: (1) the choice of a 2-D wavelet transform, (2) wavelet-domain motion estimation, and (3) the recursive filtering applied to the motion-compensated wavelet coefficients. These steps are detailed below:

2-D wavelet transform: We apply a 2-D wavelet transform to each frame, rather than a 3-D transform for the whole image sequence, and the coefficients are then filtered recursively.

A huge number of wavelet transforms have been developed (e.g., orthogonal/non-orthogonal, real-valued/complex-valued, decimated/redundant). For image denoising problems, the desired criteria are

1. Shift invariance: to suppress frequency aliasing, and thus artifacts;

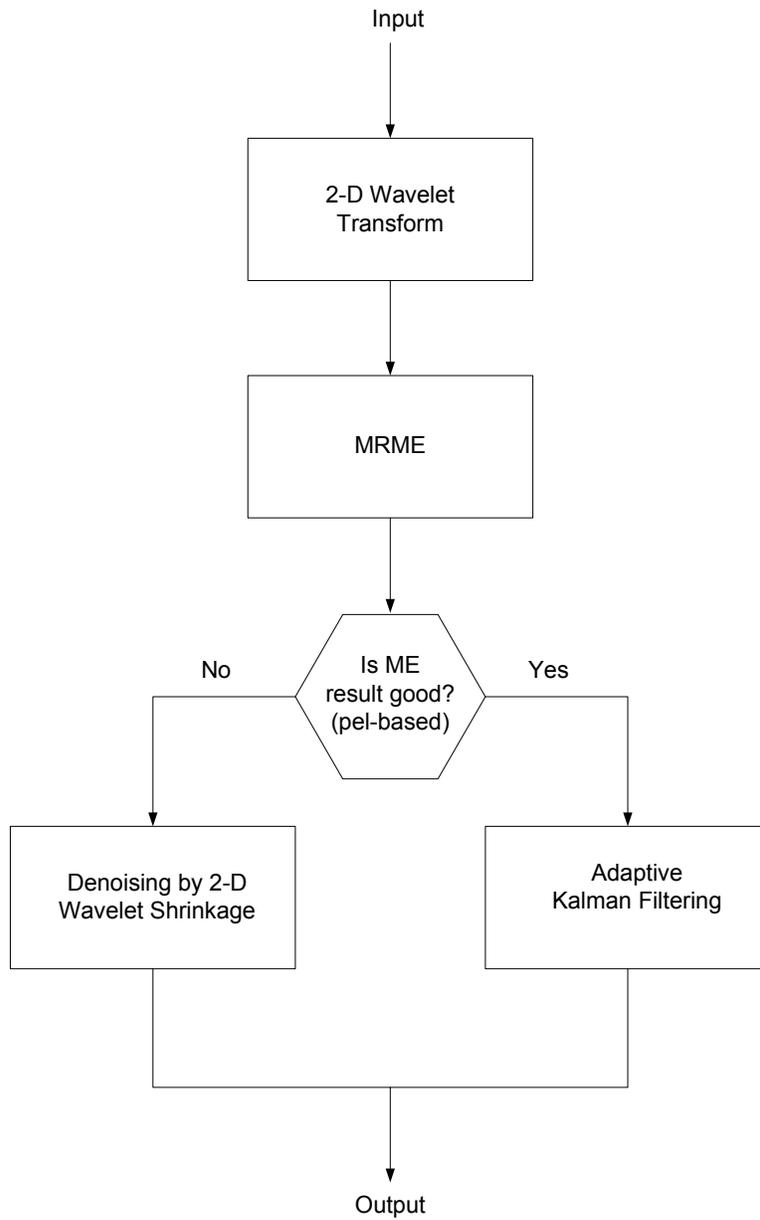


Figure 7.1: The video denoising system

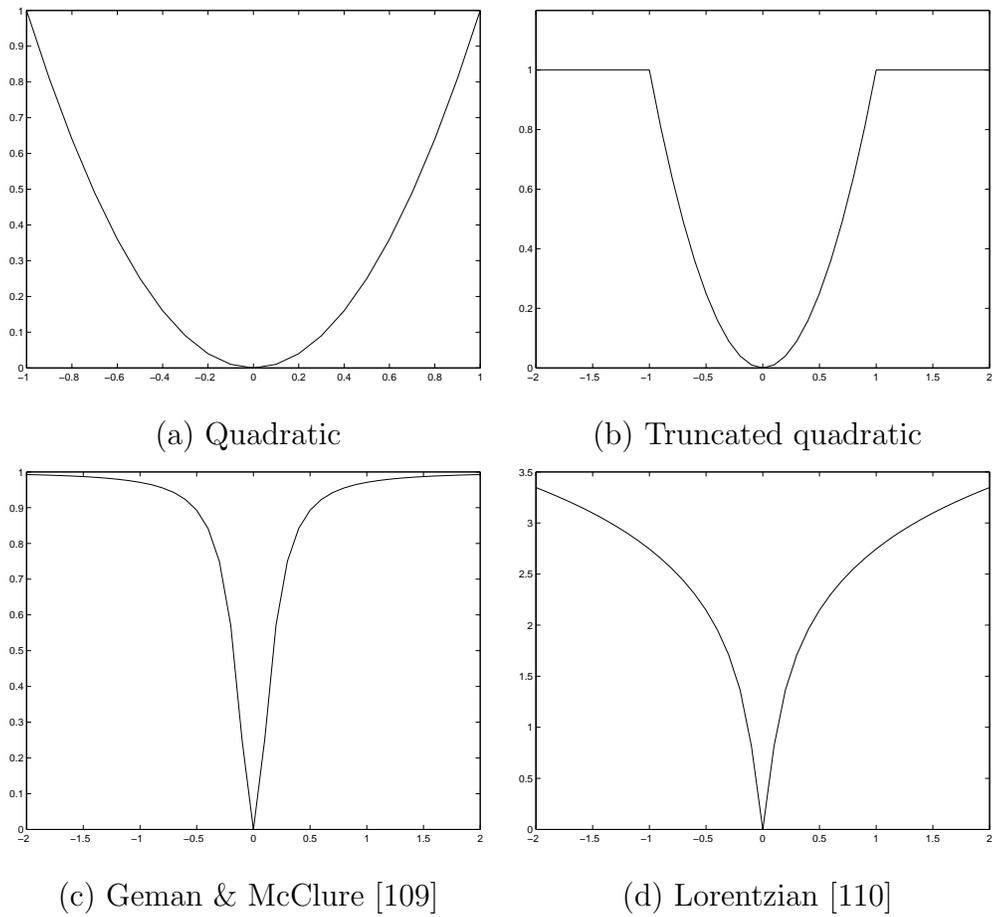


Figure 7.2: Examples of commonly-used penalty functions.

2. Direction selectivity: of importance when an image has dominant oriented features;
3. Low complexity.

The 2-D dual-tree complex wavelet proposed by Kingbury [108] satisfies these requirements very well. Unfortunately it is not convenient for motion estimation since the motion information is related to the coefficient phase, which is a non-linear function of translation. Alternatively, specially designed 2-D wavelet transforms (e.g., curvelet, contourlet) are sensitive to feature directions but are too complex for computation. In this paper, we choose to use an over-complete wavelet representation proposed by Mallat *et. al.* [39]. Although this wavelet representation does not have very good directional selectivity (e.g., it mixes 45-degree and -45 -degree features), several researchers have used it for natural image denoising with impressive results (e.g. [20],[2]).

Multiresolution Motion Estimation (MRME): We utilize the well-used block matching (BM) technique for motion estimation. Compared with other motion estimation approaches, such as optical flow and pixel-recursive methods, block matching is straightforward to compute and less sensitive to noise.

The single-resolution block matching approaches have been studied extensively [105] and are successfully used in modern video compression standards. Multi-resolution block matching (MRBM) was first proposed by Zhang *et. al.* [105] for wavelet-based video coding, and recent developments can be found in [111]. The basic idea of standard MRBM is to start block matching at the coarsest level, using this estimate as a prediction for the next finer scale. An updating step then takes place around the prediction. Supposedly, computational complexity can be reduced in this prediction-updating scheme.

Oddly, a critically decimated wavelet was used [111], which implies that the motion is not a constant function of scale, due to the shift-variance of critically decimated wavelets. A much more sensible choice of wavelet, used in this section, is the overcomplete transform which is shift-invariant, leading to consistent motion as a function of scale, except in the vicinity of motion boundaries. Clearly, this inter-scale relationship of motion should be exploited to improve accuracy. We developed a sequence of four approaches of increasing accuracy:

1. Traditional MRME scheme [111]. This approach has the lowest complexity. However, the performance depends heavily on the estimation results at the coarsest level. In the presence of strong noise, the results are often quite irregular and unreliable.
2. Block matching separately on each level, combined by median filtering.
3. Joint block matching simultaneously at all levels:

Let $\epsilon^l(\mathbf{i}, \mathbf{v})$ denote the displaced frame difference (DFD) at position \mathbf{i} of level l with displacement \mathbf{v}). Then the total DFD over all levels is defined as

$$\epsilon(\mathbf{i}, \mathbf{v}) = \sum_{l=1}^J \epsilon^l(\mathbf{i}, \mathbf{v}) \quad (7.9)$$

and the displacement field $\mathbf{v}(\mathbf{i}) = [v_x(\mathbf{i}), v_y(\mathbf{i})]$ is found by minimizing $\epsilon(\mathbf{i}, \mathbf{v})$.

4. Block matching with smoothness constraint:

The above schemes did not assert any spatial smoothness or correlation in the motion vectors, which we expect in real-world sequences. This is of considerable importance when the additive noise levels are large, leading to irregularly estimated motion

vectors. Therefore, we introduce an additional smoothness constraint and perform BM by solving the optimization problem

$$\arg \min_{\mathbf{v}} \left\{ \sum_{\mathbf{i}} \left[\gamma \cdot \epsilon(\mathbf{i}, \mathbf{v}) + \rho(v_x(\mathbf{i}) - \frac{1}{M} \sum_{m \in N_b(\mathbf{i})} v_x(\mathbf{i} + \mathbf{m})) + \rho(v_y(\mathbf{i}) - \frac{1}{M} \sum_{n \in N_b(\mathbf{i})} v_y(\mathbf{i} + \mathbf{n})) \right] \right\} \quad (7.10)$$

where $N_b(i)$ is the neighborhood set of element \mathbf{i} and M is the number of elements in N_b . γ controls the tradeoff between $\epsilon(\mathbf{i}, \mathbf{v})$, from (7.9), and smoothness. $\rho(\cdot)$ denotes a penalty function, such as those in Fig. 7.2, which controls the deviation of the reference motion vectors from their neighbor averages. The quadratic function in Fig. 7.2(a) leads to solving a linear equation set, which is relatively straightforward to compute. However, since penalty always increases with deviation, motion edges tend to be blurred. To handle motion discontinuities properly, techniques of robust statistics have been studied [112],[113] and the penalty functions shown in Fig. 7.2(b),(c),(d) are among the most often used. All of these result in nonlinear optimization problems and specific optimization methods must be used [114]. In our experiments we used a gradient-based optimization method, applied to the Geman-McClure [115] and Lorentzian [110] functions, both of which gave similar results. We did not try the truncated quadratic function since it is non-differentiable.

Experimentally, we find that Approach 4 (the last one) is the most robust to noise and yields reasonable motion estimates. So in the denoising experiments given below, we will use this approach for motion estimation.

Wavelet coefficient filtering The key to our approach is to support both spatial and temporal filtering, as appropriate.

Since the motion estimate may be inaccurate we use an additional threshold controlling whether to apply a Kalman filter. This is actually a motion detection step. Specifically, when the motion information is unambiguous, that is when

$$|x_H(m, n, k) - \eta(x_H(m, n, k - 1))| < 2\sqrt{2}\sigma_{v_H}, \quad (7.11)$$

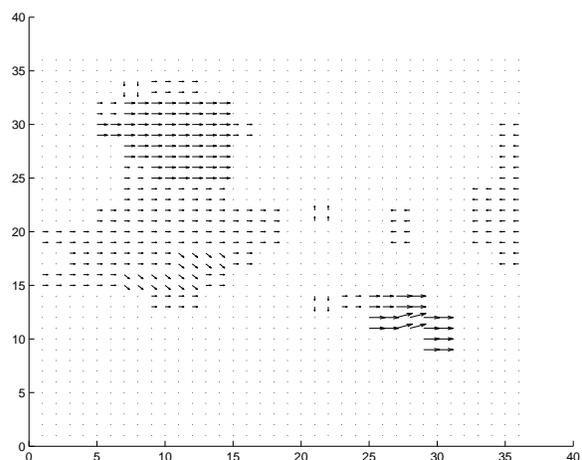
where $\eta(\cdot)$ is the motion compensation function, we restrict the filtering to be purely temporal to avoid any spatial blurring. When motion estimates are poor, that is when

$$|x_H(m, n, k) - \eta(x_H(m, n, k - 1))| \geq 2\sqrt{2}\sigma_{v_H}, \quad (7.12)$$

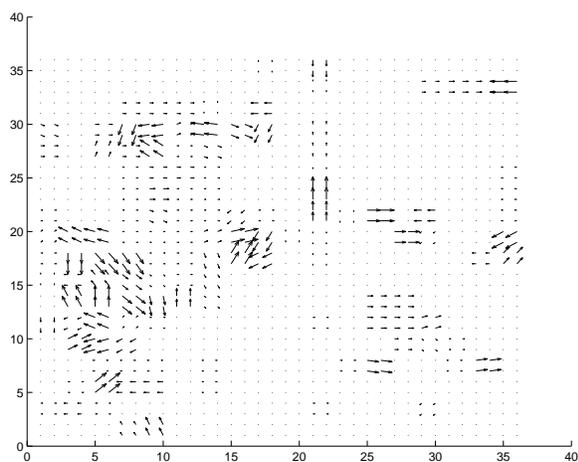
we apply the AWA-based wavelet shrinkage to exploit spatial correlations, which is discussed in Chapter 6. Although this introduces some blurring, wavelet shrinkage is state-of-the-art in static image denoising.

7.3 Experimental results

The proposed denoising approach has been tested using the standard image sequences *Miss America*, *Salesman* and *Paris*, using a three-level wavelet decomposition. First, Fig. 7.3 compares our regularized MRBM approach (7.10) with standard MRBM [111] for the *Paris* sequence. Since the true motion field is unknown we evaluate the performance of noisy motion estimation by comparing with the motion field estimated from noise-free images (Fig. 7.3(b)), and by comparing the corresponding denoising results. The standard MRBM does not exploit any smoothness or prior knowledge of motion fields,

(a) *Paris* sequence

(b) Motion estimated from noise-free sequence



(c) Standard MRBM

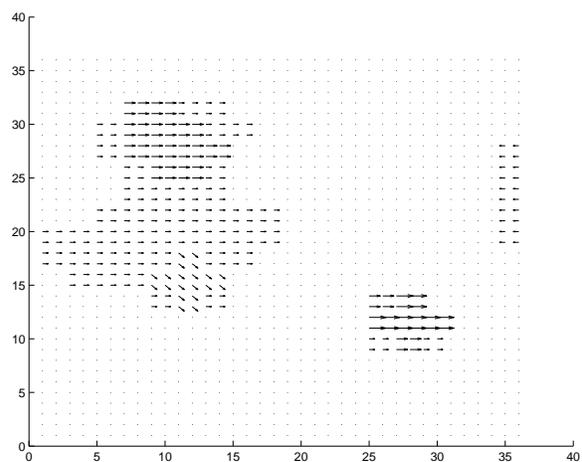
(d) Smoothness-constrained MRBM ($\gamma = 0.01$)

Figure 7.3: Standard MRBM (c) does not exploit any smoothness or prior knowledge of motion fields and performs poorly in the presence of noise. By comparison, our proposed approach (d) gives much better results.

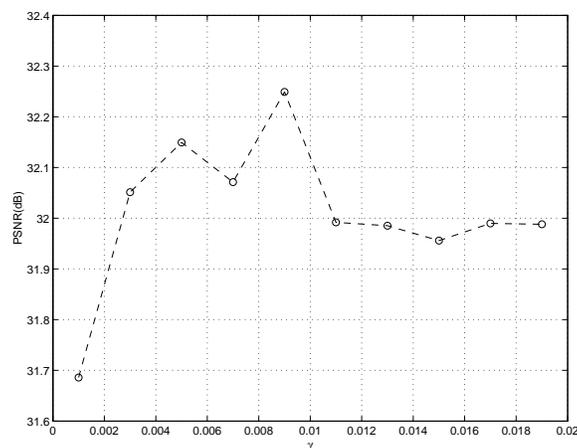


Figure 7.4: Averaged PSNR versus γ curve. PSNR is not sensitive to γ when $0.001 < \gamma < 0.02$.

therefore it performs poorly in the presence of noise (Fig. 7.3(c)). In comparison, our proposed approach gives far superior results (Fig. 7.3(d)). Although our MRBM approach introduces one new parameter γ , experimentally we found PSNR to be weakly dependent on γ ($0.001 < \gamma < 0.02$), as illustrated in Fig. 7.4. In all of the following tests we fix $\gamma = 0.01$.

Next we compare our proposed denoising approach with three recently-published methods: two wavelet-based video denoising schemes [2, 108] and one non-wavelet nonlinear approach [116]. Selesnick *et al.* [108] generalized the ideas of many well-developed $2-D$ wavelet-based image denoising methods and used a complex valued $3-D$ wavelet transform for video denoising. Pizurica *et al.* [2] combined a temporal recursive filter with sophisticated wavelet-domain image denoising, but without motion estimation. Zlokolica *et al.* [116] used multiple class averaging to suppress noise, which performs better than the traditional nonlinear methods, such as the α -trimmed mean filter [117] and the rational filter

PSNR (Original)	28.2	24.6	22.1
PSNR (Proposed method)	33.9	31.6	30.5
PSNR (β -D complex DWT [108])	32.1	30.5	29.3
PSNR (Pizurica [2])	33.7	31.7	30.5
PSNR (Zlokolica [116])	32.5	30.8	29.7

Table 7.2: Comparison of PSNR (in dB) of the proposed method and several other video denoising approaches for the *Salesman* sequence.

[118]. Table 7.2 compares the PSNRs averaged from frames 10 to 30 of the sequence *Salesman* for different noise levels. Our approach yields higher PSNRs than those in [108] and [116], and is comparable to Pizurica’s results which use a sophisticated image denoising scheme in the wavelet domain. However the similar PSNRs between the results of our proposed method and that of Pizurica [2] obscure the significant differences, as made very clear in Figs. 7.5 and 7.6. In particular, we perform less spatial smoothing, shrinking only insignificant coefficients, but rely more heavily upon temporal averaging. Thus our results have very little spatial blurring, preserving subtle textures and fine details, such as the desktop and bookshelf in Fig. 7.5 and the plant in Fig. 7.6.

7.4 Conclusions

We have proposed a new approach to video denoising, combining the power of the spatial wavelet transform and temporal filtering. Most significantly, motion estimation/compensation, temporal filtering and spatial smoothing are *all* undertaken in the wavelet domain. We



(a) Original



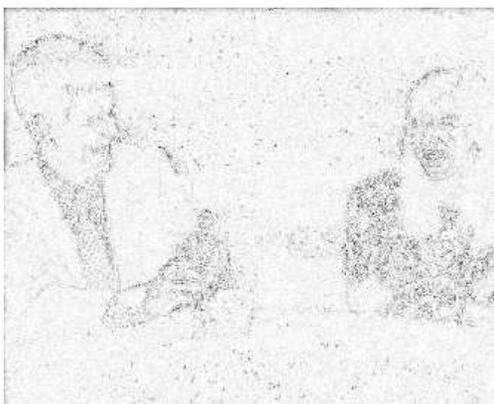
(b) Noisy



(c) Denoised by the proposed approach



(d) Denoised by Pizurica's approach [2]



(e) Absolute difference between (a) and (c)

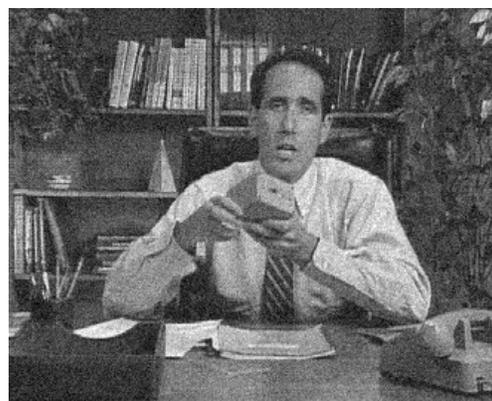


(f) Absolute difference between (a) and (d)

Figure 7.5: Comparison of our proposed (c) and Pizurica's [2] (d) approaches ($\sigma_{v_H} = 15$). Our approach can better preserve spatial details, such as textures on the desktop, as made clear in the difference images in (e) and (f).



(a) Original



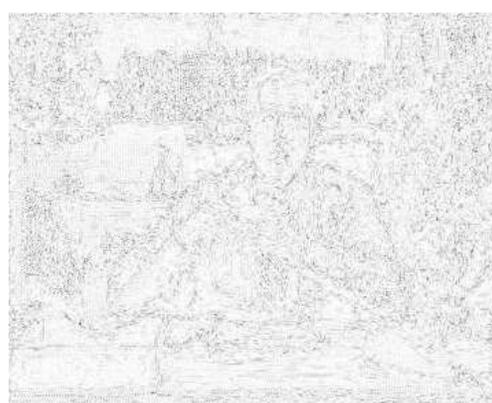
(b) Noisy



(c) Denoised by the proposed approach



(d) Denoised by Pizurica's approach [2]



(e) Absolute difference between (a) and (c) (f) Absolute difference between (a) and (d)

Figure 7.6: Denoising results for *Salesman*. Note in particular the textures of the plants, well preserved in our results (c), but obviously blurred (d).

also avoid spatial blurring by restricting to temporal filtering when motion estimates are reliable, and spatial shrinking only insignificant coefficients when the motion is unreliable. Tests on standard video sequences show that our results yield comparable PSNR to the state-of-the-art methods in the literature, but with considerably improved preservation of fine spatial details. Future improvements may include more sophisticated approaches to spatial filtering, such as that in [2], and more flexible temporal models to better represent image dynamics.

Chapter 8

Thesis Summary

In this thesis we have studied the statistics of wavelet coefficients of representative 1-D and 2-D auto-regressive processes. It is shown that the covariance matrices of wavelet coefficients are sparse with significant correlations only in the intra- and inter-scale neighborhoods. Furthermore, we found experimentally that the coefficients in such a neighborhood carry almost all of the information needed to produce an estimate. By this, we mean that the estimation error from using only the neighborhood coefficients is very close to that using *all* of the coefficients available. This observation suggests significant reduction of complexity and storage demands can be achieved if we perform estimation in the wavelet domain instead of in the original spatial domain. Motivated by this, for image denoising problems we designed and compared the optimal strip Kalman filters under some complexity constraint, in both the spatial and wavelet domains. Our experimental results clearly show two things:

1. The wavelet strip Kalman filters can yield very close results to the optimal Kalman filters, but with much smaller complexity.

2. Unlike the spatial strip Kalman filters, the performance of the wavelet strip Kalman filter is not sensitive to noise strength. It gives reliable estimates over a large range of noise levels. For comparison, the spatial strip Kalman filter depends heavily on noise levels and typically degrades quickly when noise strength increases.

We also proposed a new approach to wavelet-based image enhancement, which is based on estimating the local parameters of singularity points from a wavelet multiscale representation. The absent high-frequency components are inferred from the estimated parameters and the low-frequency information. Because our approach is adaptive locally, it can appropriately enhance edges of various shapes. This makes our approach compares favorably with other approaches in the literature.

In addition to the above results, we also carefully studied the wavelet-based image denoising problem. We found that none of the already-proposed models of wavelet coefficients can properly represent the strong correlation in edge areas, the presence of which causes annoying Gibbs-like artifacts. Therefore, we propose a pre-processing step which can estimate edge strengths and remove them prior to applying the wavelet transform. The edge-removed images have less variation, being closer to stationary, so existing models can well represent them in the wavelet domain. Our denoising results showed that Gibbs-like artifacts have been considerably suppressed.

Although the wavelet-based image denoising has been studied extensively in the literature, there are relatively very few papers on wavelet-based video denoising. We proposed a new multiresolution block matching approach with nonlinear regularization. With robust motion information in hand, we then use a wavelet-domain Kalman filter to remove noise. Compared to other video denoising approaches, experimentally we observed our method

can effectively remove noise while preserving spatial details much better.

Our future work should look to improve the proposed algorithms. Especially, two problems need more research:

1. For image denoising we found the AWA method is not very robust to noise. In (6.18), we simply used the distance between the reference pixel and a pixel in its neighborhood to classify pixels. In the spirit of the K-means algorithm in pattern recognition area, to minimize classification error we should measure the distance between a pattern and the *mean* values of classes. Therefore, we can modify the way to computing the distance in (6.18) as follows:

Initialize the mean value: Use the reference pixel as the initial mean $\mu = y(i, j)$ (see (6.18) for the definitions of notations).

Classification: Classify the pixels $y(p, q)$ in the neighborhood of the reference pixel according to:

$$\begin{aligned} \text{If } [y(p, q) - \mu]^2 < \epsilon^2 & \text{ then put } y(p, q) \text{ into class } C_1 \\ \text{otherwise,} & \text{ put } y(p, q) \text{ into class } C_2 \end{aligned} \quad (8.1)$$

Update the mean value: Update μ by the sample mean of C_1 . Then perform classification again using the new μ . Repeat this procedure until classification results do not change any more.

We expect this new classification method is more noise-resistive.

2. In our approach to video denoising, we assumed the independence model of wavelet coefficients. Supposedly, the correlation model should be able to further improve

performance. The remaining issue is how to estimate the correlation wisely. Clearly, we are not satisfied with the maximum likelihood (ML) estimate (as we did in Ch. 7) because we have strong knowledge about temporal correlation assuming accurate motion compensation. Probably MAP estimation method needs to be studied for this case to replace ML.

In the case of motion estimation, so far we only considered displacements (Eq. 7.10). Obviously, this motion model can not handle more complicated motions, such as rotation and shearing. Therefore, we are now studying to use the affine model in our proposed video denoising algorithm.

Bibliography

- [1] C.Burrus, R.Gopinath, and H. Guo, *Introduction to wavelets and wavelet transforms*, Prentice-Hall, USA, 1998.
- [2] A.Pizurica, V.Zlokolica, and W. Philips, “Combined wavelet domain and temporal denoising,” in *IEEE Internat. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, USA, 2003, pp. 70–81.
- [3] S. Mallat, “A theory for multiresolution signal decomposition: The wavelet representation,” *IEEE Trans. Pattern Recog. and Mach. Intellig.*, vol. 11, no. 7, pp. 674–693, 1989.
- [4] J.Woods and S.O’Neil, “Subband coding of images,” *IEEE Trans. Acous. Speech Signal Proc.*, vol. 35, no. 5, pp. 1278–1288, 1986.
- [5] P.Vaidyanathan and P.Hoang, “Lattice structures for optimal design and robust implementation of two-channel perfect reconstruction filter banks,” *IEEE Trans. Acous. Speech Signal Proc.*, vol. 36, no. 1, pp. 81–94, 1988.
- [6] P.Burt and E.Adelson, “The laplacian pyramid as a compact image code,” *IEEE Trans. Comm.*, vol. 31, no. 5, pp. 532–540, 1983.

- [7] M.Basseville, A.Benveniste, K.Chou, S.Golden, R.Nikoukhah, and A.Willsky, “Medeling and estimation of multiresolution stochastic processes,” *IEEE Trans. Inform. Theory*, vol. 38, no. 3, pp. 766–784, 1992.
- [8] J.C.Brailean and Aggelos K. Katsaggelos, “Simultaneous recursive displacement estimation and restoration of noisy-blurred image sequences,” *IEEE Trans. Image Processing*, vol. 4, no. 9, pp. 1236–1251, 1995.
- [9] S.G. Chang, Y. Bin, and M. Vetterli, “Spatially adaptive wavelet thresholding with context modeling for image denoising,” *IEEE Trans. on Image Proc.*, vol. 9, no. 9, pp. 1532 – 1546, 2000.
- [10] J.M.Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Image Proc.*, vol. 41, no. 2, pp. 3445 – 3462, 1993.
- [11] I.Daubechies, *Ten lectures on wavelets, vol.61 of Proc.CBMS-NSF Regional Conference Series in Applied Math.*, SIAM, Philadelphia, 1992.
- [12] A.Tewfik and M.Kim, “Correlation structure of the discrete wavelet coefficients of fractional brownian motion,” *IEEE Trans. on Info. Theory*, vol. 38, no. 2, pp. 904–909, 1992.
- [13] J.D.Villasenor, B.Belzer, and J.Liao, “Wavelet filter evaluation for image compression,” *IEEE Trans. on Image Proc.*, vol. 4, no. 8, pp. 1053 – 1060, 1995.
- [14] E.J.Candes and D.L.Donoho, “Ridgelets: the key to higher dimensional intermitency?,” *Phil. Trans. Royal Soc. London A*, vol. 357, no. 3, pp. 2495–2509, 1999.

- [15] J.L.Starck, E.J.Candes, and D.L.Donoho, “The curvelet transform for image denoising,” *IEEE Trans. on Image Proc.*, vol. 11, no. 6, pp. 670 – 684, 2002.
- [16] P.Westerink, J.Biemon, and D.Boekee, “An optimal bit allocation algorithm for subband coding,” in *Proc. IEEE Int. Conf. ASSP*, USA, 1987, pp. 1378–1381.
- [17] S.G. Chang, Y. Bin, and M. Vetterli, “Adaptive wavelet thresholding for image denoising and compression,” *IEEE Trans. on Image Proc.*, vol. 9, no. 9, pp. 1522 – 1531, 2000.
- [18] D.Andrews and C.Mallows, “Scale mixtures of normal distributions,” *J.R.Statist.Soc.*, vol. 36, no. 8, pp. 99–120, 1974.
- [19] J.Portilla, V.Strela, M.Wainwright, and E.Simoncelli, “Image denoising using scale mixtures of gaussians in the wavelet domain,” *IEEE Trans. Image Proc.*, vol. 12, no. 11, pp. 1338–1351, 2003.
- [20] M. Malfait and D. Roose, “Wavelet-based image denoising using a markov random field a prior model,” *IEEE Trans. on Image Proc.*, vol. 6, no. 4, pp. 545 – 557, 1997.
- [21] A.Pizurica, W.Philips, I.Lemahieu, and M.Acheroy, “A joint inter- and intrascale statistical model for bayesian wavelet based image denoising,” *IEEE Trans. on Image Proc.*, vol. 11, no. 5, pp. 724–735, 2002.
- [22] M.S.Crouse, R.D.Nowak, and R.G.Baraniuk, “Wavelet-based statistical signal processing using hidden markov models,” *IEEE Trans. on Signal Proc.*, vol. 46, no. 4, pp. 886 – 902, 1998.

- [23] G.Fan and X.Xia, “Wavelet-based texture analysis and synthesis using hidden markov models,” *IEEE Trans. Circuits and Systems I: Fundamental theory and applications*, vol. 50, no. 1, pp. 106–120, 2003.
- [24] M.Hayes, *Statistical digital signal processing and modeling*, John Wiley & Sons, USA, 1996.
- [25] J.Mendel, *Lessons in estimation theory for signal processing, communication and control.*, Prentice Hall, USA, 1995.
- [26] J.Woods and C.Radewan, “Kalman filtering in two dimensions,” *IEEE Trans. on Info. Theory*, vol. 23, no. 4, pp. 473–482, 1977.
- [27] N.Kingsbury, “Image processing with complex wavelets,” *Phil. Trans. Royal Society London A*, vol. 357, pp. 2543–2560, 1999.
- [28] I.Selesnick, “The design of approximate hilbert transform pairs of wavelet bases,” *IEEE Trans. on Signal Proc.*, vol. 50, no. 5, pp. 1144 – 1152, 2002.
- [29] J.Magarey and N.Kingsbury, “Motion estimation using a complex-valued wavelet transform,” *IEEE Trans. on signal Proc.*, vol. 46, no. 4, pp. 1069 – 1084, 1998.
- [30] W.Lawton, “Applications of complex valued wavelet transforms to subband decomposition,” *IEEE Trans. on Signal Proc.*, vol. 41, no. 12, pp. 3566 – 3568, 1993.
- [31] B.Belzer; J.-M.Lina; J.Villasenor, “Complex, linear-phase filters for efficient image coding,” *IEEE Trans. on signal Proc.*, vol. 43, no. 10, pp. 2425 – 2427, 1995.
- [32] N.Kingsbury, “The dual-tree complex wavelet transform: a new technique for shift invariance and directional filters,” in *Proc. 8th IEEE DSP Workshop*, USA, 1998.

- [33] N.Kingsbury, “The dual-tree complex wavelet transform: a new tool for image restoration and enhancement,” in *Proc. EUSIPCO*, Greece, 1998, pp. 319–322.
- [34] S.Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, USA, 1998.
- [35] D.Donoho and I.Johnstone, “Ideal spatial adaptation via wavelet shrinkage,,” *Biometrika*, vol. 81, no. 9, pp. 425–455, 1994.
- [36] P.F.C.de Rivaz and N.G.Kingsbury, “Complex wavelet features for fast texture image retrieval,” in *Proc. IEEE ICIP*, Japan, 1999, pp. 201–204.
- [37] E.P.Simoncelli, W.T.Freeman, E.H.Adelson, and D.J.Heeger, “Shiftable multiscale transforms,” *IEEE Trans. on Information Theory.*, vol. 38, no. 2, pp. 587 – 607, 1992.
- [38] R.R.Coifman and D.L.Donoho, *Translation-invariant denoising in Wavelets and Statistics*, Springer-Verlag, Berlin,Germany, 1995.
- [39] S. Mallat and S. Zhong, “Characterization of signals from multiscale edges,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no. 7, pp. 710 – 732, 1992.
- [40] P.Abry and P.Flandrin, “Multiresolution transient detection,” in *Proc. IEEE-SP Int. Symp. Time-freq. Time-scale Anal.*, Chicago,USA, 1994, pp. 225–228.
- [41] E.Ozturk, O.Kucur, and G.Atkin, “Waveform encoding of binary signals using a wavelet and its hilbert transform,” in *Proc. ICASSP*, 2000.
- [42] W.T.Freeman and E.H.Adelson, “The design and use of steerable filters,” *IEEE Trans. on Pattern Anal. Machine Intell.*, vol. 13, no. 9, pp. 891–906, 1991.

- [43] G.Beylkin and B.Torresani, "Implementation of operators via filter banks: autocorrelation shell and hardy wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 164–185, 1996.
- [44] M.N.Do and M.Vetterli, "The finite ridgelet transform for image representation," *IEEE Trans. on Image Proc.*, vol. 12, no. 1, pp. 16–28, 2003.
- [45] M.N.Do and M.Vetterli, "Contourlets: a directional multiresolution image representation," in *Proc. Intl. Conf. Image Processing, 2002*, USA, 2002, pp. 22–25.
- [46] L.Sendur and I.W.Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Trans. on Signal Proc.*, vol. 50, no. 11, pp. 2744 – 2756, 2002.
- [47] R.Gonzalez and R.Woods, *Digital image processing*, Addison-Wesley, USA, 1992.
- [48] A.Jain, *Fundamentals of digital Image Processing*, Prentice-Hall, USA, 1989.
- [49] G.McLean and E.Jernigan, "Indicator functions for adaptive image processing," *Optical Engineering*, vol. 29, no. 1, pp. 141–157, 1991.
- [50] M.Sundareshan, T.Ji, and H.Roehrig, "Adaptive image contrast enhancement based on human visual properties," *IEEE Trans. Med. Imag.*, vol. 13, no. 12, pp. 168–178, 1994.
- [51] S.Thurnohofer and S.Mitra, "Edge-enganced image zooming," *Optical Engineering*, vol. 35, no. 7, pp. 1862–1870, 1996.

- [52] S.Mitra, H.Li, I.Lin, and T.Yu, “A new class of nonlinear filters for image enhancement,” in *Proc. IEEE Int. Conf. Acoust. Speech, signal Proc.*, USA, 1991, pp. 2525–2528.
- [53] H.Greenspan, C.Anderson, and S.Akber, “Image enhancement by nonlinear extrapolation in frequency space,” *IEEE Trans. Image Processing*, vol. 9, no. 6, pp. 1035–1047, 2000.
- [54] P.Burt and R.Kolczybski, “Enhanced image capture through fusion,” in *Proc. Int. Conf. Computer Vision*, Germany, 1993, pp. 173–182.
- [55] K.Kinebuchi, D.Muresan, and T.Parks, “Image interpolation using wavelet-based hidden markov trees,” in *IEEE Int. Conf. Acoust. Speech Signal Proc.*, USA, 2001, pp. 7–11.
- [56] D.Woo, I.Eom, and Y.Kim, “Image interpolation based on inter-scale dependency in wavelet domain,” in *submitted to IEEE Int. Conf. Image Proc.*, Singapore, 2004, pp. ?–?
- [57] M.Crouse, R.Nowak, and R.Baraniuk, “Wavelet-based statistical signal processing using hidden markov models,” *IEEE Trans. Signal Processing*, vol. 46, no. 4, pp. 886–902, 1998.
- [58] J.Romberg, H.Choi, and R.Baraniuk, “Bayesian tree-structured image modeling using wavelet-domain hidden markov models,” *IEEE Trans. Image Proc.*, vol. 10, no. 7, pp. 1056–1068, 2001.

- [59] W.Carey, D.Chuang, Sheila, and S.Hemami, “Regularity-preserving image interpolation,” *IEEE Trans. Image Processing*, vol. 8, no. 9, pp. 1293–1297, 1999.
- [60] J.Canny, “A computational approach in edge detection,” *IEEE Trans. Patt.Anal.Machine Intell.*, vol. 8, no. 6, pp. 679–698, 1986.
- [61] S.Mallat and W.Hwang, “Singularity detection and processing with wavelets,” *IEEE Trans. Inform. Theory*, vol. 38, no. 4, pp. 617–643, 1992.
- [62] Y. Meyer, *Wavelets and operators*, Cambridge University Press, England, 1992.
- [63] A.Witkin, “Scale space filtering,” in *Proc. Int. Conf. Artificial Intelligence*, USA, 1983, pp. 625–631.
- [64] D.Marr and E.Hildreth, “Theory of edge detection,” *Proc. Royal Soc.*, vol. 207, no. 6, pp. 187–217, 1980.
- [65] G.Beylkin, R.Coifman, and R.Rokhlin, “Fast wavelet transforms and numerical algorithms,” *Comm. Pure and Appl. Math.*, vol. 44, no. 9, pp. 141–183, 1991.
- [66] P.Flandrin, “Wavelet analysis and synthesis of fractional brownian motion,” *IEEE Trans. Inform. Theory*, vol. 38, no. 3, pp. 910–916, 1992.
- [67] J.Zhang, D.Wang, and Q.Tran, “Wavelet-based multiresolution stochastic models,” *IEEE Trans. Image Proc.*, vol. 7, no. 11, pp. 1621–1626, 1998.
- [68] A.Katsaggelos, R.Kleihorst, S.Efstratiadis, and R.Lagendijk, “Adaptive image sequence noise filtering methods,” in *Proc. SPIE Visual Comm. and Image Proc.*, USA, 1991, pp. 716–727.

- [69] J.Kim and J.Woods, “Spatio-temporal adaptive 3-d kalman filter for video,” *IEEE Trans. on Image Proc.*, vol. 6, no. 3, pp. 414–424, 1997.
- [70] S.Golden, *Identifying multiscale statistical models using the wavelet transform*, MIT, USA, 1991.
- [71] G.Golub and C.Loan, *Matrix computations*, Johns Hopkins, USA, 1996.
- [72] P.Fieguth, “Multiply-rooted multiscale models for large scale estimation,” *IEEE Trans. on Image Proc.*, vol. 10, no. 11, pp. 1676–1686, 2001.
- [73] G.Wornell, “Wavelet-based representations for the 1/f family of fractal processes,” *Proceedings of IEEE*, vol. 81, no. 10, pp. 1428–1450, 1993.
- [74] J.Liu and P.Moulin, “Information-theoretic analysis of interscale and intrascale dependencies between image wavelet coefficients,” *IEEE Trans. Image Proc.*, vol. 10, no. 11, pp. 1647–1658, 2001.
- [75] A.Rosenfeld and A.Kak, *Digital picture processing.*, Academic Press, USA, 1982.
- [76] M.Mihcak, I.Kozintsev, and K.Ramchandran, “Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, USA, 1999, pp. 3253–3256.
- [77] D.Angwin and H.Kaufman, *Nonhomogeneous image identification and restoration procedures (in the book Digital Image Processing)*, Springer-verlag, Germany, 1991.
- [78] R.Gallager, *Information theory and reliable communications*, John Wiley & Sons, USA, 1968.

- [79] D.Donoho and I.Johnstone, “Adapting to unknown smoothness via wavelet shrinkage,” *J.Amer.Stat.Assoc.*, vol. 90, no. 12, pp. 1200–1224, 1995.
- [80] J.Pesquet, H.Krim, and E.Hamman, “Bayesian approach to best basis selection,” in *IEEE Int. Conf. ASSP*, USA, 1996, pp. 2634–2637.
- [81] H.Chipman, E.Kolaczyk, and R.McCulloch, “Adaptive bayesian wavelet shrinkage,” *J.Amer.Stat.Assoc.*, vol. 92, no. 7, pp. 1430–1442, 1997.
- [82] N.Lee, Q.Huynh, and S.Schwarz, “New methods of linear time-frequency analysis for signal detection,” in *IEEE Int. Symp. Time-Freq. Time-Scale Anal.*, USA, 1996, pp. 26–29.
- [83] D.Taubman and M.Marcellin, *JPEG2000: Image compression fundamentals*, Kluwer, USA, 2002.
- [84] T.Ebrahimi and F.Pereira, *The MPEG book*, Prentice Hall, USA, 2002.
- [85] D. T. Kuan, A. A. Sawchuk, T. C. Strand, and P. Chavel, “Adaptive noise smoothing filter for images with signal-dependent noise,” *IEEE Trans. Pattern Recog. and Mach. Intellig.*, vol. 6, no. 5, pp. 373–383, 1985.
- [86] J. S. Lee, “Digital image enhancement and noise filtering by use of local statistics,” *IEEE Trans. Pattern Recog. and Mach. Intellig.*, vol. 2, no. 5, pp. 165–168, 1980.
- [87] F.Abramovich and T.Sapatinas, “Wavelet thresholding via a bayesian approach,” *Journal of Royal Stat. Soc., Ser.B*, vol. 60, no. 3, pp. 725–749, 1998.
- [88] J.lee, “Digital image smoothing and the sigma filter,” *Computer Vision, Graphics and Image Processing*, vol. 24, no. 7, pp. 255–269, 1983.

- [89] S. LoPresto, K. Ramchandran, and M.Orchard, “Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework,” in *Proc. Data Compression Conf.*, USA, 1997, pp. 321–230.
- [90] E.Simoncelli and E.Adelson, “Noise removal via bayesian wavelet coring,” in *Proc. IEEE Int. Conf. Image Proc.*, Switzerland, 1996, pp. 379–382.
- [91] L.Zhang, P.Bao, and X.Wu, “Hybrid inter- and intra-wavelet scale image restoration,” *Pattern Recognition*, vol. 36, no. 8, pp. 1737–1746, 2003.
- [92] M.Banham and A.Katsaggelos, “Spatially adaptive wavelet-based multiscale image restoration,” *IEEE Trans. Image Proc.*, vol. 5, no. 4, pp. 619–634, 1996.
- [93] J.Liu and P. Moulin, “Image denoising based on scale-space mixture modeling for wavelet coefficients,” in *Proc. IEEE Int. Conf. Image Proc.*, Japan, 1999, pp. 386–390.
- [94] J. Scharcanski, C.Jung, and R. Clarke, “Adaptive image denoising using scale and space consistency,” *IEEE Trans. Image Proc.*, vol. 11, no. 9, pp. 1092–1101, 2002.
- [95] G.Fan and X.Xia, “Improved hidden markov models in the wavelet domain,” *IEEE Trans. Signal Proc.*, vol. 49, no. 1, pp. 115–120, 2001.
- [96] M.Jansen and A.Bultheel, “Empirical bayes approach to improve wavelet thresholding for image noise reduction,” *J.Amer.Statist.Assoc.*, vol. 96, no. 454, pp. 629–639, 2001.
- [97] F.Jin, P.Fieguth, and L.winger, “Image denoising using complex wavelet and markov prior models,” in *Proc. IEEE Int. Conf. Image Proc.*, Singapore, 2004, p. submitted.

- [98] D.Donoho, “Denoising by soft-thresholding,,” *IEEE Info. Theory*, vol. 41, no. 5, pp. 613–627, 1995.
- [99] D.Donoho and I.Johnstone, “Wavelet shrinkage: Asymptopia?,” *J.R.Stat.Soc.B*, vol. 57, no. 2, pp. 301–369, 1995.
- [100] F.Jin, P.Fieguth, L.winger, and Ed. Jernigan, “Adaptive wiener filtering of noisy images and image sequences,” in *Proc. IEEE Int. Conf. Image Proc.*, Spain, 2003, pp. 349–352.
- [101] M.Ozkan, I.Sezan, and A.Tekalp, “Adaptive motion-compensated filtering,” *IEEE Trans. on Circuits Syst. Video Technol.*, vol. 3, no. 8, pp. 277–290, 1993.
- [102] J.Buf and T.Campbell, “A quantitative comparison of edge-preserving smoothing techniques,” *Signal Processing*, vol. 21, no. 10, pp. 289–301, 1990.
- [103] N.Kingsbury, “A dual-tree complex wavelet transform with improved orthogonality and symmetry properties,” in *Proc. IEEE Int. Conf. Image Proc.*, Canada, 2000, pp. 3253–3256.
- [104] G.Box and C.Tiao, *Bayesian inference in statistical analysis*, Addison-Wesley, USA, 1992.
- [105] J.Brailean, R.Kleihorst, S.Efstratiadis, A.Katsaggelos, and A.Lagendijk, “Noise reduction filters for dynamic image sequences: A review,” *Proc. of IEEE*, vol. 83, no. 9, pp. 1272–1292, 1995.
- [106] G.Arce, “Multistage order statistic filters for image sequence processing,” *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 39, no. 5, pp. 1147–1163, 1991.

- [107] P.Roosmalen, S.Westen, R.Lagendijk, and J.Biemond, “Noise reduction for image sequences using an orientated pyramid thresholding technique,” in *Proc. Int. Conf. Image Proc.*, USA, 1996, pp. 375–378.
- [108] I.Selesnick and K.L.Li, “Video denoising using 2d and 3d dual-tree complex wavelet transforms,” in *Proc. SPIE Wavelet Applications in Signal and Image Processing X.*, USA, 2003, pp. 700–707.
- [109] S.Geman and D.McClure, “A nonlinear filter for film restoration and other problems in image processing,” *Computer Vis., Graphics, Image Proc.*, vol. 4, no. 7, pp. 281–289, 1992.
- [110] A.Blake and A.Zisserman, *Visual reconstruction*, The MIT Press, USA, 1987.
- [111] J.Zan, M.Ahmad, and S.Swamy, “New techniques for multi-resolution motion estimation,” *IEEE Trans. on Circuits and Systems: Videp technology*, vol. 12, no. 9, pp. 793–802, 2002.
- [112] M.Black and P.Anandan, “The robust estimation of multiple motion: parametric and piecewise-smooth flow fields,” *Computer Vision and Image Understanding*, vol. 63, no. 1, pp. 75–104, 1996.
- [113] P.Huber, *Robust statistics*, Wiley, USA, 1981.
- [114] S.Nash and A.Sofer, *Linear and nonlinear programming*, McGraw-Hill, USA, 1996.
- [115] S.Geman and D.McClure, “Statistical methods for tomographic image reconstruction,” *Bull.Int.Statist. Inst.*, vol. 4, no. 4, pp. 5–21, 1987.

- [116] Zlokolica V. and Philips W., “Motion and detail adaptive denoising of video,” in *IS&T/SPIE Symposium on Electronic Imaging*, USA, 2004, pp. 378–381.
- [117] Bednar J. and Wat T., “Alpha-trimmed means and their relationship to median filters,” *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 32, no. 2, pp. 145–153, 1984.
- [118] F.Cocchia, S.Carrato, and G.Ramponi, “Design and real-time implementation of a 3-d rational filter for edge-preserving filtering,” *IEEE Trans. Consumer Electronocs.*, vol. 43, no. 4, pp. 1291–1300, 1997.