

Wavelet Methods for Time Series and Spatial Data

Idris Arthur Eckley

September 2001



A thesis submitted to the University of Bristol in accordance
with the requirements of the degree of Doctor of Philosophy
in the Faculty of Science.

Abstract

This thesis considers the application of wavelet methods to the analysis of time series and spatial data. In the first part, we propose a locally stationary model of the covariance structure for data which lie on a regular grid. This is achieved by moving from a (global) Fourier decomposition of structure to a *localised* decomposition involving a set of discrete, non-decimated wavelets. The proposed model is subsequently applied to various texture analysis problems. These range from the classification of images taken from the standard Brodatz texture collection to subtle discrimination problems encountered by an industrial collaborator.

The second part proposes an efficient construction of the inner product matrix of discrete autocorrelation wavelets — a quantity which is of crucial importance in the unbiased estimation of local wavelet spectra. The proposed scheme relates neighbouring elements of the matrix which lie on a given diagonal using a two-scale relationship of the autocorrelation wavelets. This results in a construction which is considerably more efficient than the brute force approach used to date.

Finally, we conclude by detailing the results of initial research on the estimation of the local autocovariance structure of locally stationary time series. These results provide an interesting interpretation of this quantity in terms of familiar (stationary) time series measures.

Acknowledgements

I would like to thank the Statistics group at the University of Bristol for providing such a stimulating environment in which to work. In particular, I would like to thank my advisor, Guy Nason, for all his help, encouragement and patience over the last three years. It really has been a wonderful experience.

The financial support of the EPSRC and Unilever Research through the CASE scheme is gratefully acknowledged, as is the warmth of the Measurement Sciences group at Unilever Research's Port Sunlight Laboratory where I spent three very happy months. I would especially like to thank Rob Treloar, my industrial advisor at Port Sunlight, whose guidance and humour made my placements so memorable. My thanks must also go to Matt Reed and Francis McGlone.

Many of my teachers have inspired and enthused me throughout my education, none more so than Dafydd Chilton, Sian Beidas and Llifon Jones. Thanks also to Clive Sparkes for all his advice over the years. Jon, Ben, Dave, Ruth and especially Louise — thank you for being there through the good and not quite so good times. I am grateful to you all for being such good listeners!

Finally, I would like to thank my parents whose support and encouragement has been constant from day one. Diolch o galon i chi.

Author's Declaration

I declare that the work in this thesis was carried out in accordance with the Regulations of the University of Bristol. The work is original except where indicated by special reference in the text and no part of the thesis has been submitted for any other degree. Any views expressed in the thesis are those of the author and in no way represent those of the University of Bristol. The thesis has not been presented to any other University for examination either in the United Kingdom or overseas.

Idris A. Eckley

Contents

1	Introduction	1
2	Literature review	3
2.1	Introduction	3
2.2	The Fourier transform	4
2.3	What is a wavelet?	5
2.3.1	Multiresolution analysis	8
2.3.2	Fourier properties of ϕ	11
2.3.3	Deriving a wavelet from a MRA	12
2.4	The discrete wavelet transform	15
2.5	The non-decimated wavelet transform	21
2.5.1	Other wavelet transforms	24
2.6	Multidimensional wavelet transforms	25
2.6.1	Two-dimensional multiresolution analysis	26
2.6.2	Discrete wavelet transforms in two dimensions	27
2.7	Wavelets in time series analysis	32
2.7.1	Introduction	32
2.7.2	Locally stationary time series models	33
2.7.3	Discrete wavelets and their autocorrelations	34
2.7.4	Model specification	36
2.7.5	The evolutionary wavelet spectrum	37
2.8	Wavelet shrinkage and thresholding	38
2.9	Summary	44

3	Locally stationary wavelet processes	45
3.1	Motivation	45
3.2	Discrete wavelets	48
3.3	Locally stationary wavelet random fields	50
3.3.1	Examples of LS2W processes	53
3.4	Measuring local power	56
3.5	A.C. wavelets and the LWS representation	58
3.5.1	The inner product matrix of a.c. wavelets	61
3.5.2	Uniqueness of the LWS	64
3.6	Estimating the LWS	65
3.6.1	Smoothing the local wavelet periodogram	69
3.7	Capturing the autocovariance structure	72
3.7.1	Local autocovariance	73
3.7.2	Invertibility of the autocovariance representation	74
3.8	Summary	75
4	Texture analysis	77
4.1	Introduction	77
4.2	What is texture?	78
4.3	Statistical approaches to texture description	81
4.3.1	Fourier-based approaches	82
4.3.2	Textural edgeness	83
4.3.3	Gray tone co-occurrence matrices	84
4.3.4	Other statistical approaches	84
4.4	Wavelets for texture analysis	86
4.4.1	An introduction to wavelets for texture analysis	86
4.4.2	The discrete wavelet transform	87
4.4.3	Discrete wavelet packet transforms	87
4.4.4	Non-decimated discrete wavelet transforms	89
4.4.5	Other wavelet-based approaches to texture analysis	90
4.4.6	Concluding remarks	91
4.5	The LS2W model and texture analysis	92

4.5.1	The Brodatz experiment	93
4.6	Applications of the LS2W model	99
4.6.1	Exploratory analysis of pilled material images	99
4.6.2	Exploratory analysis of hair images	107
4.6.3	Non-stationary texture classification	111
4.7	Summary	113
5	The inner product matrix of discrete autocorrelation wavelets	115
5.1	Introduction	115
5.2	Autocorrelation wavelets and time series	116
5.2.1	Inner product matrix: brute force construction	118
5.3	Inner product matrix: recursive construction	118
5.3.1	The leading diagonal	119
5.3.2	Computational effort of calculating the leading diagonal recursively	120
5.3.3	General diagonal case	125
5.3.4	Efficiency of adopting the recursive approach for other diagonals	126
5.4	Constructing the inner product matrix top row	131
5.5	Extension to two-dimensions	133
5.6	Concluding remarks	134
6	Local autocovariance estimation	135
6.1	Introduction	135
6.2	Interpreting the CLACV estimator	136
6.3	The CLACV and stationary processes	141
6.4	Summary	145
7	Conclusions and future directions	147
A	Chapter 3 Proofs	151
B	Chapter 5 Proofs	171
C	The LS2W software suite	177
	Bibliography	196

List of Figures

2.1	Fourier series approximation of the box-car function.	6
2.2	Successive approximations of a smooth function containing a single discontinuity using the Haar MRA.	11
2.3	Mother wavelets of a selection of Daubechies' compactly supported orthonormal wavelet bases.	16
2.4	DWT decomposition of a discrete signal \mathbf{c}_j into scaling function (smooth) coefficients \mathbf{c}_{j-1} and wavelet (detail) coefficients \mathbf{d}_{j-1}	19
2.5	DWT of the test function considered in figure 2.2 using Daubechies' Extremal Phase (N=3) wavelets.	19
2.6	DWT reconstruction of a signal \mathbf{c}_{j+1} from the scaling function (smooth) coefficients, \mathbf{c}_j , and wavelet (detail) coefficients, \mathbf{d}_j	20
2.7	Example of the DWT's lack of translation-equivariance.	22
2.8	Example of the translation equivariance of the non-decimated wavelet transform.	23
2.9	Schematic representation of the two dimensional DWT algorithm (after Mallat, 1989b).	29
2.10	Schematic representation of the inverse 2D DWT (after Mallat, 1989b).	29
2.11	Finest scale wavelet coefficients of a Haar DWT of the Lennon image.	30
2.12	Finest scale wavelet coefficients of a Haar NDWT of the Lennon image.	31
2.13	Realisation of the LSW process considered in Section 2.3 and associated EWS.	39
3.1	An image of a face.	47
3.2	Realisations of Haar moving average fields.	55
3.3	Realisation of a non-stationary LS2W process.	56

3.4	Discrete Haar autocorrelation wavelets at scale 5 in the (a) diagonal, (b) vertical and (c) horizontal decomposition directions.	62
3.5	Corrected local wavelet spectra of the non-stationary process.	70
3.6	Uncorrected local wavelet spectra of the non-stationary process.	71
4.1	Sample images taken from the Brodatz (1966) image library.	79
4.2	Example of a multi-textured region.	81
4.3	Frequency-based texture statistics.	83
4.4	Brodatz images used to test the LS2W model's ability to discriminate and classify between different textures.	94
4.5	Plot of the first two linear discriminant axes for LS2W-based features obtained from the Brodatz textures displayed in figure 4.4.	96
4.6	LDA plots for measures of the Brodatz textures based on the (uncorrected) non-decimated wavelet transform.	96
4.7	LDA plots for measures of the Brodatz textures based on the discrete wavelet transform.	97
4.8	LDA plots for measures of the Brodatz textures based on the Fourier transform.	97
4.9	Images of materials pilled to varying degrees. Image (1) contains a fine pill material, whilst image (6) contains heavy pilling. Images provided by Unilever Research.	100
4.10	LDA plot for measures of the Pill images based on the LS2W model.	102
4.11	LDA plot of the first and third discriminant variables for the Pill images based on the LS2W model.	102
4.12	LDA plot of the first and fourth discriminant variables for the Pill images based on the LS2W model.	103
4.13	LDA plots for measures of the Pill images based on the uncorrected, non-decimated wavelet transform.	103
4.14	LDA plots for measures of the Pill images based on the discrete wavelet transform.	104
4.15	LDA plots for measures of the Pill images based on the Fourier transform.	104
4.16	Images of straight hair.	108

4.17	LDA plots for measures of the Hair images based on the LS2W model, NDWT, DWT and Fourier Transform.	109
4.18	Simulated examples of non-stationary textures.	112
5.1	Recursive scheme for (symmetric) inner product matrix A calculation. .	116
6.1	Mean of the CLACV estimate, $\widehat{C}_7(z, 1)$, for 1000 simulations of the MA(3) process.	144
C.1	Scale 2, discrete Haar autocorrelation wavelet in the diagonal decomposition direction.	181
C.2	A realisation of a montage process whose wavelet spectral contributions lie in the vertical direction.	196

List of Tables

3.1	Biased contributions which enter into the LWP estimate of $S_3^v(\mathbf{z})$	67
4.1	Percentage of Brodatz textures classified correctly using various feature extraction techniques.	95
4.2	Percentage of Brodatz textures classified correctly using various feature extraction techniques in conjunction with a two-stage classification scheme.	99
4.3	The first and second discriminant variables for the LS2W-based linear discriminant analysis model of the pill images, constructed using the Daubechies Extremal Phase (N=3) wavelets ($\times 10^{-6}$).	105
4.4	Percentage of Pill textures classified correctly using a single step classification algorithm.	106
4.5	Percentage of Pill textures classified correctly using various feature extraction techniques.	107
4.6	The first discriminant variable for the LS2W-based linear discriminant analysis model of the hair images.	108
4.7	Percentage of hair sub-images classified correctly using various feature extraction techniques (Daubechies Extremal Phase (N=4)).	110
4.8	Percentage of hair sub-images classified correctly using various feature extraction techniques (Haar).	111
4.9	Results of a simulated example: Percentage of tile types classified correctly using the LS2W and NDWT-based classification approaches. .	113
5.1	Order of computations required.	116
6.1	Time-averaged estimates of the mean of 1000 estimates of the CLACV for the MA(3) process defined in equation (6.21).	145

Chapter 1

Introduction

Wavelets are a comparatively new mathematical tool which have received a great deal of interest within the statistical community in recent years. These localised basis functions are capable of both efficiently modelling high frequency signal components, for example discontinuities, and capturing smooth signal structure. Such qualities have resulted in a considerable amount of research into the potential of using wavelets for signal denoising.

More recently, several researchers have considered the application of wavelets to time series analysis, for example the work of Nason, von Sachs & Kroisandt (2000) introduced a model of *locally stationary wavelet* processes to the literature. In contrast to the traditional, frequency-based Cramér representation, the model proposed by Nason *et al.* permits a location-scale decomposition of the covariance structure of time series which appear to be stationary within localised regions, although their form may evolve from one region to another. A review of the key aspects of wavelet theory, together with a survey of its application to time series analysis and signal denoising, is presented in Chapter 2.

Many of the scenes and images which we encounter in our everyday lives possess locally stationary, multiscale structure. Thus, to obtain insights into such processes it would be useful to have a multiscale model for their covariance structure. Such a model is proposed in Chapter 3, extending the work of Nason *et al.* (2000) to two dimensions. The model provides a framework for the analysis of locally stationary processes which fall on a regular grid. As a result, we can obtain a location-scale decomposition of the covariance structure into three directions: horizontal, vertical and diagonal.

A potential application of our modelling approach lies in the field of texture analysis, a discipline which is concerned with the discrimination, classification and segmentation of images whose structure is in some sense regular. The first half of Chapter 4 provides a review of this field, paying particular attention to recently proposed wavelet-based texture analysis methods. The majority of these approaches are motivated by wavelet *properties* rather than being wavelet-based models of structure. The second half of Chapter 4 focuses on the application of our model to various texture analysis problems, contrasting our results with alternative approaches.

Chapter 5 considers the construction of the inner product matrix of discrete autocorrelation wavelets. This matrix plays an important role in the unbiased estimation of location-scale measures of power associated with locally stationary processes, be they time series or random fields which lie on a regular grid. The direct, brute-force construction of this quantity becomes computationally intensive when dealing with large datasets as it requires $O(N \log N)$ operations. Utilising a well-known recursion scheme associated with discrete autocorrelation wavelets, we are able to derive an efficient scheme for the construction of this inner product matrix. The scheme relates neighbouring elements of the matrix which lie on a given diagonal. This results in an $O(\log(N)^3)$ construction.

We conclude this thesis by returning to the time series model proposed by Nason *et al.* (2000). As a consequence of the location-scale decomposition afforded by their model, Nason *et al.* were able to propose a time-localised measure of the autocovariance structure contained within locally stationary time series. Chapter 6 details the results of an initial study which has been made into the application of this quantity. To facilitate our understanding of this measure, we start by interpreting the local autocovariance in terms of statistics which are used in classical stationary time series analysis. We then consider the application of a local autocovariance estimator to the special case of second-order stationary time series.

The estimation theory associated with the work presented in Chapter 3 has been implemented as a suite of S-Plus routines which tie in with the WaveThresh3 package released by Nason (1998). The suite may be downloaded from:

<http://www.stats.bris.ac.uk/~maiae/LS2W>.

Note that many of the figures contained in this thesis were produced using WaveThresh.

Chapter 2

Literature review

2.1 Introduction

The last fifteen years have seen an explosion of interest in wavelets. Although initially embedded in fields such as functional analysis, signal processing and geophysics, wavelets are now involved in many diverse disciplines: from solving partial differential equations to compressing images and analysing economic data — see Hubbard (1998) or Graps (1995) for further examples. Several stimulating books have also appeared on the subject in recent years, see for example Chui (1992), Daubechies (1992), Meyer (1992), Mallat (1999) and Vidakovic (1999).

This chapter summarises the principal aspects of wavelet theory which are required for the work presented in this thesis and reviews various elements of the application of wavelets in statistics. It should be noted that a review of wavelets in the field of texture analysis is postponed until Chapter 4.

The first half of the chapter focuses on the fundamentals of wavelet theory. We start by providing a brief synopsis of Fourier theory, highlighting the difficulty that Fourier series have in representing functions with discontinuities. Then, in Section 2.3 we define a wavelet and consider its connection to multiresolution analysis. Various discrete wavelet transforms are introduced in Sections 2.4 and 2.5 including the discrete wavelet transform, the non-decimated wavelet transform and the wavelet packet transform. The review of wavelet theory is concluded in Section 2.6 where we consider the extension of the wavelet transform to higher dimensions.

In the second part of this chapter we consider the application of wavelet methods

to time series analysis, paying particular attention to recent work on the modelling of locally stationary processes. Finally, in Section 2.8 we review the application of wavelets to one-dimensional smoothing problems.

2.2 The Fourier transform

Prior to introducing wavelets, we review some basic results from Fourier theory. The aim of this review is to motivate the need for efficient series representations for certain function types, for example functions which possess a finite number of discontinuities. Additionally, it is often useful to consider the Fourier-domain properties of wavelets. Thus it is hoped that this section will also act as a useful *aide-memoire*. For additional details on this topic we refer the reader to Priestley (pp. 184–194, 1981) or, for a more detailed treatment, Champeney (1973).

Following Vidakovic (1999), we define the Fourier transform of a function as follows:

Definition 2.1

Let $\langle f, g \rangle = \int f(x)\overline{g(x)} dx$ denote the inner product of two functions f and g . Then the **Fourier transform** of a function $f \in L^1(\mathbb{R})$ is defined by

$$\begin{aligned}\widehat{f}(\omega) = \mathcal{F}[f(x)] &= \langle f(x), e^{i\omega x} \rangle \\ &= \int_{\mathbb{R}} f(x)e^{-i\omega x} dx.\end{aligned}$$

Further, if $\widehat{f} \in L^1(\mathbb{R})$ is the Fourier transform of $f \in L^1(\mathbb{R})$, then

$$f(x) = \mathcal{F}^{-1}[\widehat{f}(w)] = \frac{1}{2\pi} \int_{\mathbb{R}} \widehat{f}(\omega)e^{i\omega x} d\omega$$

is defined to be the **inverse Fourier transform**.

(See Vidakovic (pp. 30–31) for a list of important Fourier transform properties.)

A function, f , periodic on $[0, 2\pi)$ may also be represented as a sum of sinusoids:

$$f(x) = \frac{a_0}{2\pi} + \frac{1}{\pi} \sum_{n=1}^{\infty} a_n \cos(nx) + b_n \sin(nx),$$

where

$$a_n = \int_0^{2\pi} f(x) \cos(nx), \quad b_n = \int_0^{2\pi} f(x) \sin(nx) dx.$$

This representation is commonly known as the *Fourier series* representation of a function. Such a representation is possible as $\{1, \cos(nx), \sin(nx)\}$ constitutes an orthonormal basis of $L^2([0, 2\pi))$.

When representing a function in terms of its Fourier series it is often desirable that the expansion should be economical. In other words, that there are only a few non-zero coefficients. Such sparsity can facilitate our understanding of the function's structure. Unfortunately, for many functions the Fourier series is not sparse. Consider, for example, the Fourier series representation in figure 2.1. Note how artefacts are situated near the points of discontinuity. These are known as *Gibbs effects* and occur because the series coefficients $\{a_n, b_n\}$ are evaluated using values of the function across the whole interval $[0, 2\pi)$. Thus a large number of high frequency terms are required to represent the discontinuity accurately. Recently, researchers have been investigating ways of overcoming such problems. One such approach is called wavelet theory.

2.3 What is a wavelet?

As the term “wavelet” suggests, a wavelet is simply a small localised wave. Today the word is frequently used to describe an orthonormal basis of $L^2(\mathbb{R})$. The wavelet basis is formed by *translating* and *dilating* a basic function ψ , which we call the “mother wavelet”. Following Meyer (1992), a mother wavelet may be defined as follows:

Definition 2.2

Let $m \in \mathbb{N}$. Then for $x \in \mathbb{R}$, a function $\psi(x)$ is called a mother wavelet of order m if the following properties hold:

W1 If $m = 0$, $\psi(x) \in L^\infty(\mathbb{R})$. If $m \geq 1$, then $\psi(x)$ and all its derivatives up to order m belong to $L^\infty(\mathbb{R})$.

W2 $\psi(x)$ and all its derivatives up to order m decrease rapidly as $x \rightarrow \pm\infty$.

W3 For each $k \in \{0, \dots, m\}$,

$$\int_{-\infty}^{\infty} x^k \psi(x) dx = 0.$$

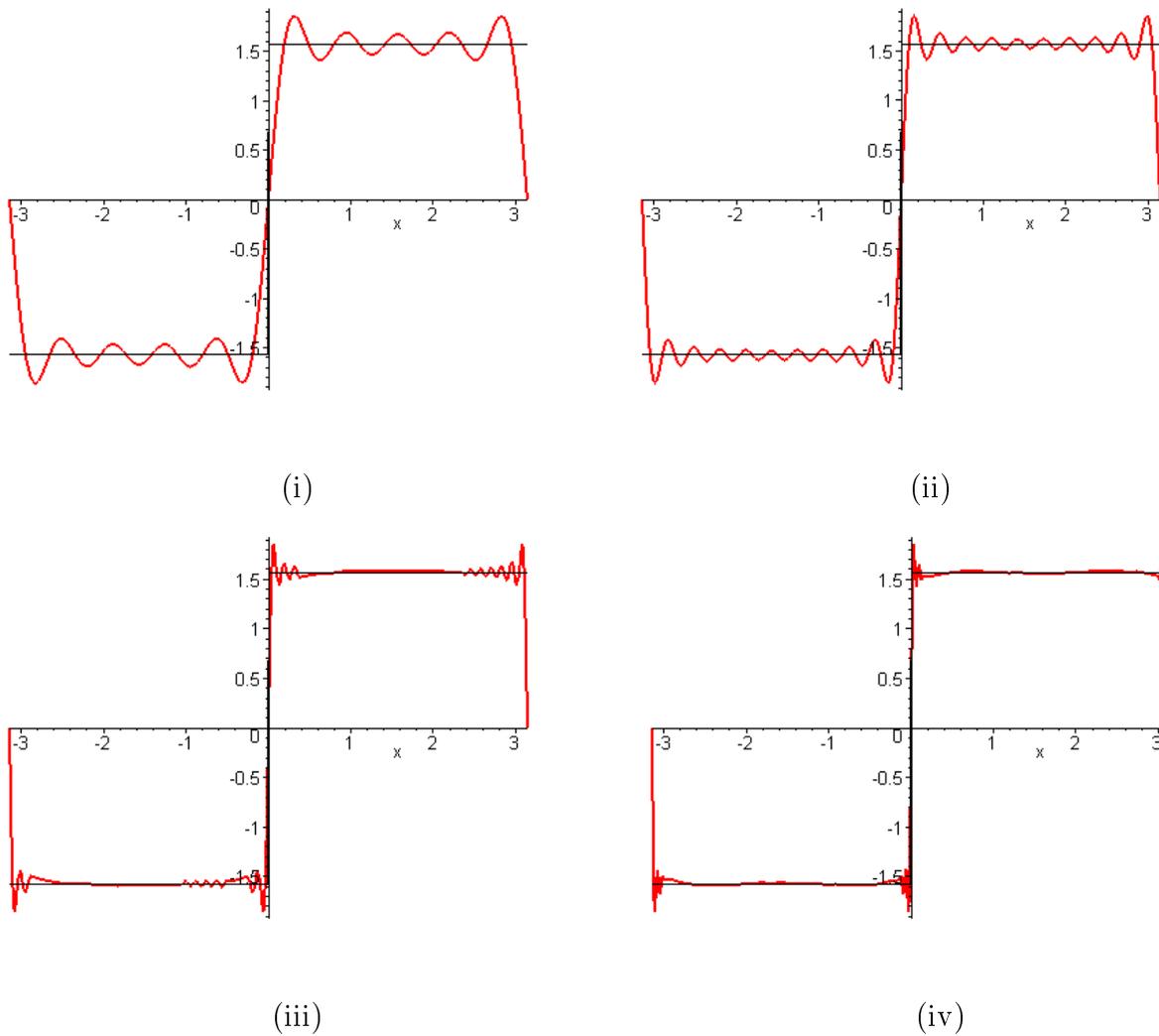


Figure 2.1: Fourier series approximation of the box-car function (black line) using the first (i) 10, (ii) 20, (iii) 50 and (iv) 100 terms of the representation (increasing in frequency).

W4 The collection $\{\psi_{j,k}\}_{j,k \in \mathbb{Z}}$ forms an orthonormal basis of $L^2(\mathbb{R})$, the $\psi_{j,k}$ being constructed from the mother wavelet using the identity

$$\psi_{j,k}(x) = 2^{j/2} \psi(2^j x \Leftrightarrow k). \quad (2.1)$$

Condition W1 expresses the regularity (i.e. smoothness) of the wavelet, whilst conditions W2 and W3 address the localisation and oscillation of ψ . Frequently, W3 is referred to as the *vanishing moments* property. Finally, the parameters j, k in condition W4 are called the dilation and translation parameters respectively.

Whereas the Fourier basis consists of only one choice of basis functions

$$\{1, \cos(nx), \sin(nx)\}$$

there are *many* possible mother wavelets from which we can choose. Possibly the simplest example of a wavelet basis is that proposed by Haar (1910). Haar wavelets are generated by the following mother wavelet, of order 0,

$$\psi(x) = \begin{cases} 1 & \text{if } 0 \leq x < 1/2 \\ \Leftrightarrow -1 & \text{if } 1/2 \leq x < 1 \\ 0 & \text{otherwise.} \end{cases} \quad (2.2)$$

Clearly properties W1 \rightarrow W3 of Definition 2.2 are satisfied by the Haar basis. It remains therefore to show W4, that the $\{\psi_{j,k}\}$ form an orthonormal basis. To demonstrate this, we must appeal to multiresolution analysis (see Section 2.3.1). A detailed examination of this orthonormality is provided by Kovac (Chapter 2, 1998).

Shannon's wavelet, also known as the Littlewood-Paley wavelet, possesses time-frequency properties which are complementary to those of the Haar basis as it is compactly supported in the Fourier domain:

$$\widehat{\psi}(\omega) = \begin{cases} (2\pi)^{-1/2} & \text{for } \pi \leq |\omega| < 2\pi \\ 0 & \text{otherwise.} \end{cases}$$

In the time domain, this results in a wavelet of infinite support:

$$\psi(x) = (\pi x)^{-1} (\sin(2\pi x) \Leftrightarrow \sin(\pi x)).$$

See Daubechies (Section 4.2.1, 1992) for further details.

Meyer (1986) suggests a modification of Shannon's wavelet, smoothing the sharp edges which exist in the Fourier-domain using a "taper" function, whilst preserving the orthonormality of the wavelet. Following Daubechies (Section 4.2.1, 1992), Meyer wavelets are defined in the frequency domain as follows:

$$\hat{\psi}(\omega) = \begin{cases} (2\pi)^{-1/2} e^{i\omega/2} \sin \left[\frac{\pi}{2} v \left(\frac{3}{2\pi} |\omega| \Leftrightarrow 1 \right) \right] & \text{for } 2\pi/3 \leq |\omega| < 4\pi/3, \\ (2\pi)^{-1/2} e^{i\omega/2} \cos \left[\frac{\pi}{2} v \left(\frac{3}{4\pi} |\omega| \Leftrightarrow 1 \right) \right] & \text{for } 4\pi/3 \leq |\omega| < 8\pi/3, \\ 0 & \text{otherwise,} \end{cases}$$

the taper v being chosen such that the following properties are satisfied:

$$\begin{aligned} v(x) + v(1 \Leftrightarrow x) &= 1, \\ v(x) &= 0, x \leq 0, \\ \text{and } v(x) &= 1, x \geq 1. \end{aligned}$$

Other wavelet families include Daubechies' Extremal Phase and Least Asymmetric wavelets (see Section 2.3.3) and Franklin wavelets. A comprehensive review of these and other wavelet families may be found in Vidakovic (Section 3.4, 1999) or Percival & Walden (Chapter 4, 2000). Daubechies (1992) provides a more mathematical treatment.

By analogy with Fourier series, for certain choices of mother wavelet, the $\{\psi_{j,k}\}$ constitute an orthonormal basis of $L^2(\mathbb{R})$. Thus any function $f \in L^2(\mathbb{R})$ may be represented as

$$f(x) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} w_{j,k} \psi_{j,k}(x) \quad (2.3)$$

where $w_{j,k} = \int_{\mathbb{R}} f(t) \psi_{j,k}(t) dt$. The $w_{j,k}$ provide information about the structure on "scale" 2^j near position $2^{-j}k$. In other words, the series is comprised of localised scale information. This is in stark contrast to the Fourier approach, where coefficients are calculated using the whole function on the interval $[0, 2\pi)$. As a consequence of this localised information, wavelets have the ability to provide sparse representations of functions with a finite number of discontinuities.

2.3.1 Multiresolution analysis

We now consider the concept of multiresolution analysis, first introduced by Meyer (1986) and Mallat (1989a). Put simply, multiresolution analysis provides a framework

for examining functions at different scales: zooming in to see the fine detail and moving out to view the broader picture. It enables us to understand wavelet bases and construct new examples. As we shall see in Section 2.4, multiresolution analysis also has an important role in the formation of the discrete wavelet transform. Stimulating reviews of multiresolution analysis are provided by Mallat (1989a, 1999), Daubechies (1992), Vidakovic (1999) and Jawerth & Sweldens (1994).

Following Mallat (1989a) we define a multiresolution analysis as follows:

Definition 2.3

A **multiresolution analysis** (MRA) is a nested sequence of closed subspaces, $V_j \subset L^2(\mathbb{R})$ for $j \in \mathbb{Z}$,

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \quad (2.4)$$

such that

1. the spaces have a trivial intersection:

$$\bigcap_{j \in \mathbb{Z}} V_j = \{0\}; \quad (2.5)$$

2. the union is dense in $L^2(\mathbb{R})$:

$$\overline{\bigcup_{j \in \mathbb{Z}} V_j} = L^2(\mathbb{R}); \quad (2.6)$$

3. the following two-scale relation exists:

$$f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \quad \forall j \in \mathbb{Z}; \quad (2.7)$$

- 4.

$$f(x) \in V_0 \Leftrightarrow f(x \Leftrightarrow k) \in V_0 \quad \forall k \in \mathbb{Z}; \quad (2.8)$$

5. and, finally, there exists a scaling function, $\phi \in V_0$, whose integer translations $\{\phi_{0,k} : k \in \mathbb{Z}\}$ constitute an orthonormal basis of V_0 .

Note that conditions (2.7) and (2.8) imply that $\{\phi_{j,k} : k \in \mathbb{Z}\}$ constitutes an orthonormal basis of V_j . Furthermore since $V_0 \subset V_1$, the function $\phi(x) \in V_0$ may

be represented as a linear combination of functions from V_1 . In other words for some coefficients $\{h_k\}_{k \in \mathbb{Z}}$, we can express

$$\phi(x) = \sum_{k \in \mathbb{Z}} h_k \phi_{1,k}(x) = \sum_{k \in \mathbb{Z}} h_k 2^{1/2} \phi(2x \Leftrightarrow k). \quad (2.9)$$

Equation (2.9) is often called the *scaling equation*. The coefficients $\mathbf{h} \equiv \{h_k\}_{k \in \mathbb{Z}}$ are of fundamental importance in the construction of wavelets and in the development of an efficient recipe for performing the discrete wavelet transform (see Section 2.4 for further details). We will often refer to \mathbf{h} as a filter — a term which arises naturally from the role which this quantity plays in the discrete wavelet transform.

We note two important properties of the wavelet filters, \mathbf{h} , associated with an orthogonal MRA:

$$\sum_{k \in \mathbb{Z}} h_k = \sqrt{2} \quad \text{and} \quad \sum_{k \in \mathbb{Z}} h_k h_{k-2l} = \delta_l.$$

See Vidakovic (pp. 53–55, 1999) for proofs of the above.

Example 2.1 *Possibly the simplest example of a MRA is that which can be constructed using the Haar scaling function:*

$$\phi(x) = \begin{cases} 1 & \text{for } 0 \leq x < 1 \\ 0 & \text{otherwise.} \end{cases} \quad (2.10)$$

It is easily verified that $\{\phi(x \Leftrightarrow k)\}_{k \in \mathbb{Z}}$ forms an orthonormal set. Simply note $\int_{\mathbb{R}} \phi(x) dx = 1$ and that integer translations do not overlap. Furthermore, setting

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

we find that

$$\phi(x) = \sum_{k \in \mathbb{Z}} h_k 2^{1/2} \phi(2x \Leftrightarrow k) = 2^{-1/2} \phi_{1,0}(x) + 2^{-1/2} \phi_{1,1}(x).$$

Thus

$$h_k = \begin{cases} 2^{-1/2} & \text{for } k = 0, 1 \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, $\sum_k h_k^2 = 1$ and $\sum_k h_k = \sqrt{2}$ in this case.

An approximation of a function f at resolution level j is given by

$$f_j(x) = \sum_{k \in \mathbb{Z}} c_{j,k} \phi_{j,k}(x) = \mathcal{P}_j f,$$

where \mathcal{P}_j is the projection operator onto $\{\phi_{j,k}(\cdot)\}_{k \in \mathbb{Z}}$. This is equivalent to saying that \mathcal{P}_j is the projection operator onto V_j , for $\text{span}\{\phi_{j,k}\}_{k \in \mathbb{Z}} = V_j$. Note that as $\{\phi_{j,k}\}_{k \in \mathbb{Z}}$ are orthonormal, the $\{c_{j,k}\}$ may be obtained using

$$c_{j,k} = \langle f, \phi_{j,k} \rangle = \int_{\mathbb{R}} f(x) \phi_{j,k}(x) dx. \quad (2.11)$$

Figure 2.2 displays an example of successive approximations of a test function, considered by Nason & Silverman (1994), at various resolutions using the Haar MRA.

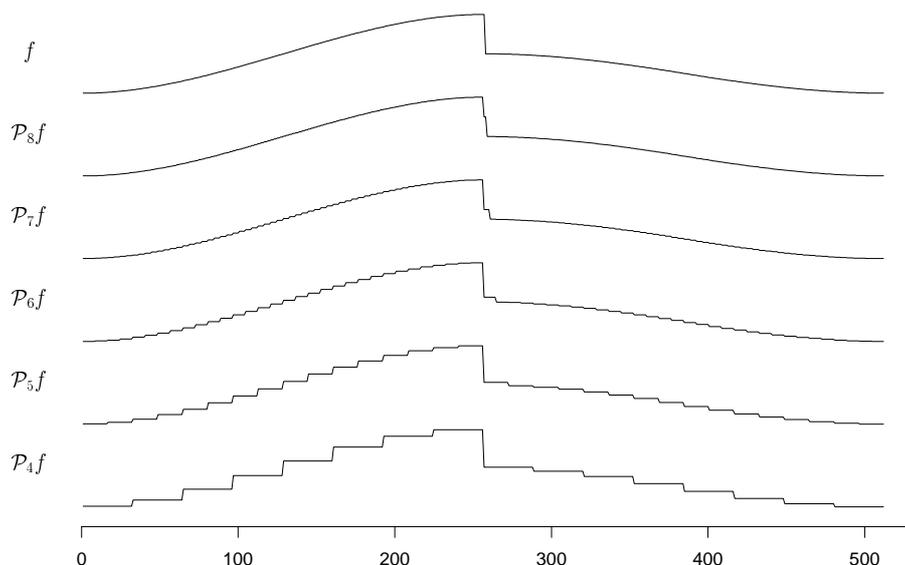


Figure 2.2: Successive approximations of a smooth test function, f , containing a single discontinuity using the Haar MRA. The lowest image denotes a coarse (broad-scale) approximation of the function (upper image), whilst $\mathcal{P}_5 f \rightarrow \mathcal{P}_8 f$ display successively finer scale approximations.

2.3.2 Fourier properties of ϕ

In Section 2.3.3 we consider the derivation of a mother wavelet, ψ , from the scaling function ϕ associated with a MRA. When doing so, it is useful to consider the Fourier

properties of both ϕ and the vector of coefficients, \mathbf{h} , obtained from the scaling equation (2.9). To this end we define

$$m_0(\omega) = 2^{-1/2} \sum_{k \in \mathbb{Z}} h_k e^{-ik\omega}. \quad (2.12)$$

This function describes the behaviour of the filter, \mathbf{h} , in the frequency domain.

It is easily shown that

$$\widehat{\phi}(\omega) = m_0\left(\frac{\omega}{2}\right) \widehat{\phi}\left(\frac{\omega}{2}\right), \quad (2.13)$$

where $\widehat{\phi}(\omega)$ denotes the Fourier transform of ϕ . For a proof of this relation see, for example, Vidakovic (p. 53, 1999). Additionally, Daubechies (p. 132, 1992) demonstrates that the orthonormality of the scaling function leads to the condition that

$$|m_0(\omega)|^2 + |m_0(\omega + \pi)|^2 = 1. \quad (2.14)$$

Combining relation (2.13) with the result that $|\widehat{\phi}(0)| = 1$ (Mallat, 1989a), it follows that

$$|m_0(0)| = 1 \quad \Rightarrow \quad |m_0(\pi)| = 0. \quad (2.15)$$

These identities will prove useful in Chapter 3.

2.3.3 Deriving a wavelet from a MRA

Daubechies (p. 130, 1992) writes that

“The basic tenet of multiresolution analysis is that whenever a collection of closed subspaces satisfies ... [the conditions of a multiresolution analysis] ... then there exists an orthonormal wavelet basis $\{\psi_{j,k} : j, k \in \mathbb{Z}\}$ of $L^2(\mathbb{R})$ ”.

Indeed Daubechies (p. 136, 1992) observed that “every orthonormal wavelet basis of practical interest ... is associated with a multiresolution analysis”. The question therefore is how can a mother wavelet be derived from the scaling function?

The key to the answer lies in thinking about what happens to the detail information which is lost when we move down from one resolution space, V_{j+1} , to a coarser space

V_j . The difference between the two subspaces may be expressed in operator notation by

$$\mathcal{Q}_j = \mathcal{P}_{j+1} \ominus \mathcal{P}_j, \quad \forall j \in \mathbb{Z}.$$

Thus $\mathcal{Q}_j f$ is simply the projection of f on the orthogonal complement of V_j in V_{j+1} . Alternatively, we can define a detail space as follows:

Definition 2.4

The detail space W_j is defined to be the orthogonal complement of V_j in V_{j+1} . Thus,

$$V_{j+1} = V_j \oplus W_j$$

where $W_j \perp V_j$ and $W_j \perp W_{j'}$ if $j \neq j'$.

Hence for $J > j$,

$$V_J = V_j \oplus \bigoplus_{k=j}^{J-1} W_k \quad (2.16)$$

where the subspaces $\{W_k\}_{k=j}^{J-1}$ are mutually orthogonal.

Note that equation (2.16), in tandem with (2.5) and (2.6), ensures that

$$\bigoplus_{j \in \mathbb{Z}} W_j = L^2(\mathbb{R}). \quad (2.17)$$

Thus, $L^2(\mathbb{R})$ may be decomposed into mutually orthogonal subspaces. Additionally, the $\{W_j\}$ inherit the scale-relating property (2.7) of the $\{V_j\}$. In other words,

$$f(x) \in W_j \iff f(2x) \in W_{j+1}. \quad (2.18)$$

Thus if a function, ψ , can be found such that its integer translations form an orthonormal basis of W_0 , then the relations given in (2.17) and (2.18) ensure that

$$\{\psi_{j,k} : \psi_{j,k}(x) = 2^{j/2} \psi(2^j x \ominus k)\}_{j,k \in \mathbb{Z}}$$

form an orthonormal basis of $L^2(\mathbb{R})$.

To derive a wavelet function, ψ , from the scaling function we note that as $W_0 \subset V_1$, $\psi(x) \in V_1$. Hence $\psi(x)$ may be represented as

$$\psi(x) = \sum_{k \in \mathbb{Z}} g_k \sqrt{2} \phi(2x \ominus k) \quad (2.19)$$

for some coefficients $\{g_k\}_{k \in \mathbb{Z}}$. Setting

$$m_1(\omega) = \frac{1}{\sqrt{2}} \sum_k g_k e^{-ik\omega} \quad (2.20)$$

it may be shown that

$$\widehat{\psi}(\omega) = m_1\left(\frac{\omega}{2}\right) \widehat{\phi}\left(\frac{\omega}{2}\right).$$

By considering the Fourier properties of the $\{h_k\}$ and $\{g_k\}$, Mallat (1989a) demonstrates that the decomposition of a function, $f \in V_0$, into orthogonal components V_{-1} and W_{-1} can be achieved only if the following conditions are satisfied:

$$|m_0(\omega)|^2 + |m_1(\omega)|^2 = 1 \quad (2.21)$$

and

$$m_0(\omega) \overline{m_1(\omega)} + m_1(\omega + \pi) \overline{m_0(\omega + \pi)} = 0. \quad (2.22)$$

For further details see Daubechies (Chapter 5, 1992) or Vidakovic (pp. 57–59, 1999).

One possible choice of m_1 which satisfies equations (2.21) and (2.22) is

$$m_1(\omega) = \Leftrightarrow e^{-i\omega} \overline{m_0(\omega + \pi)}.$$

This is equivalent to setting:

$$g_n = (\Leftrightarrow 1)^n h_{1-n}. \quad (2.23)$$

This relation is commonly referred to as the *quadrature mirror filter* relation. The following example highlights how, given ϕ , we can derive ψ using the relation given in equation (2.23).

Example 2.2 Recall from Example 2.1 that for the Haar scaling function

$$h_k = \begin{cases} 2^{-1/2} & \text{for } k = 0, 1, \\ 0 & \text{otherwise.} \end{cases}$$

Using (2.23), it follows that

$$g_k = \begin{cases} 2^{-1/2} & \text{if } k = 0, \\ \Leftrightarrow 2^{-1/2} & \text{if } k = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Hence

$$\begin{aligned}\psi(x) &= \phi(2x) \Leftrightarrow \phi(2x \Leftrightarrow 1) \\ &= \begin{cases} 1 & \text{if } 0 \leq x < 1/2, \\ \Leftrightarrow 1 & \text{if } 1/2 \leq x < 1, \\ 0 & \text{otherwise.} \end{cases}\end{aligned}$$

Daubechies' compactly supported wavelets

Daubechies (1988, 1992) introduced two classes of compactly supported wavelets, having a pre-assigned degree of smoothness. These wavelets are commonly known as Daubechies' *extremal phase* and *least asymmetric* wavelets respectively. As we shall see later, such compactly supported wavelets are particularly useful when dealing with discrete data, for the filters \mathbf{h} and $\mathbf{g} = \{g_k\}$ associated with the scaling relations (2.9) and (2.19) only have a finite number of non-zero coefficients.

The Daubechies' compactly supported wavelets are constructed using the usual condition that $|m_0(\omega)|^2 + |m_0(\omega + \pi)|^2 = 1$. Additionally, we suppose that ψ has $N(\geq 2)$ vanishing moments. Thus, by Theorem 5.5.1 of Daubechies (1992), these wavelets must also satisfy:

$$m_0(\omega) = \left(\frac{1 + e^{-i\omega}}{2} \right)^{N+1} \mathcal{L}(\omega) \quad (2.24)$$

where \mathcal{L} is a 2π -periodic continuous function having continuous derivatives up to order N . Examples of Daubechies' compactly supported wavelets are displayed in figure 2.3. For details of the construction of such wavelets, we refer the reader to the excellent summary provided by Vidakovic (Section 3.4.5, 1998) or for a more technical exposition, Daubechies (Chapters 6–8, 1992).

2.4 The discrete wavelet transform

Frequently, functions or data sets are observed at a finite number of discrete time points. Thus a continuous representation such as that displayed in equation (2.3) is unsuitable. Rather a discrete analogue is required. The *discrete wavelet transform* (DWT), proposed by Mallat (1989a, b), connects wavelets with multiresolution analysis to provide an efficient scheme for performing a discrete, wavelet-based transformation.

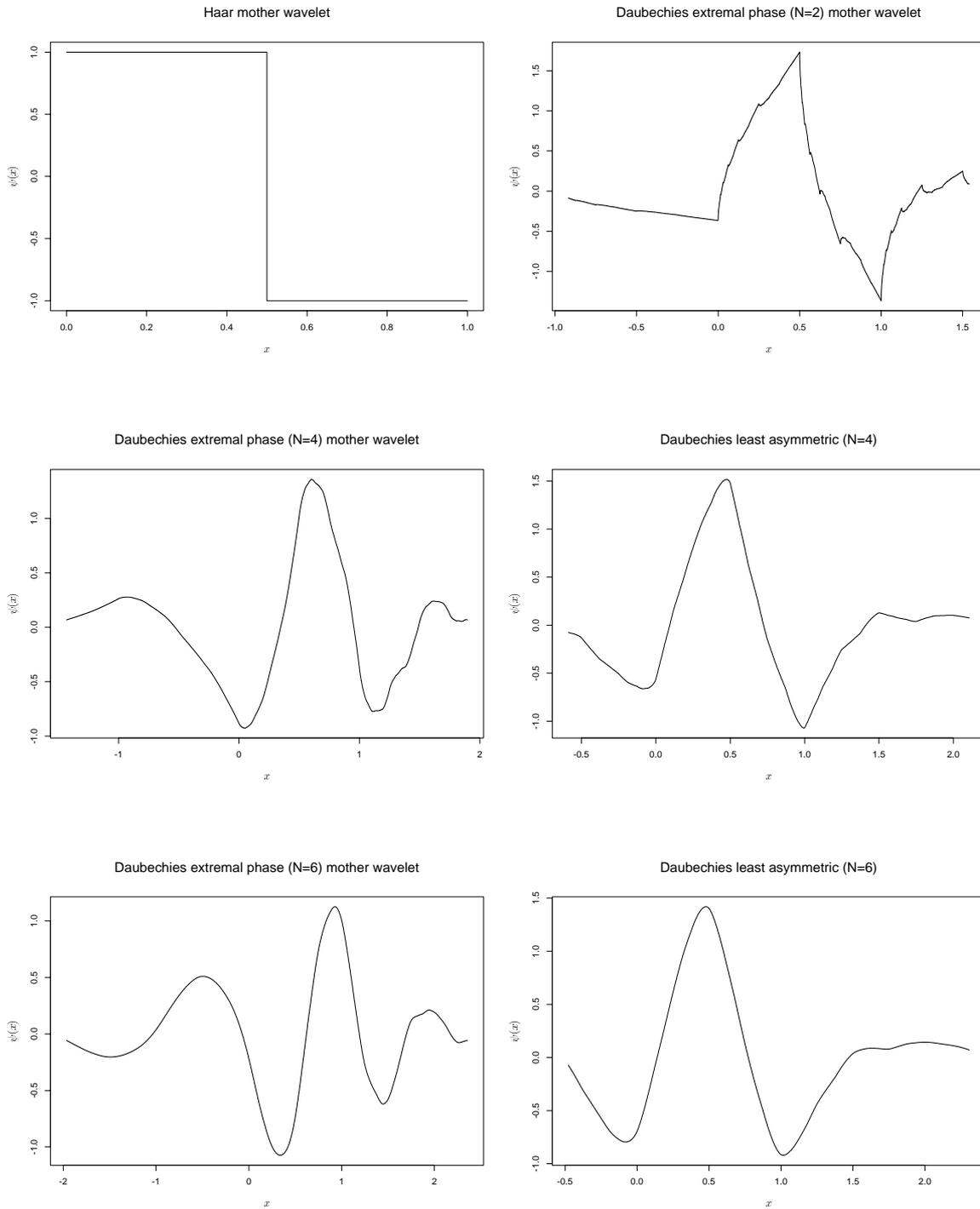


Figure 2.3: Mother wavelets of a selection of Daubechies' compactly supported orthonormal wavelet bases.

The basic idea of the scheme is to filter the data sequence, using the filters \mathbf{h} and \mathbf{g} associated with the scaling equations (2.9) and (2.19), to obtain the wavelet coefficients at different levels. Our description of the DWT follows Vidakovic (Chapter 4, 1999).

Suppose by way of introduction that a function, f , is observed at a dyadic number of equally spaced points:

$$c_{J,k} = f(t_k) \quad \text{for } k = 0, \dots, 2^{J-1} \text{ for some } J \in \mathbb{N},$$

where $t_k = t_0 + k\Delta$ for some $t_0, \Delta \in \mathbb{R}$. A function \bar{f} can then be constructed using $\{\phi_{J,k}(x)\}_{k \in \mathbb{Z}}$ as follows:

$$\bar{f}(x) = \sum_k c_{J,k} \phi_{J,k}(x).$$

Clearly \bar{f} is an element of V_J . It therefore follows that multiresolution analysis may be used to provide a wavelet decomposition of the data. To this end, consider a MRA

$$\dots \subset V_{j-1} \subset V_j \subset V_{j+1} \subset \dots.$$

As $V_j = V_{j-1} \oplus W_{j-1}$, any function $v_j \in V_j$ may be represented uniquely as

$$v_j(x) = v_{j-1}(x) + w_{j-1}(x)$$

where $v_{j-1} \in V_{j-1}$ and $w_{j-1} \in W_{j-1}$. Moreover, a straightforward substitution of indices in equations (2.9) and (2.19) leads to

$$\phi_{j-1,k}(x) = \sum_l h_{l-2k} \phi_{j,k}(x) \quad \text{and} \quad \psi_{j-1,k}(x) = \sum_l g_{l-2k} \phi_{j,k}(x). \quad (2.25)$$

It therefore follows that

$$\begin{aligned} v_j(x) &= \mathcal{P}_j \bar{f} = \mathcal{P}_{j-1} \bar{f} + \mathcal{Q}_{j-1} \bar{f} \\ &= \sum_l c_{j-1,l} \phi_{j-1,l}(x) + \sum_l d_{j-1,l} \psi_{j-1,l}(x) \\ &= v_{j-1}(x) + w_{j-1}(x). \end{aligned}$$

The coefficients $\{c_{j,l}\}$ and $\{d_{j,l}\}$ are commonly known as the smooth and detail coefficients of the transformation.

Using the relations in (2.25), together with the orthogonality of the $w_{j-1}(x)$ and $\phi_{j-1,l}(x)$ for all j and l , we obtain

$$\begin{aligned} c_{j-1,l} &= \langle v_j, \phi_{j-1,l} \rangle \\ &= \langle v_j, \sum_l h_{k-2l} \phi_{j,k} \rangle \\ &= \sum_k h_{k-2l} \langle v_j, \phi_{j,k} \rangle \\ &= \sum_k h_{k-2l} c_{j,k}. \end{aligned}$$

Similarly, $d_{j-1,l} = \sum_k g_{k-2l} c_{j,k}$.

In summary, Mallat's scheme is implemented as follows: Given a function f which is observed at 2^J equally spaced time points $\{t_i\}_{i=0,\dots,2^J-1}$, set $c_{j,k} = f(t_i)$ for $i = 0, \dots, 2^j \Leftrightarrow 1$. The DWT of the sequence is then obtained by recursively using the relations

$$c_{j-1,l} = \sum_k h_{k-2l} c_{j,k} \quad (2.26)$$

and

$$d_{j-1,l} = \sum_k g_{k-2l} c_{j,k} \quad (2.27)$$

to obtain

$$(c_0, \mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{J-2}, \mathbf{d}_{J-1}), \quad (2.28)$$

where $\mathbf{d}_j = \{d_{j,l}\}$. This scheme is commonly referred to as Mallat's *pyramid algorithm*. Figure 2.4 provides a schematic representation of this algorithm, whilst figure 2.5 displays the DWT of the test function displayed in figure 2.2. The structure in figure 2.4 is very sparse, with only a few non-zero coefficients in the region of the discontinuity.

For compactly supported wavelets, such as those proposed by Daubechies (1988, 1992), the summations in equations (2.26) and (2.27) are finite — an attractive feature which leads to fast computation.

Due to the orthogonality of the $\psi_{j,k}$, the DWT algorithm is invertible. Consider the coefficients associated with approximation and detail spaces V_{j-1} and W_{j-1} . The

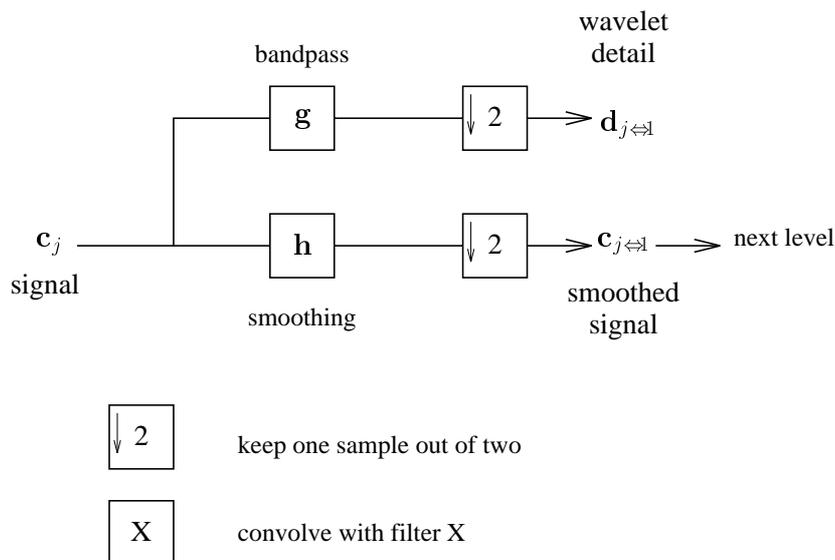


Figure 2.4: DWT decomposition of a discrete signal \mathbf{c}_j into scaling function (smooth) coefficients \mathbf{c}_{j-1} and wavelet (detail) coefficients \mathbf{d}_{j-1} . This figure is reproduced with permission from Nason & Silverman (1994) (after Mallat, 1989b).

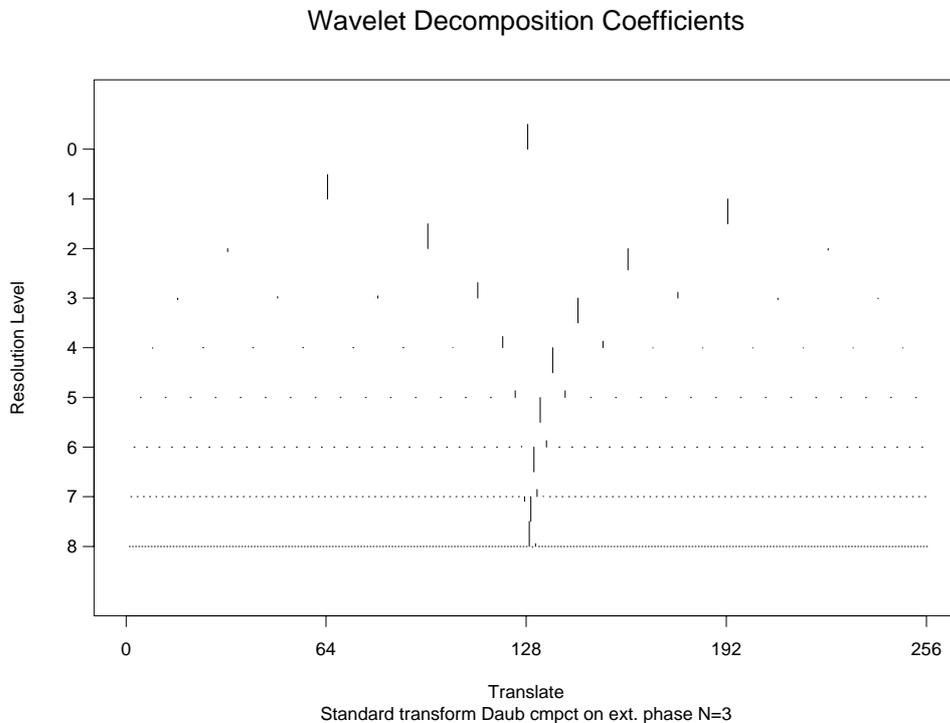


Figure 2.5: DWT of the test function considered in figure 2.2 using Daubechies' Extremal Phase ($N=3$) wavelets.

coefficients $\{c_{j,k}\}$ associated with V_j may be obtained as follows:

$$\begin{aligned} c_{j,k} &= \langle v_j, \phi_{j,k} \rangle \\ &= \sum_l c_{j-1,l} \langle \phi_{j-1,l}, \phi_{j,k} \rangle + \sum_l d_{j-1,l} \langle \psi_{j-1,l}, \phi_{j,k} \rangle \\ &= \sum_l c_{j-1,l} h_{k-2l} + \sum_l d_{j-1,l} g_{k-2l}, \end{aligned}$$

for $\langle \phi_{j-1,l}, \phi_{j,k} \rangle = h_{k-2l}$ and $\langle \psi_{j-1,l}, \phi_{j,k} \rangle = g_{k-2l}$. Hence when performing the DWT, we need only store the detail coefficients $\{\mathbf{d}_j\}$ and the coarsest scale smooth, c_0 . See figure 2.6 for a schematic representation of this reconstruction algorithm.

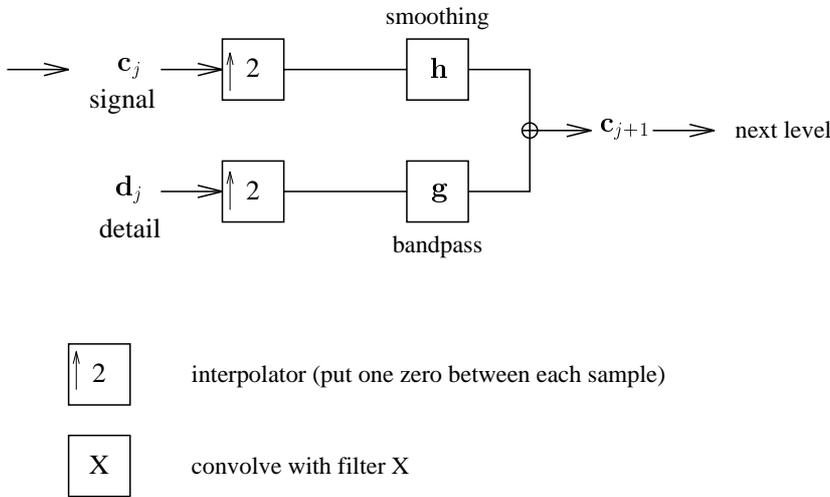


Figure 2.6: DWT reconstruction of a signal \mathbf{c}_{j+1} from the scaling function (smooth) coefficients, \mathbf{c}_j , and wavelet (detail) coefficients, \mathbf{d}_j . This figure is reproduced with permission from Nason & Silverman (1994) (after, Mallat 1989b).

Aside: In many situations, for example when using Daubechies' compactly supported wavelets, the convolutions in (2.26) and (2.27) are such that the filter extends beyond the range of the data. In other words, a boundary problem occurs. Several approaches exist for dealing with this issue. One is to assume that the behaviour at the boundary is symmetric; i.e. $(y_0, \dots, y_{n-1} | y_{n-2}, y_{n-3}, \dots)$. Alternatively, one can assume that the boundary is periodic: $(y_0, \dots, y_{n-1} | y_0, y_1, \dots)$. A third option is to pad out the vector using a constant, for example $(y_0, \dots, y_{n-1} | 0, 0, \dots)$. See Nason & Silverman (1994) for a detailed account of the periodic and symmetric boundary schemes.

An alternative approach to considering the DWT is in terms of operators. Following Nason & Silverman (1995), let \mathcal{H} and \mathcal{G} represent convolutions with the filters \mathbf{h} and \mathbf{g} respectively. Thus

$$\begin{aligned} (\mathcal{H}\mathbf{s})_k &= \sum_n h_{n-k} s_n \\ \text{whilst } (\mathcal{G}\mathbf{s})_k &= \sum_n g_{n-k} s_n. \end{aligned}$$

Further, let \mathcal{D}_0 denote a binary decimation operator which chooses every even element of a sequence:

$$(\mathcal{D}_0\mathbf{s})_j = s_{2j} \quad \forall j \in \mathbb{Z}.$$

Then the DWT of a sequence of 2^J equally spaced observations, $\{c_{J,k}\}_{k=0,\dots,2^J-1}$ may be represented as follows for $j = J \Leftrightarrow 1, \dots, 1$:

$$\begin{aligned} \mathbf{d}_{j-1} &= \mathcal{D}_0\mathcal{G}\mathbf{c}_j, \\ \mathbf{c}_{j-1} &= \mathcal{D}_0\mathcal{H}\mathbf{c}_j. \end{aligned}$$

Note that both \mathbf{c}_j and \mathbf{d}_j are sequences of length 2^j .

2.5 The non-decimated wavelet transform

Vidakovic (p. 145, 1998) provides a brief history of the stationary, or *non-decimated wavelet transform* (NDWT) as it is more commonly referred to. Our description of this transformation follows that of Nason & Silverman (1995) who consider its implementation in a statistical context.

Recall from Section 2.4 that the DWT consists of a convolution followed by dyadic decimation. Although an orthogonal transformation, one undesirable consequence of the decimation step is that a simple integer shift in the coefficient sequence leads to a non-trivial change in the wavelet transform (see figure 2.7 for example). In other words, the DWT of a shifted data set is *not* a shift of the transform of the original data.

The essence of the NDWT is “to ‘fill in the gaps’ caused by the decimation step in the standard wavelet transform” Nason & Silverman (1995). This leads to a *translation equivariant* (TE) representation of the data: a shift in the data manifests itself as a

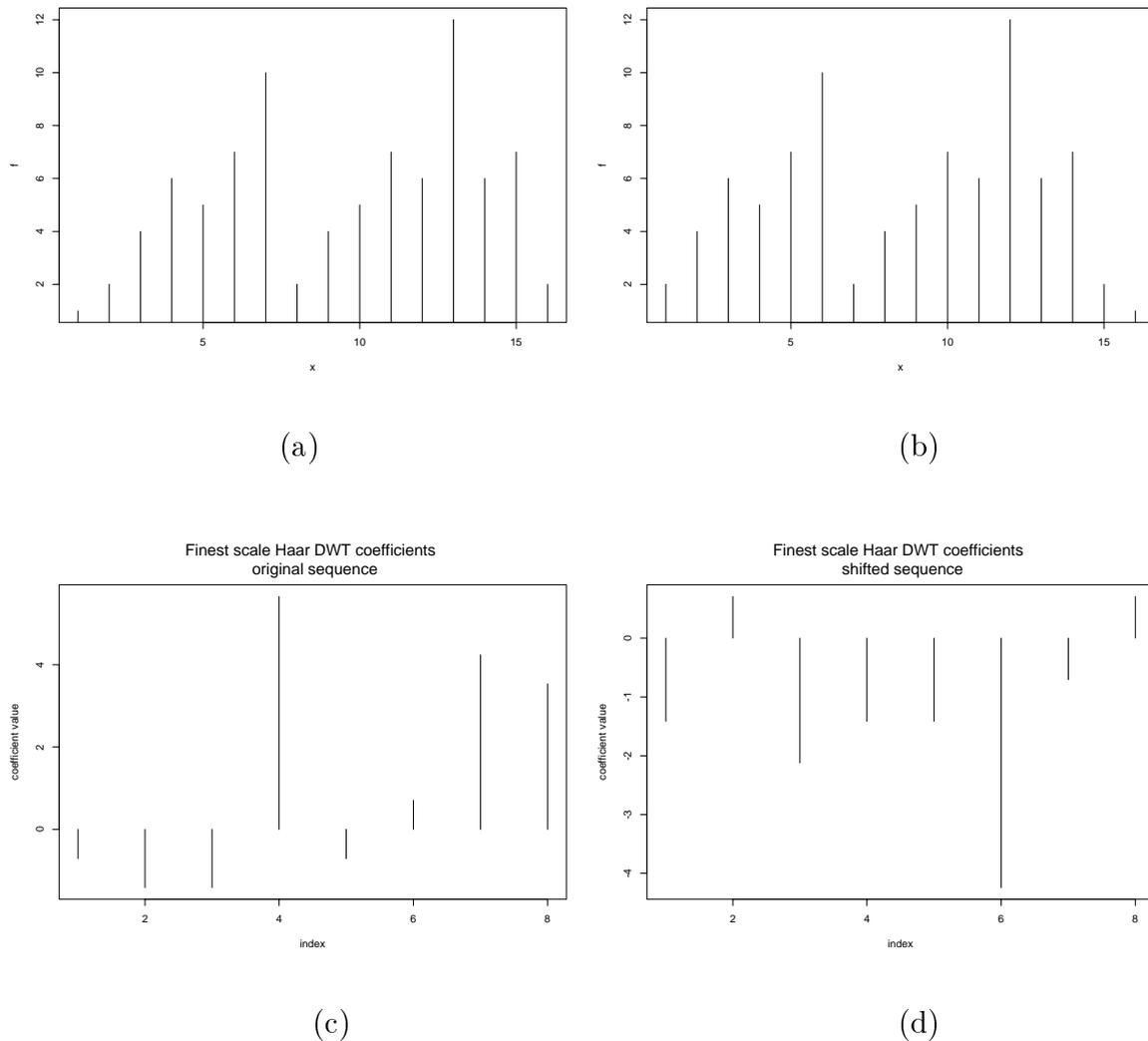


Figure 2.7: Example of the DWT's lack of translation-equivariance. Figure (a) depicts a sequence of sixteen observations whilst (b) depicts the sequence rotated by a simple unit shift. Figures (c) and (d) denote the finest scale detail coefficients, \mathbf{d}_{J-1} , of the Haar DWT for the original and shifted sequence respectively. Note how the finest scale detail coefficients associated with the shifted sequence do not correspond to a simple shift of the detail coefficients associated with the original sequence.

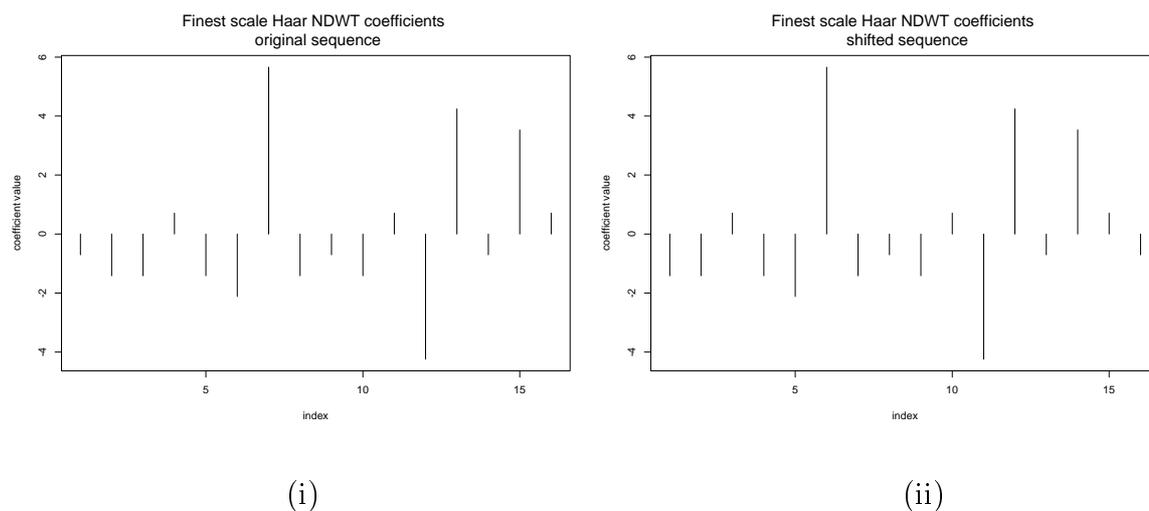


Figure 2.8: Example of the translation equivariance of the non-decimated wavelet transform. Figure (i) represents the finest scale non-decimated Haar wavelet coefficients, d_{J-1} , of the sequence displayed in figure 2.7(a) whilst (ii) represents the finest scale coefficients, d_{J-1} , of the shifted sequence in figure 2.7(b). Note how the coefficients in (ii) are a unit shift of the coefficients displayed in (i).

shift in the (non-decimated) wavelet coefficients (see figure 2.8). An additional benefit of the NDWT is that it provides equal numbers of equi-spaced wavelets at each scale, thus providing more information than the DWT at medium and low resolution levels. However, the price we pay for these features is that the NDWT provides a *redundant*, non-orthogonal representation of the original data.

Following Nason & Silverman (1995), the non-decimated wavelet transform may be implemented as follows:

Suppose we have a sequence $\tilde{\mathbf{c}}_J = \{c_0, \dots, c_{N-1}\}$ where $N = 2^J$ for some $J \in \mathbb{N}$. Then apply the filters \mathcal{H} and \mathcal{G} to \mathbf{c}^J , but *do not* decimate the resulting coefficients. Thus we obtain the vectors

$$\tilde{\mathbf{c}}_{J-1} = \mathcal{H}\tilde{\mathbf{c}}_J \quad \text{and} \quad \tilde{\mathbf{d}}_{J-1} = \mathcal{G}\tilde{\mathbf{c}}_J, \quad (2.29)$$

each of length N . $\tilde{\mathbf{c}}_{J-1}$ represents the finest scale non-decimated smooth whilst $\tilde{\mathbf{d}}_{J-1}$ represents the fine scale detail.

Next let \mathcal{Z} denote the operator which pads out a sequence with zeros as follows:

$$(\mathcal{Z}x)_{2j} = x_j \quad \text{and} \quad (\mathcal{Z}x)_{2j+1} = 0.$$

In other words, \mathcal{Z} inserts a zero between each element of a sequence $\{x\}$. Defining the filters $\mathcal{H}^{[r]}$ and $\mathcal{G}^{[r]}$ to have weights $\mathcal{Z}^r \mathbf{h}$ and $\mathcal{Z}^r \mathbf{g}$ respectively, set

$$\tilde{\mathbf{c}}_{j-1} = \mathcal{H}^{[J-j]} \tilde{\mathbf{c}}_j \quad \text{and} \quad \tilde{\mathbf{d}}_{j-1} = \mathcal{G}^{[J-j]} \tilde{\mathbf{c}}_j,$$

for $j = J \Leftrightarrow 1, \dots, 1$.

The collection of vectors $\{\tilde{\mathbf{c}}_0, \tilde{\mathbf{d}}_1, \tilde{\mathbf{d}}_2, \dots, \tilde{\mathbf{d}}_{J-1}\}$ represents the non-decimated wavelet transform of \mathbf{c}_J .

It is perhaps instructive to relate the NDWT to the DWT. Recall that within each level of the DWT a dyadic decimation operation occurs, retaining only *even* elements of a sequence. Equivalently, an operator which retains *odd* elements of a sequence could be used. Thus, at each step of the DWT we are faced with two equally viable decimation options. This gives rise to the idea of an ϵ -decimated discrete wavelet transform:

Let ϵ be an integer whose binary representation is given by

$$\epsilon = \epsilon_0 \epsilon_1 \dots \epsilon_{J-1}.$$

Then for each level j of a discrete wavelet transformation, let ϵ_j denote whether to retain odd or evenly indexed elements of a sequence. See Nason & Silverman (1995) or Vidakovic (Section 5.5.1, 1999) for further details of this transform.

Nason & Silverman (1995) observe that the NDWT contains the coefficients of the ϵ -decimated DWT for *each* value of ϵ .

2.5.1 Other wavelet transforms

Wavelet packets In the DWT proposed by Mallat (1989b), the transform proceeds from one level to the next by decomposing the smooth sequences, \mathbf{c}_j . The DWT therefore provides a progressive analysis of the low-frequency smooths. However, the most significant information contained within a signal is frequently contained within the middle or high frequencies. Thus, an alternative decomposition which provides a suitably refined partition of these frequency bands is desirable.

Such a decomposition is afforded by the *wavelet packet transform* introduced by Coifman, Meyer & Wickerhauser (1992). This transform is implemented by not only decomposing the smooth sequences, \mathbf{c}_j , but also the detail sequences, \mathbf{d}_j . This provides a multitude of wavelet packets. If all such packets are included, the transformation is redundant. However various approaches can be used to select an orthogonal multiscale representation based upon wavelet packets. The simplest of these is the “vertical line rule” — see Vidakovic (Section 5.3.2, 1999) for details. A more refined approach is afforded by the “Best Basis” algorithm proposed by Coifman & Wickerhauser (1992).

The lifting scheme Sweldens (1996) introduces the *lifting scheme* — an alternative approach to constructing wavelets which exploits the connection between the filters \mathbf{h} and \mathbf{g} . The approach also permits the construction of second-generation, *biorthogonal wavelets* (see Sweldens (1997)). Perhaps one of the most interesting features of the scheme is that it can be used to provide a multiscale decomposition of irregularly spaced data and general meshes. See Daubechies *et al.* (1999) for further details.

2.6 Multidimensional wavelet transforms

We conclude our review of wavelet theory by providing a brief introduction to multidimensional wavelet transforms. Our discussion follows that of Daubechies (Chapter 10, 1992), focusing on orthogonal wavelet representations in two-dimensions — the situation in higher dimensions being analogous. We restrict our discussion to the *separable* multiresolution approximations of $L^2(\mathbb{R}^2)$ considered by Daubechies (1988) and Mallat (1989b). However, it is important to note that other (non-separable) approaches to the construction of two-dimensional wavelets exist. See, for example Meyer (1992) or Daubechies (1992).

Note that conventionally, two-dimensional scaling functions and wavelets are denoted by $\Phi(x, y)$ and $\Psi(x, y)$. We break with this, representing such functions by $\phi(x, y)$ and $\psi(x, y)$ respectively, reserving Φ and Ψ to represent autocorrelation scaling functions and wavelets (see Section 3.5).

2.6.1 Two-dimensional multiresolution analysis

We begin by considering the tensor product of two, one-dimensional multiresolution analyses. Thus, define the spaces \mathbf{V}_j , for $j \in \mathbb{Z}$, by

$$\mathbf{V}_0 = V_0 \otimes V_0 = \overline{\text{span}\{F(x, y) = f(x)g(y) : f, g \in V_0\}}$$

and

$$F(2^j x, 2^j y) \in \mathbf{V}_j \iff F(x, y) \in \mathbf{V}_0.$$

Then the \mathbf{V}_j form a multiresolution ladder in $L^2(\mathbb{R}^2)$ such that

$$\cdots \subset \mathbf{V}_{-2} \subset \mathbf{V}_{-1} \subset \mathbf{V}_0 \subset \mathbf{V}_1 \subset \mathbf{V}_2 \subset \cdots$$

and

$$\bigcap_{j \in \mathbb{Z}} \mathbf{V}_j = \{\mathbf{0}\} \quad \text{and} \quad \overline{\bigcup_{j \in \mathbb{Z}} \mathbf{V}_j} = L^2(\mathbb{R}^2).$$

Further, assume that there exists a scaling function $\phi \in V_0$ such that its integer translations span V_0 and the set $\{\phi(\cdot \Leftrightarrow k)\}_{k \in \mathbb{Z}}$ forms an orthonormal basis of V_0 . Then the collection of

$$\phi_{n_x, n_y}(x, y) = \phi(x \Leftrightarrow n_x) \phi(y \Leftrightarrow n_y) \quad \text{for } n_x, n_y \in \mathbb{Z}$$

forms an orthonormal basis of \mathbf{V}_0 . Setting

$$\begin{aligned} \phi_{j, n_x, n_y}(x, y) &= \phi_{j, n_x}(x) \phi_{j, n_y}(y) \\ &= 2^j \phi(2^j x \Leftrightarrow n_x, 2^j y \Leftrightarrow n_y) \quad \text{for } n_x, n_y \in \mathbb{Z} \end{aligned}$$

it follows that $\{\phi_{j, n_x, n_y}(x, y)\}_{n_x, n_y \in \mathbb{Z}}$ constitutes an orthonormal basis of \mathbf{V}_j .

Recalling the multiresolution analysis construction in one dimension, for each $j \in \mathbb{Z}$ let \mathbf{W}_j be the orthogonal complement of \mathbf{V}_j in \mathbf{V}_{j+1} . Thus,

$$\begin{aligned} \mathbf{V}_{j+1} &= V_{j+1} \otimes V_{j+1} \\ &= (V_j \oplus W_j) \otimes (V_j \oplus W_j) \\ &= V_j \otimes V_j \oplus \{(W_j \otimes V_j) \oplus (V_j \otimes W_j) \oplus (W_j \otimes W_j)\} \\ &= \mathbf{V}_j \oplus \mathbf{W}_j. \end{aligned}$$

The complement space, \mathbf{W}_j , consists of three parts, the orthonormal bases of these portions being given by:

$$\begin{aligned} & \psi_{j,n_x}(x)\phi_{j,n_y}(y) \quad \text{for } W_j \otimes V_j, \\ & \phi_{j,n_x}(x)\psi_{j,n_y}(y) \quad \text{for } V_j \otimes W_j \\ \text{and } & \psi_{j,n_x}(x)\psi_{j,n_y}(y) \quad \text{for } W_j \otimes W_j. \end{aligned}$$

Consequently, we are led to define three wavelets for this separable construction:

$$\left. \begin{aligned} \psi^h(x, y) &= \phi(x)\psi(y) \\ \psi^v(x, y) &= \psi(x)\phi(y) \\ \psi^d(x, y) &= \psi(x)\psi(y). \end{aligned} \right\} \quad (2.30)$$

As $\{\psi_{j,\mathbf{n}}^l(\mathbf{x}) : \mathbf{n} \in \mathbb{Z}^2, l = h, v, \text{ or } d\}$ constitutes an orthonormal basis of \mathbf{W}_j for all $j \in \mathbb{Z}$, it therefore follows that

$$\{\psi_{j,\mathbf{n}}^l(\mathbf{x}) : j \in \mathbb{Z}, \mathbf{n} \in \mathbb{Z}^2 \text{ and } l = h, v, \text{ or } d\}$$

forms an orthonormal basis of $\overline{\bigoplus_{j \in \mathbb{Z}} \mathbf{W}_j} = L^2(\mathbb{R}^2)$.

Note that the superscripts h , v and d in equation (2.30) denote “horizontal”, “vertical” and “diagonal” respectively. The motivation behind this labelling convention is that given an image, each wavelet tends to extract the associated features from a specific direction. For an instructive example, see figure 10.3 of Daubechies (1992). Further details of separable wavelet bases are provided by Mallat (Section 7.7, 1999) and Vidakovic (Section 5.7, 1999).

2.6.2 Discrete wavelet transforms in two dimensions

Mallat (1989b) proposes a two-dimensional analogue of the DWT founded upon the separable wavelet construction detailed above. Suppose we have a square image, C^J , of size $2^J \times 2^J$ for some $J \in \mathbb{N}$. Then the formulae for calculating the smooth and

detail coefficients for $j = J \Leftrightarrow 1, \dots, 1$ are given by

$$\begin{aligned} C_{u,v}^{j-1} &= \sum_m \sum_n h_{m-2u} h_{n-2v} C_{m,n}^j, \\ D_{u,v}^{h,j-1} &= \sum_m \sum_n g_{m-2u} h_{n-2v} C_{m,n}^j, \\ D_{u,v}^{v,j-1} &= \sum_m \sum_n h_{m-2u} g_{n-2v} C_{m,n}^j \\ \text{and } D_{u,v}^{d,j-1} &= \sum_m \sum_n g_{m-2u} g_{n-2v} C_{m,n}^j. \end{aligned}$$

From an implementational perspective, the algorithm is structured as follows (see figure 2.9 for a graphical representation):

1. The rows of C^J are convolved with the one dimensional filters \mathbf{h} and \mathbf{g} , retaining alternate *columns*. This results in two matrices, each of size $2^J \times 2^{J-1}$.
2. The columns of the resulting matrices are then convolved with \mathbf{h} and \mathbf{g} , retaining alternate *rows*. This results in four matrices, each of size $2^{J-1} \times 2^{J-1}$. These represent the level $J \Leftrightarrow 1$ smooth, horizontal, vertical and diagonal details.

The above is repeated for each stage of the decomposition.

As the DWT is orthogonal, it is possible to invert the above algorithm (see figure 2.10). The algorithm is similar to that considered in the one dimensional case. See Mallat (1989b) or Nason & Silverman (1994) for further details.

The NDWT has a similar extension to two dimensions. Suppose that our original data, C^J , is once more a matrix of size $2^J \times 2^J$ for some $J \in \mathbb{N}$ and construct the filters $\mathcal{H}^{[r]}$ and $\mathcal{G}^{[r]}$ as in Section 2.5. Assuming that the NDWT has reached level j , the level $j \Leftrightarrow 1$ coefficients are obtained by firstly convolving the *rows* of C^j with the zero-padded filters, $\mathcal{H}^{[J-j]}$ and $\mathcal{G}^{[J-j]}$, retaining *all* columns. The columns of the two resulting $2^J \times 2^J$ matrices are subsequently convolved with $\mathcal{H}^{[J-j]}$ and $\mathcal{G}^{[J-j]}$, again retaining all coefficients. The four resulting matrices represent the level $j \Leftrightarrow 1$ smooth and horizontal, vertical and diagonal details.

Figures 2.11 and 2.12, below, display the finest scale detail coefficients of a Haar DWT and NDWT of the Lennon image respectively. Both transforms are able to identify regions of change which exist within the image, for example around the glasses and nose, and can extract features which lie in certain directions, such as the horizontal

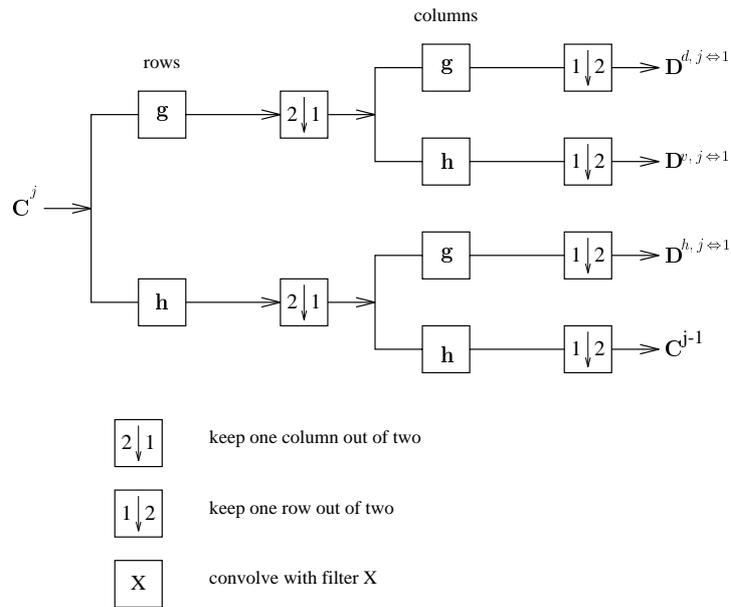


Figure 2.9: Schematic representation of the two dimensional DWT algorithm (after Mallat (1989b)). Figure reproduced with permission from Nason & Silverman (1994).

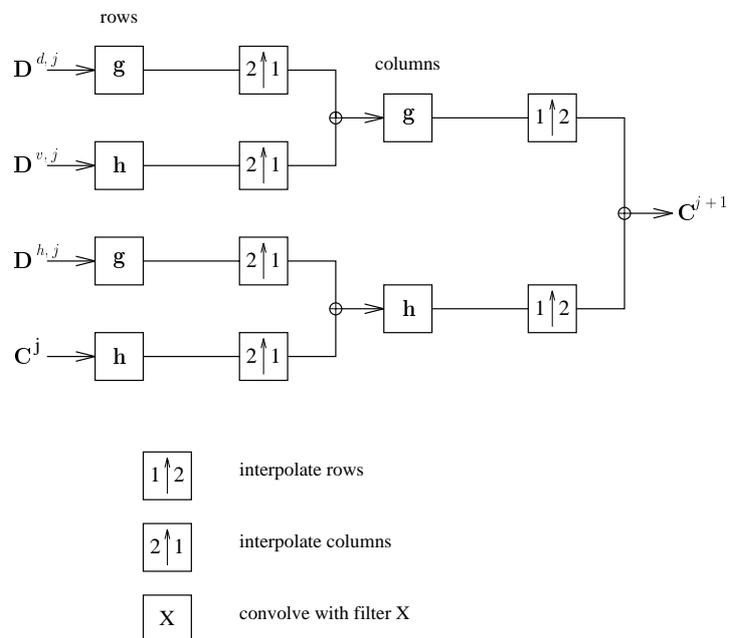


Figure 2.10: Schematic representation of the inverse 2D DWT (after Mallat (1989b)). Figure reproduced with permission from Nason & Silverman (1994).

structure in the region of the mouth. Note that the size of the finest scale detail images returned by the DWT are half those of the original image, whereas the finest scale NDWT detail images are precisely the same size as the original.

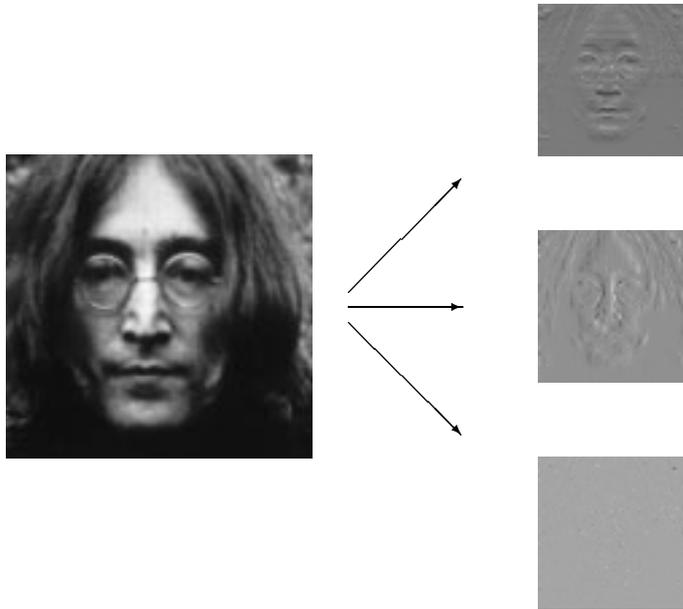


Figure 2.11: Finest scale wavelet coefficients of a Haar DWT of the Lennon image. The upper right hand image depicts the vertical coefficients, the middle image depicts the horizontal coefficients whilst the lowest image depicts the finest scale coefficients in the diagonal direction.

Aside The two-dimensional NDWT is implemented in `WaveThresh` using the command `imwd(..., type="station")`. Herrick (2000) notes this function returns the horizontal and vertical coefficients the wrong way around. An additional bug found with this code is that the implementation of the NDWT does not pad out the filters with zeros. Rather, it simply applies the filters \mathbf{h} and \mathbf{g} recursively. Corrected code for this function is available at

<http://www.stats.bris.ac.uk/~maiae/LS2W>.

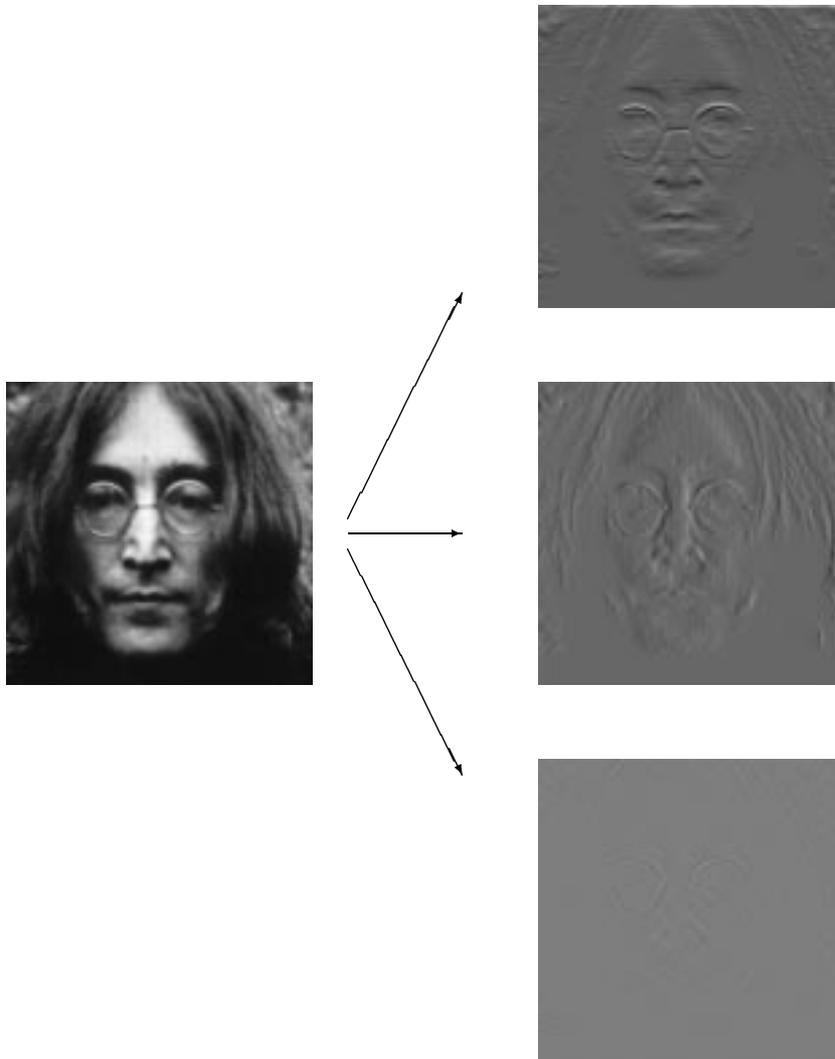


Figure 2.12: Finest scale wavelet coefficients of a Haar NDWT of the Lennon image. The upper right hand image depicts the vertical coefficients, the middle image depicts the horizontal coefficients whilst the lowest image depicts the finest scale coefficients in the diagonal direction.

2.7 Wavelets in time series analysis

2.7.1 Introduction

The sparse, location-scale decomposition of structure and ability to represent signals with sudden changes has made wavelets an appealing tool for statisticians and the focus of much recent research. An early example of the use of wavelets in statistics appears in a paper by Donoho & Johnstone (1994), who consider the application of wavelets to the problem of recovering a signal in the presence of noise. More recently, wavelet methods have been applied to:

1. Density estimation (Hall & Patil, 1995; Donoho *et al.*, 1996; Penev & Dechevsky, 1997; Vannucci & Vidakovic, 1997; Herrick, Nason & Silverman, 2002);
2. Change point problems (Ogden & Parzen, 1996a,b);
3. Hypothesis testing (Fan, 1996);
4. Hazard rate estimation (Antoniadis, Grégoire & Nason, 1999);
5. Spectral estimation of stationary processes (von Sachs & Schneider, 1996; Gao, 1997; Walden, Percival & McCoy, 1998);
6. Estimation of evolutionary spectra (von Sachs & Schneider, 1996; Neumann & von Sachs, 1997);

Stimulating reviews of the application of wavelet methods in statistics are provided by Antoniadis (1997), Morettin (1997), Vidakovic (1999) and Abramovich, Bailey & Sapatinas (2000).

The application of wavelets in time series analysis has also seen a surge in interest over the past few years, as can be seen in the review article of Nason & von Sachs (1999) and the recent monograph of Percival & Walden (2000). For example, Chiann & Morettin (1999) consider the application of wavelets to the spectral analysis of *stationary* time series, developing a wavelet periodogram based on orthonormal wavelets. The results presented in their paper provide an alternative representation to a frequency analysis of a time series. However, as Vidakovic (1999) observes, “the

best one can do for stationary processes is already contained in the classical Cramér spectral representation”.

The *wavelet variance*, which has been extensively studied by Percival, Walden and collaborators, provides an alternative measure of the structure contained within a time series (see, for example, Percival & Walden (Chapter 8, 2000) and references therein). It is a scale-based measure, formed by calculating the variance of the wavelet coefficients of a time series at a given scale. For stationary time series, this measure permits a scale-based decomposition of the process variance, related to the classical spectral decomposition using the Fourier-based spectral density (see Percival & Walden (p. 296, 2000) for further details).

More recently, Nason and collaborators have been investigating the potential of using wavelet *packet* transforms in time series analysis. See for example Nason & Sapatinas (2001), Nason, Sapatinas & Sawczenko (2001) and Hunt & Nason (2002). The problem which they consider is the modelling of a response time series, $\{Y_t\}$, in terms of an explanatory time series, $\{X_t\}$. Having transformed the explanatory series using a wavelet packet transform, standard statistical modelling techniques such as CART, multiple regression or GLMs are then applied to establish which packets are useful for modelling the $\{Y_t\}$. This approach is not founded upon a formal stochastic model.

In this section, we focus on the *locally stationary* time series modelling approach proposed by Nason *et al.* (2000). Their paper builds upon the work of Nason & Silverman (1995) who suggested the use of wavelets as an exploratory technique for producing local spectral density estimates of time series data.

2.7.2 Locally stationary time series models

Nason *et al.* (2000) consider the statistical analysis of locally stationary time series consisting of N ($= 2^J$) data points. Broadly speaking, the term locally stationary means that on close-range inspection, such a series would appear to be stationary (see Nason & von Sachs 1999). Thus if we can collect sufficient information in the region of local stationarity, a sensible estimate of its statistical properties may be obtained.

A brief review of the literature on modelling locally stationary processes is provided by Nason & von Sachs (Section 3, 1999). One approach, proposed by Priestley (1965,

1981) is to generalise the classical Cramér representation of a stationary stochastic process $\{X_t\}_{t \in \mathbb{Z}}$,

$$X_t = \int_{-\pi}^{\pi} A(\omega) e^{i\omega t} d\xi(\omega)$$

where $d\xi(\omega)$ is an orthonormal increments process, to allow for a time varying amplitude $A(\omega, t)$ which changes slowly in a neighbourhood of t . Dahlhaus (1997) considers an interesting alternative, again using the set of harmonics $\{e^{i\omega t} : \omega \in [\ominus\pi, \pi]\}$. The key to Dahlhaus' revolutionary approach lies in his reformulation of the problem of estimating the statistical properties of the time series. Instead of supposing that $\{A(\omega, t)\}$ is observed on a grid $t \in \{1, \dots, T\}$, Dahlhaus re-scales the problem with respect to the length of the time series, T . Thus as T increases, we observe A on an increasingly finer grid. This permits the collection of increasing amounts of information about the local structure of $\{A(\omega, t/T)\}$ as T increases. Hence, asymptotically, the model structure may be identified.

The approach proposed by Nason *et al.* (2000) replaces the harmonics $\{e^{i\omega t}\}$ by compactly supported wavelets — thus decomposing the structure contained within a time series by location and scale. Prior to defining the model proposed by Nason *et al.*, we review the basic building blocks of their *locally stationary wavelet processes*: the discrete wavelet.

2.7.3 Discrete wavelets and their autocorrelations

Discrete wavelets are founded upon the low and high pass quadrature mirror filters \mathbf{h} and \mathbf{g} used in the construction of Daubechies' compactly supported wavelets (Daubechies 1988, 1992). Following Nason *et al.* (2000) we define such wavelets as follows:

Definition 2.5

Let $\{h_k\}_{k \in \mathbb{Z}}$ and $\{g_k\}_{k \in \mathbb{Z}}$ be two filters related by the quadrature mirror filter relation of equation (2.23). Furthermore, for $j \in \mathbb{N}$, define the **discrete wavelet length** as

$$L_j = (2^j \ominus 1)(N_h \ominus 1) + 1,$$

where N_h is the number of non-zero elements contained within $\{h_k\}$. Note that trivially, $L_1 = N_h$. The **discrete wavelets**, $\{\psi_j\}$, associated with these filters are compactly

supported, of length L_j , and are defined as follows:

$$\psi_j = (\psi_{j,0}, \dots, \psi_{j,(L_j-1)}); \quad (2.31)$$

where the elements of this wavelet are defined recursively by

$$\psi_{1,n} = \sum_k g_{n-2k} \delta_{0,k} = g_n, \quad \text{for } n = 0, \dots, L_1 \Leftrightarrow 1; \quad (2.32)$$

$$\text{and } \psi_{j,n} = \sum_k h_{n-2k} \psi_{j-1,k}, \quad \text{for } n = 0, \dots, L_j \Leftrightarrow 1 \text{ when } j > 1. \quad (2.33)$$

Here $\delta_{0,k}$ is the Kronecker delta. We define the quantity $\psi_{j,k}(\tau)$ to be $\psi_{j,k-\tau}$, the $(k \Leftrightarrow \tau)^{\text{th}}$ element of the vector ψ_j . A related set of discrete father wavelets, ϕ_j , can be constructed similarly, replacing the g_{n-2k} and g_n in equation (2.32) by h_{n-2k} and h_n respectively.

Note that a modified form of the usual Meyer-Mallat scale numbering scheme is adopted here. The scheme which we use is based upon that which is proposed by Nason *et al.* (2000), whereby the time series data live on scale $j = 0$, the finest resolution wavelet detail live on scale $j = 1$ whilst coarser resolution wavelet detail exist on scale $j = J$, for $J = \log_2 N \in \mathbb{N}$.

As we review below, Nason *et al.* (2000) use discrete *non-decimated* wavelets to construct locally stationary stochastic processes. The autocorrelation functions of discrete wavelets are required to obtain a measure of the local autocovariance structure of these processes. These are defined as follows:

Definition 2.6

Let $j \in \mathbb{N}$ and $\tau \in \mathbb{Z}$. Then the **discrete autocorrelation wavelet**, $\Psi_j(\tau)$, is defined by

$$\Psi_j(\tau) = \sum_{k=\max\{0,\tau\}}^{L_j-1+\min\{0,\tau\}} \psi_{j,k} \psi_{j,k-\tau}. \quad (2.34)$$

The **discrete autocorrelation father wavelet**, $\Phi_j(\tau)$, is defined by replacing ψ by ϕ in (2.34).

Autocorrelation wavelets have several interesting and well-known properties. For example, they are compactly supported, positive semi-definite functions defined on $[1 \Leftrightarrow L_j, \dots, L_j \Leftrightarrow 1]$ and are symmetric about $\tau = 0$. For further details see Nason *et al.* (2000).

Finally, we define the inner product matrix of discrete autocorrelation wavelets.

Definition 2.7

Let $J \in \mathbb{N}$. The J -dimensional **discrete autocorrelation wavelet inner product matrix**, A_J , is defined by

$$A_J = (A_{j,k})_{j,k \in \{1, \dots, J\}}, \quad (2.35)$$

where,

$$A_{j,k} = \langle \Psi_j, \Psi_k \rangle := \sum_{\tau=1-\min\{L_j, L_k\}}^{\min\{L_j, L_k\}-1} \Psi_j(\tau) \Psi_k(\tau) \quad (2.36)$$

$$= 1 + 2 \sum_{\tau=1}^{\min\{L_j, L_k\}-1} \Psi_j(\tau) \Psi_k(\tau). \quad (2.37)$$

2.7.4 Model specification

Having summarised the basic building blocks used in this modelling approach, we now define the locally stationary wavelet model proposed by Nason *et al.* (2000).

Definition 2.8

A **locally stationary wavelet (LSW) process** is a sequence of doubly indexed stochastic processes $\{X_{t,T}\}_{t=0, \dots, T-1}$, $T = 2^J \geq 1$ having the following representation:

$$X_{t,T} = \sum_{j=1}^J \sum_k w_{j,k;T} \psi_{jk}(t) \xi_{jk}, \quad (2.38)$$

where $\{\xi_{jk}\}$ are a mutually orthonormal sequence of random variables having zero mean and variance σ^2 , $\{\psi_{jk}(t)\}_{jk}$ is a discrete non-decimated family of wavelets for $j = 1, \dots, J(T)$, $k = 0, \dots, T \Leftrightarrow 1$, whilst the $w_{j,k;T}$ quantify the energy contribution made to the process at location k and scale j .

The above model permits a local representation of structure. To quantify how the size of the amplitudes, $\{w_{j,k}\}$, change over time, Nason *et al.* embed their model into the Dahlhaus (1997) framework, relating the amplitudes to a collection of Lipschitz-continuous functions, $\{W_j(z)\}$, for $z \in (0, 1)$. These, in turn, regulate the degree of local stationarity of $X_{t,T}$. See Nason *et al.* (2000) for further details.

2.7.5 The evolutionary wavelet spectrum

Recall from traditional time series analysis that the classical spectrum quantifies the contribution to variance in a stationary stochastic process over *frequency* (see, for example, Priestley (1981) or Brockwell & Davis (1991)). Analogously, the following evolutionary wavelet spectrum quantifies the contribution to variance within a LSW process over (rescaled) time, $z = k/T$, and scale j .

Definition 2.9

Let $X_{t,T}$ be a LSW process. The **evolutionary wavelet spectrum (EWS)** is defined in terms of $W_j(z)$ and is approximated by

$$S_j(z) \approx |W_j(z)|^2 \quad (2.39)$$

for $j = 1, \dots, J(T)$ and $z \in (0, 1)$.

The LSW model thus delivers a time-scale decomposition paralleling the time-frequency decomposition of Dahlhaus (1997). Additionally, a measure of the local autocovariance may be associated with the EWS. This quantity is defined using the discrete autocorrelation wavelets of Definition 2.6 as follows:

Definition 2.10

Define the local autocovariance (LACV) of an LSW process with EWS $\{S_j(z)\}$ by

$$C(z, \tau) = \sum_{j=1}^{\infty} S_j(z) \Psi_j(\tau) \quad \text{for } \tau \in \mathbb{Z}, z \in (0, 1).$$

The *wavelet periodogram*, $\mathbf{I}(z)$, formed by squaring empirical non-decimated discrete wavelet transform coefficients turns out to be a biased estimate of the EWS (Proposition 4, Nason *et al.* (2000)):

$$\mathbb{E}(\mathbf{I}(z)) = A_J \mathbf{S}(z) + O(T^{-1}), \quad \forall z \in (0, 1). \quad (2.40)$$

Here A_J is the inner product matrix from Definition 2.7, $\mathbf{S}(z) \equiv \{S_j(z)\}_{j=1, \dots, J}$ is the EWS and $\mathbf{I}(z) \equiv \{\mathbf{I}_{l, [zT]}\}_{l=1, \dots, J}$ is the wavelet periodogram. The bias is due to the redundancy of the non-decimated wavelet transform. As can be seen in the simulated example below, the bias spreads power across scales. However, it may be corrected by premultiplying \mathbf{I} by A_J^{-1} . It also transpires that the wavelet periodogram is an inconsistent estimator of the EWS (Proposition 4, Nason *et al.* (2000)) — a result which

parallels the classical approach to time series analysis. Thus, to obtain consistency, the estimator needs to be smoothed (see Section 2.8 for further details).

Example 2.3 *An example of the bias spreading power across scales can be seen in figure 2.13 below. This is based on an EWS of the form:*

$$S_j(z) = \begin{cases} 1 & \text{for } j = 1, z \in (0, 1/4) \\ 1 & \text{for } j = 2, z \in (1/4, 1/2) \\ 1 & \text{for } j = 3, z \in (1/2, 3/4) \\ 1 & \text{for } j = 4, z \in (3/4, 1) \\ 0 & \text{otherwise.} \end{cases} \quad (2.41)$$

Figure 2.13 (i) shows a realisation of a process possessing EWS $S_j(z)$, whilst figure 2.13 (ii) depicts the true, underlying, wavelet spectral structure. An estimate of the (uncorrected) wavelet periodogram for the series appears in figure 2.13 (iii). Each line of the figure denotes biased estimates of $S_j(z)$. The corrected (unbiased) EWS estimate appears in figure 2.13 (iv). Comparing (iii) to (iv) shows how power is spread in the biased estimate (iii) and that (iv) is a better estimate of (ii) than (iii).

2.8 Wavelet shrinkage and thresholding

The problem which we consider here is the standard non-parametric regression problem of trying to recover a signal in the presence of noise. Suppose we have $N (= 2^J)$ noisy samples of a function f :

$$y_i = f(t_i) + \epsilon_i, \quad \text{for } i = 1, \dots, N \quad (2.42)$$

where $t_i = (i \Leftrightarrow 1)/N$ and the ϵ_i are independent, identically distributed $N(0, \sigma^2)$ random variables. Our goal is to estimate the unknown vector $\mathbf{f} = (f(t_1), \dots, f(t_n))$.

The wavelet shrinkage approach to this problem consists of three steps:

Step 1: Decompose the data using the DWT.

Step 2: Modify the coefficients in some way, removing the effects of the noise.

Step 3: Invert the modified wavelet coefficients to obtain an estimate of \mathbf{f} .

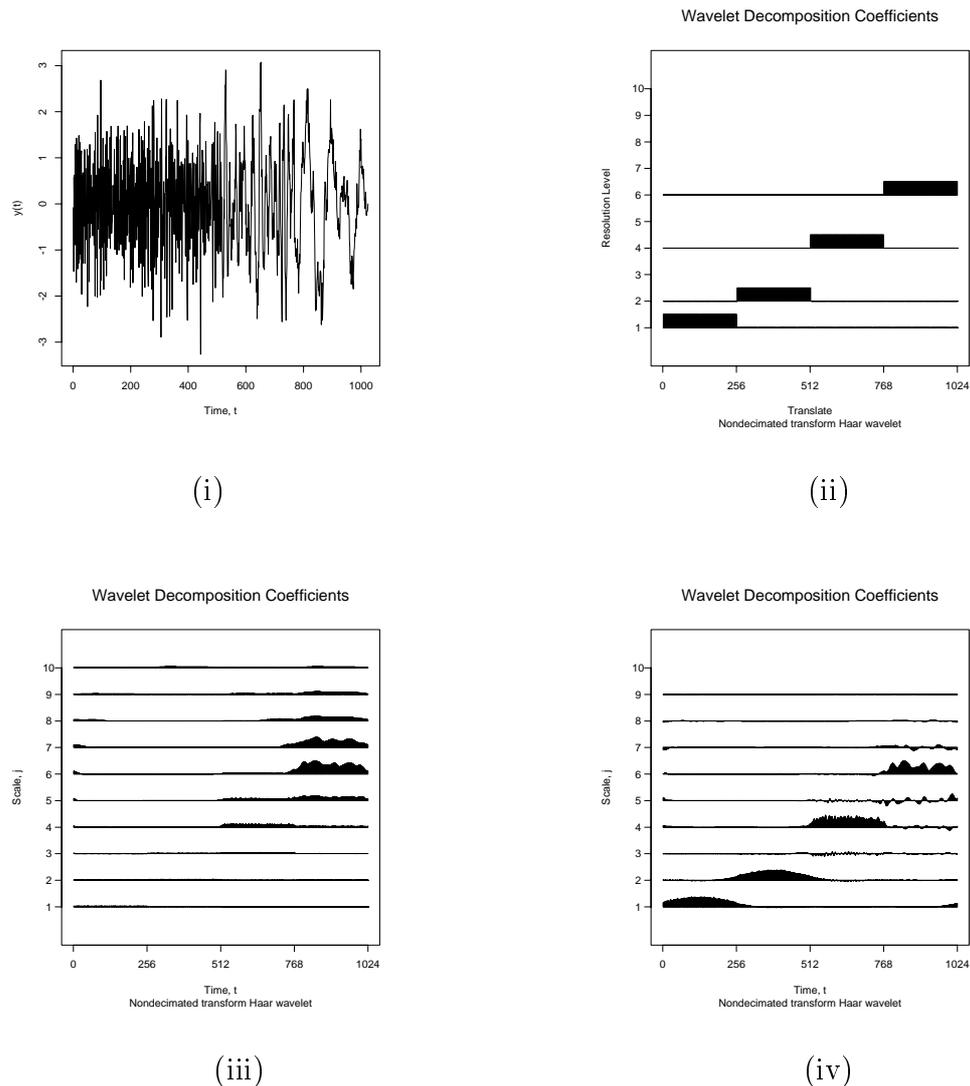


Figure 2.13: (i) A realisation of a LSW process with EWS given in Section 2.3; (ii) True underlying EWS for the realization given in (i); (iii) Mean wavelet periodogram taken over 100 realizations of process with EWS given in (ii); (iv) Mean corrected EWS estimate taken over 100 realizations of process with EWS given in (ii).

Clearly, the key stage in the above is the modification of the wavelet coefficients.

A popular coefficient modification approach, advocated by Donoho & Johnstone (1994) and Donoho *et al.* (1995) is *thresholding*. This method is motivated by two important properties of the DWT. Firstly, as a consequence of the vanishing moments property (condition W3 of Definition 2.2), a smooth function f is sparsely represented in the wavelet domain. Furthermore, as wavelets are well localised in time, a discontinuity in f will only result in large wavelet coefficients, $d_{j,k}$, for those values of k which correspond to the location of the feature. Thus *many* functions, including those with a finite number of discontinuities are sparsely represented in the wavelet

domain.

The second important feature of the DWT is that under the model (2.42), *all* wavelet coefficients will be contaminated by noise. More precisely, as the $\{\epsilon_i\}_{i=1,\dots,N}$ are iid Gaussian, then the wavelet coefficients are also contaminated by independent Gaussian noise. The wavelet coefficients obtained from the Gaussian-corrupted $\{y_i\}$ are thus related to the wavelet coefficients of the true $f(t_i)$, $\{d_{j,k}\}$, as follows:

$$\tilde{d}_{j,k} = d_{j,k} + \epsilon_{j,k}.$$

Here $\tilde{d}_{j,k}$ denotes the corrupted signal wavelet coefficients and the $\epsilon_{j,k}$ are independent Gaussian noise. This feature arises as a consequence of the orthogonality of the DWT.

The essence of thresholding is to determine which wavelet coefficients are noise and which are signal. The two most common thresholding policies are known as *hard* and *soft*. Hard thresholding removes all coefficients which are smaller than a threshold, τ :

$$\delta^H(\tilde{d}_{j,k}, \tau) = \tilde{d}_{j,k} I(|\tilde{d}_{j,k}| > \tau)$$

whilst soft thresholding is defined as follows:

$$\delta^S(\tilde{d}_{j,k}, \tau) = \text{sgn}(\tilde{d}_{j,k})(|\tilde{d}_{j,k}| \ominus \tau) I(|\tilde{d}_{j,k}| > \tau).$$

This second threshold takes into consideration the fact that all wavelet coefficients are contaminated by noise, shrinking larger coefficients by τ and removing the smallest coefficients.

The choice of threshold, τ , is fundamental to the success of this denoising approach. If it is set too high, then signal structure will be lost. Alternatively, if τ is set too low, then noise will be present in the resulting estimate. Below, we consider two of the most popular choices of threshold:

The universal threshold

Many different schemes have been proposed for choosing the threshold level τ . For stimulating reviews see Antoniadis (1997), Abramovich *et al.* (2000) or Vidakovic (Chapter 6, 2000). Possibly the most famous threshold of all is the *universal threshold*, proposed by Donoho & Johnstone (1994). This is given by

$$\tau^{\text{univ}} = \sigma \sqrt{2 \log N},$$

where the noise variance, σ , invariably needs to be estimated from the data. Donoho & Johnstone (1994) suggest that σ be estimated using the median absolute deviation of the finest level wavelet coefficients divided by 0.6745. This constant is simply the median of a standard normal variable $X \sim N(0, 1)$.

The universal threshold is a noise suppressor and can be motivated by the following argument: suppose we have a sequence of iid white noise, $Z_i \sim N(0, 1)$ for $i = 1, \dots, N$. Then, as a consequence of a result due to von Mises, as $n \rightarrow \infty$,

$$P(\max_i |Z_i| > \sqrt{2 \log n}) \rightarrow 0.$$

Thus, all the pure noise coefficients will be thresholded to zero with high probability. A consequence of this feature is that the universal threshold tends to over-smooth in practice.

SURE threshold selection

An alternative threshold, proposed by Donoho & Johnstone (1995) is based upon the minimization of Stein's unbiased risk estimator (SURE). In considering the problem of estimating the mean of a multivariate normal distribution, Stein (1981) proves the following result:

Theorem 2.1 (Theorem 1, Stein 1981)

Let $d_i \sim N(\theta_i, 1)$ for $i = 1, \dots, N$ be iid random variables and let $\hat{\theta}$ be an estimator of θ of the form $\hat{\theta}(\mathbf{d}) = \mathbf{d} + \mathbf{g}(\mathbf{d})$. If $\mathbf{g} = \{g_i\}_{i=1}^N$ is weakly differentiable, then

$$\mathbb{E}_\theta \|\hat{\theta} \leftrightarrow \theta\| = N + \mathbb{E}_\theta \{ \|\mathbf{g}(\mathbf{d})\|^2 + 2\nabla \mathbf{g}(\mathbf{d}) \},$$

where $\nabla \mathbf{g} = \sum_{i=1}^N \frac{\partial}{\partial d_i} g_i$.

Recall that the soft threshold is defined as

$$\delta^S(\tilde{d}_i, \tau) = \text{sgn}(\tilde{d}_i)(|\tilde{d}_i| \leftrightarrow \tau)I(|\tilde{d}_i| > \tau).$$

This may be re-written as

$$\delta^S(\tilde{d}_i, \tau) = \tilde{d}_i \leftrightarrow \text{sgn}(\tilde{d}_i) \min(|\tilde{d}_i|, \tau).$$

Thus, using Stein's result,

$$\text{SURE}(\tau, \tilde{\mathbf{d}}) = N \leftrightarrow 2 \sum_{i=1}^N I(|\tilde{d}_i| \leq \tau) + \sum_{i=1}^N \min(|\tilde{d}_i|, \tau)^2$$

is an unbiased estimate of the risk $\mathbb{E}_\theta \|\hat{\theta} \Leftrightarrow \theta\|$. This motivates the following choice of threshold

$$\tau^{\text{SURE}} = \operatorname{argmin}_{0 \leq \tau \leq \sqrt{2 \log N}} \text{SURE}(\tau, \tilde{\mathbf{d}}).$$

The *SureShrink* procedure adopted by Donoho & Johnstone (1995) is in fact a hybrid thresholding approach, utilising both the universal and the SURE thresholds. If the wavelet representation at a given level is sparse, then the universal threshold is selected. Otherwise, if the representation is not too sparse, the level-dependent SURE threshold, τ_j^{SURE} , is used. The motivation behind this approach is that in situations where the representation is sparse, noise will swamp the information contributed to SURE, resulting in over-smoothing.

Several other popular threshold policies exist. See for example Goel & Vidakovic (1995), Nason (1996), Abramovich & Benjamini (1996) and Abramovich, Sapatinas & Silverman (1998). Vidakovic (Section 6.6, 1999) provides an interesting review of these and other approaches.

Thresholding correlated data

The thresholding schemes considered above focus on the situation where the noise is assumed to be uncorrelated. Johnstone & Silverman (1997) consider an alternative scenario, supposing that the noise process is correlated and stationary. They highlight two interesting features of the wavelet transform, namely that:

1. within each level, j , the autocorrelation of the $d_{j,k}$ decays rapidly,
2. there tends to be little, or even no correlation between levels.

In other words, the wavelet transform decorrelates data. As a consequence, Johnstone & Silverman (1997) propose the use of a level-dependent thresholding scheme in the case of correlated noise.

Smoothing the wavelet periodogram

Recall from Section 2.7.5 that the *local wavelet periodogram* is an inconsistent estimator of the local wavelet spectrum. Thus, to obtain consistency we must smooth this

estimator. In this section, we review the application of wavelet thresholding to this problem, considering the contribution of Nason *et al.* (2000).

For each fixed scale, j , Nason *et al.* (2000) consider using non-linear wavelet thresholding to smooth the $\{\widehat{S}_j(z) = I_{j,[zT]}\}_{z \in (0,1)}$, before applying the inverse of the inner product matrix, A , to achieve an asymptotically unbiased estimator of the wavelet spectrum. The motivation behind this approach is that the problem of applying wavelet shrinkage to χ^2 -distributed random variables results in a considerably simpler asymptotic analysis than the alternative of correcting by A^{-1} *prior* to smoothing. Furthermore, the problem of smoothing χ^2 -distributed random variables via wavelet thresholding has also been studied in the context of both estimating the classical periodogram and Dahlhaus' evolutionary spectrum. See, for example Neumann and von Sachs (1995, 1997), von Sachs & Schneider (1996) and Gao (1997).

The smoothing scheme is implemented using an orthonormal second-stage wavelet basis of $L_2([0, 1])$, the first-stage wavelet basis being that which is used in the modelling of the process. Having transformed a given wavelet periodogram, $I_{j,[zT]}$, into the wavelet domain, the resulting wavelet coefficients, $\{\widehat{d}_{l,m}\}$, are denoised via non-linear thresholding. Finally, the smoothed EWS estimator, $\widetilde{S}_j(z)$, is obtained by inverting the thresholded wavelet coefficients.

Drawing on the work of Neumann & von Sachs (1995), Nason *et al.* (2000) advocate the use of a “universal” threshold

$$\tau^2(l, m; j) = \text{Var}(\widehat{d}_{l,m}) \log^2(T)$$

for Gaussian LSW processes. This threshold takes the heavier tails of the χ^2 -distribution into account. For non-normality, techniques such as those proposed by Neumann & von Sachs (1997) using the threshold

$$\tau^2(l, m; j) = \text{Var}(\widehat{d}_{l,m}) \log(T)$$

are advocated.

The logarithmic transformation of the periodogram has the effect of stabilising the variance of the periodogram coefficients, pulling their distribution closer to normality. See for example, Stuart, Ord & Arnold (Sections 32.38–32.40, 1999). Thus an alternative approach to the problem of obtaining consistency is to consider

smoothing the log-wavelet periodogram. This permits the use of a universal threshold for (asymptotically) normally distributed data and parallels existing techniques for smoothing the classical (e.g. Priestley (1981)) and evolutionary periodograms (e.g. von Sachs & Schneider (1996)).

2.9 Summary

This chapter has summarised the foundations of wavelet theory required for the work presented in this thesis. Having introduced the concept of a wavelet and multiresolution analysis, we considered the discrete and non-decimated wavelet transforms. Both transforms provide a localised, scale-based decomposition of the information contained within a signal. However, whilst being an orthogonal transform, it is important to appreciate that the discrete wavelet transform is *not* translation equivariant. In other words, a simple shift in a data sequence does not generally manifest itself as a simple coefficient shift in the transform domain. By contrast, the non-decimated wavelet transform *is* translation equivariant and also provides more information than the discrete wavelet transform at medium and low resolution levels. The price we pay for this additional information is a non-orthogonal representation of the original sequence.

The second half of the chapter focused on recent research which has applied wavelets to time series analysis and non-parametric regression. Particular attention was paid to the work of Nason *et al.* (2000), which introduced the concept of locally stationary wavelet processes to time series analysis. In Chapter 3, we will extend this modelling approach to two-dimensions, proposing a multiscale model of the covariance structure contained within locally stationary data which lie on a regular grid.

Chapter 3

Locally stationary wavelet processes

In this chapter we propose and investigate a model of the covariance structure for random fields which lie on a regular grid. Traditional techniques of modelling the covariance of such processes have dealt with situations where one can model the covariance, $C_X(\mathbf{y}, \mathbf{z}) = \text{Cov}(X_{\mathbf{y}}, X_{\mathbf{z}})$, as a function of $\mathbf{z} \Leftrightarrow \mathbf{y}$ or $\|\mathbf{z} \Leftrightarrow \mathbf{y}\|$. We depart from such approaches and instead consider the situation where $C_X(\mathbf{y}, \mathbf{z})$ is a function of both \mathbf{y} and \mathbf{z} . To model this alternative form, we move from using a (global) Fourier decomposition of the covariance to a *localised* decomposition involving a set of non-decimated wavelets. Proofs of results stated in this chapter may be found in Appendix A.

3.1 Motivation

Suppose, by way of introduction, that we have a random field which lies on a regular grid, $\{X_{\mathbf{r}}\}_{\mathbf{r} \in \mathbb{Z}^2}$, for which we wish to estimate the covariance, $\text{Cov}(X_{\mathbf{r}}, X_{\mathbf{s}}) = \gamma_{\mathbf{r}, \mathbf{s}}$. The covariance structure could take many different forms, one of the simplest being that the field is (second-order) *stationary*: $\gamma_{\mathbf{r}, \mathbf{s}} = \gamma(\mathbf{r} \Leftrightarrow \mathbf{s}) \forall \mathbf{r}, \mathbf{s}$. In other words, the covariance is a function of the vector difference between the two locations. As such, given a realisation of a stationary process, $\{x_{\mathbf{r}}\}$ where $\mathbf{r} = (r, s)$ and $r = \{1, \dots, R\}$, $s = \{1, \dots, S\}$, the covariance may be estimated by:

$$\hat{\gamma}(\tau_1, \tau_2) = \frac{1}{RS} \sum_{r=1}^{R-\tau_1} \sum_{s=1}^{S-\tau_2} x_{r,s} x_{r+\tau_1, s+\tau_2}.$$

Should the size, i.e. the number of process observations $R \times S$, increase, this estimate will improve (see Priestley (Section 9.7, 1981) for further details).

Alternatively, the process may be *intrinsically* stationary. In other words the *variance* is a function of the vector difference between two locations: $\text{Var}(X_{\mathbf{r}} \Leftrightarrow X_{\mathbf{s}}) = 2\nu(\mathbf{r} \Leftrightarrow \mathbf{s}) \forall \mathbf{r}, \mathbf{s}$. Following Cressie (1991), let

$$N(\boldsymbol{\tau}) = \{(\mathbf{r}, \mathbf{s}) : \mathbf{r} \Leftrightarrow \mathbf{s} = \boldsymbol{\tau}, \text{ for } \mathbf{r}, \mathbf{s} \text{ and } \boldsymbol{\tau} \in \mathbb{Z}^2\}$$

and set $|N(\boldsymbol{\tau})|$ to be the number of distinct pairs contained within $N(\boldsymbol{\tau})$. Then given a realisation, $\{x_{\mathbf{r}}\}$, of an intrinsically stationary process, the method of moments estimator of the variogram is given by

$$2\hat{\nu} = \frac{1}{|N(\boldsymbol{\tau})|} \sum_{N(\boldsymbol{\tau})} (x_{\mathbf{r}} \Leftrightarrow x_{\mathbf{s}})^2.$$

Conversely, we may suspect that the covariance structure changes as a function of location – i.e. that

$$\text{Cov}(X_{\mathbf{r}}, X_{\mathbf{s}}) = \gamma(\mathbf{r}, \mathbf{s}), \forall \mathbf{r}, \mathbf{s} \in \mathbb{Z}^2$$

where $\gamma(\mathbf{r}, \mathbf{s}) = \gamma(\mathbf{t}, \mathbf{u})$ if, and only if, $\mathbf{r} = \mathbf{t}$ and $\mathbf{s} = \mathbf{u}$. In this case, the field may possess a highly non-stationary form. Furthermore, as the only information about $\gamma(\mathbf{r}, \mathbf{s})$ comes from the single observed $x_{\mathbf{r}}$ and $x_{\mathbf{s}}$, there is little hope of obtaining a reliable estimate of the covariance from a single realisation of the process.

The final form of covariance which we consider is to suppose that the structure changes slowly as a function of location. Heuristically speaking, such processes appear to be stationary on close range inspection. Thus, the covariance around a particular location, \mathbf{r} , may be estimated by pooling information from those $X_{\mathbf{r}}$ close to \mathbf{r} , assuming that sufficient data can be collected. Fields which exhibit this slowly varying structure are what we term to be *locally stationary random fields*.

Many of the scenes and images which we encounter in our everyday lives, although highly complex processes, possess a locally stationary, multi-scale structure. For example, figure 3.1, is characterised by fine scale detail in the region of the hair, edges (i.e. discontinuities) defining the dominant features of the face and a gradual change of lighting across the skin. Clearly, the variance structure within such an image is neither

stationary nor does it appear to be a rapidly varying function of location. Rather, it is *locally* stationary.



Figure 3.1: An image of a face. Image provided by Unilever Research.

In recent years there have been a number of developments in the modelling of non second-order stationary spatial processes. See for example Haas (1990), Sampson & Guttorp (1992), Loader & Switzer (1992), Le & Zidek (1992), Le, Sun & Zidek (1997) and Higdon, Swall & Kern (1999). Of these, only Haas (1990) and Higdon *et al.* (1999) provide methods for fitting a non-stationary model using a *single* realisation. Furthermore, as these studies have considered the more general problem of modelling spatial processes which do *not* fall on a regular grid, there has been no formal consideration of the concepts of local stationarity or scale structure — features which are arguably of importance when considering the structure within images, for example.

In this chapter, we attempt to model locally stationary processes in two dimensions using wavelets, thus building upon the work Nason *et al.* (2000). In Section 3.2 we introduce our basic building block, the discrete non-decimated wavelet, before proceeding to define and give examples of our random field model in Section 3.3. A local measure of power, called the *local wavelet spectrum*, is proposed in Section 3.4. As this measure is based upon the redundant NDWT, it is important that we establish its uniqueness: this problem is considered in Section 3.5. We conclude, in Section 3.7, by introducing a measure of the local autocovariance structure.

3.2 Discrete wavelets

Prior to defining our model for locally stationary random fields which lie on a regular grid, we introduce the basic building block of our model. Naturally, when dealing with data which are inherently discrete, we must use a discrete transform. Below, we follow Nason *et al.* (2000) and use a set of discrete (non-decimated) wavelets founded upon the low and high-pass quadrature mirror filters used in the construction of Daubechies' compactly supported wavelets (Daubechies 1988, 1992). See Section 2.7.3 for further details.

A brief account of the history of the non-decimated wavelet transform, together with an explanation of the transforms construction, is provided by Vidakovic (1999) whilst a brief description of the non-decimated transform in two-dimensions is provided in Chapter 2. The key difference between the traditional, decimated, wavelet transform and its non-decimated counterpart is that non-decimated wavelets are positioned at every location $\mathbf{x} \in \mathbb{Z}^2$ within each scale of a decomposition. This is in stark contrast to the standard scheme, where wavelets only appear at locations $2^j \mathbf{x}$ at each scale j .

Following the separable construction suggested by Mallat (1989b), we define *discrete two-dimensional* wavelets as a suitable product of the discrete wavelets and father wavelets defined in Definition 2.5:

Definition 3.1

Let $\{h_n\}_{n \in \mathbb{Z}}$ and $\{g_n\}_{n \in \mathbb{Z}}$ be the usual low and high-pass quadrature mirror filters associated with the construction of Daubechies' compactly supported, continuous-time, wavelets. Further, let $\mathbf{k} = (k_1, k_2)$. Then the **two-dimensional discrete wavelets**, $\{\psi_j^l\}$, associated with these filters are compactly supported, of dimension L_j^2 , and are defined as follows:

$$\psi_j^l = \begin{bmatrix} \psi_{j,(0,0)}^l & \cdots & \psi_{j,(0,L_j-1)}^l \\ \vdots & \vdots & \vdots \\ \psi_{j,(L_j-1,0)}^l & \cdots & \psi_{j,(L_j-1,L_j-1)}^l \end{bmatrix} \quad \text{for } l = h, v \text{ or } d,$$

where h denotes the wavelet in the horizontal direction, v the vertical direction and d

the diagonal direction. The constituent elements of these wavelets are defined by:

$$\left. \begin{aligned} \psi_{j,\mathbf{k}}^h &\equiv \phi_{j,k_1} \psi_{j,k_2} \\ \psi_{j,\mathbf{k}}^v &\equiv \psi_{j,k_1} \phi_{j,k_2} \\ \text{and } \psi_{j,\mathbf{k}}^d &\equiv \psi_{j,k_1} \psi_{j,k_2} \end{aligned} \right\} \text{for } k_1, k_2 = 0, \dots, L_j \Leftrightarrow 1, \quad (3.1)$$

where $\psi_{j,k}$ and $\phi_{j,k}$ are the one-dimensional discrete wavelets defined in Definition 2.5. Similarly, the two-dimensional discrete father wavelet is defined as:

$$\phi_{j,\mathbf{k}} \equiv \phi_{j,k_1} \phi_{j,k_2}. \quad (3.2)$$

It is important to note that a modified form of the usual Meyer-Mallat scale numbering scheme is adopted here. The scheme which we use is based upon that proposed by Nason *et al.* (2000): the data live on scale 0, the finest resolution wavelet detail live on scale 1 and coarser resolution wavelet detail exist on scale J , for $J \in \mathbb{N}$. The advantage of this altered numbering scheme is that we keep the support of the wavelets on the finest scale fixed and constant with respect to the size of the observed field. More importantly, as the size increases, we can observe broader and broader cycles within the process.

As a consequence of adopting a separable two-dimensional discrete wavelet construction, any given two-dimensional discrete wavelet may be formed by taking a tensor product of the appropriate discrete mother and father wavelets. For example, the finest scale discrete Haar wavelet in the diagonal decomposition direction is given by:

$$\psi_1^d = \psi_1 \otimes \psi_1 = \begin{bmatrix} 1/2 & \Leftrightarrow 1/2 \\ \Leftrightarrow 1/2 & 1/2 \end{bmatrix}; \quad (3.3)$$

whilst, the second finest scale discrete Haar wavelet in the vertical direction is given by

$$\psi_2^v = \psi_2 \otimes \phi_2 = \begin{bmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 1/4 & 1/4 \\ \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 \\ \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 \end{bmatrix}. \quad (3.4)$$

3.3 Locally stationary wavelet random fields

In this section we define a new class of processes, the locally stationary two-dimensional wavelet process, thereby extending the recent time series work of Nason *et al.* (2000). Several assumptions are made about the structure of such processes, these being stated separately from the process definition. In Section 3.3.1 examples are provided which demonstrate the ability of this approach to capture both stationary and non-stationary structure.

Definition 3.2

Let $\mathbf{R} = (R, S)$ where $R = 2^m$, $S = 2^n \geq 1$ for $m, n \in \mathbb{N}$ and set $J(R, S) = \log_2[\min(R, S)]$. Further, let $\mathbf{r} = (r, s)$ and $\mathbf{u} = (u, v)$ for $\mathbf{r}, \mathbf{u} \in [0, R \Leftrightarrow 1] \times [0, S \Leftrightarrow 1]$. Then a class of **locally stationary two-dimensional wavelet processes (LS2W)** is defined to be a sequence of stochastic processes which lie on a regular grid:

$$\{X_{\mathbf{r};\mathbf{R}}\}_{\mathbf{r} \in [0, R-1] \times [0, S-1]}. \quad (3.5)$$

Such processes have the following representation in the mean-square sense:

$$X_{\mathbf{r};\mathbf{R}} = \sum_l \sum_{j=1}^J \sum_{\mathbf{u}} w_{j,\mathbf{u};\mathbf{R}}^l \psi_{j,\mathbf{u}}^l(\mathbf{r}) \xi_{j,\mathbf{u}}^l, \quad (3.6)$$

where the sum over l is over decomposition directions v , h and d . The decomposition consists of amplitudes, $\{w_{j,\mathbf{u};\mathbf{R}}^l\}$, which quantify the contribution made to the process at location (u, v) in direction l at scale j ; a discrete non-decimated family of wavelets

$$\begin{aligned} \{\psi_{j,\mathbf{u}}^l(\mathbf{r})\} & \quad \{j = 1, \dots, J(R, S)\} \\ & \quad \{\mathbf{u} \in [0, R-1] \times [0, S-1]\} \end{aligned}$$

and a random orthonormal increment sequence, $\xi_{j,u,v}^l$:

$$\text{Cov} [\xi_{j,\mathbf{k}}^l, \xi_{m,\mathbf{n}}^p] = \delta_{j,m} \delta_{\mathbf{k},\mathbf{n}} \delta_{l,p}, \quad (3.7)$$

where $l, p \in \{h, v, d\}$.

We shall henceforth drop the dependence on \mathbf{R} , though naturally it is still assumed.

Model interpretation: The above model permits a *local* representation of a random field, capitalising on the wavelet property that fast (high frequency) oscillations can

change quickly whilst slow oscillations vary slowly. It possesses the ability to decompose the covariance structure of a random field into contributions made by each location at various scale-direction pairs (j, l) . These contributions are measured by the amplitudes $\{w_{j,\mathbf{u}}^l\}$. In broad terms, such amplitudes are expected to be large if there is a high correlation between $X_{\mathbf{u}}$ and $X_{\mathbf{u}-\boldsymbol{\tau}}$ or $X_{\mathbf{u}+\boldsymbol{\tau}}$, for some lag $\boldsymbol{\tau}$ which matches the associated “wavelength” of $\psi_{j,\mathbf{u}}(\mathbf{x})$.

Modelling assumptions

In order that we can estimate the local covariance structure of a LS2W process, we must naturally impose some assumptions on the behaviour of such a process.

Assumption 1: The first assumption which we make is that

$$\mathbb{E}[\xi_{j,\mathbf{u}}^l] = 0, \quad \forall j, \mathbf{u} \text{ and } l. \quad (3.8)$$

Hence, for all locations $\mathbf{r} \in [0, R \Leftrightarrow 1] \times [0, S \Leftrightarrow 1]$, it follows that $\mathbb{E}(X_{\mathbf{r}}) = 0$. In other words, all LS2W processes have zero mean. Naturally, this is unlikely to be the case in reality. Thus, should a non-zero trend exist, it should be estimated and subsequently removed if one is to use the modelling approach which we describe here. Such trend removal could be carried out in any number of ways, for example using median polish, multivariate regression or a two-dimensional extension of the approach proposed by von Sachs & MacGibbon (2000).

It is important to understand that we are *not* observing a fixed continuous spatial process on an increasingly finer grid as $\min(R, S) \rightarrow \infty$. To overcome this handicap, we draw inspiration from Dahlhaus (1997), adopting *rescaled* location in our final assumption. In other words, we re-scale as follows:

$$z_u = \frac{u}{R} \text{ and } z_v = \frac{v}{S}, \quad \text{where } z_u, z_v \in (0, 1);$$

thus permitting increasing amounts of information to be collected about the local structure of a location \mathbf{z} as $\min(R, S) \rightarrow \infty$.

Assumption 2: Within each decomposition direction, l , there exists a Lipschitz-continuous function (with respect to the L_1 -norm) $W_j^l(\mathbf{z})$ for each level $j \geq 1$, where $\mathbf{z} \in (0, 1)^2$. These functions satisfy the following properties $\forall j$ and l :

1.

$$\sum_l \sum_{j=1}^{\infty} |W_j^l(\mathbf{z})|^2 < \infty \quad (3.9)$$

uniformly in $\mathbf{z} \in (0, 1)^2$;

2. The Lipschitz constants, L_j^l , of W_j^l are uniformly bounded in j, l and

$$\sum_l \sum_{j=1}^{\infty} 2^{2j} L_j^l < \infty. \quad (3.10)$$

3. Let $\frac{\mathbf{u}}{\mathbf{R}} = (\frac{u}{R}, \frac{v}{S})$. Then there exists a sequence of constants C_j^l such that for each dimension set \mathbf{R} ,

$$\sup_{\mathbf{u}} \left| w_{j,\mathbf{u}}^l \Leftrightarrow W_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right| \leq \frac{C_j^l}{\max\{R, S\}} \quad (3.11)$$

where for each $j = 1, \dots, J(\mathbf{R}) = \log_2(\min(R, S))$ the sup is over all pairs of coordinates $\mathbf{u} \in [0, R] \times [0, S]$ and where $\{C_j^l\}$ fulfills

$$\sum_l \sum_{j=1}^{\infty} C_j^l < \infty. \quad (3.12)$$

The essence of this final assumption is to attempt to encapsulate the way in which the *local* structure of the LS2W process becomes increasingly stationary with respect to a given wavelet basis as we obtain more information. This is achieved by introducing the $\{W_j^l(\mathbf{z})\}$ and imposing certain restrictions about their smoothness (variation), namely

1. the Lipschitz-continuity of the functions,
2. and through the variation limiting constants $\{C_j^l / \max(R, S)\}$.

The variation-limiting constants simply restrict the amount by which the $\{w_{j,\mathbf{u}}^l\}$ are allowed to differ from the smooth locally stationary “amplitudes” $\{W_j^l(\mathbf{z})\}$. Together, these conditions ensure that the region which determines the variation at each spatial location becomes asymptotically arbitrarily small.

Note that whilst we have assumed here that the $W_j^l(\mathbf{z})$ are Lipschitz with respect to the L_1 -norm, other norms could be chosen. Naturally, a change of norm will affect some of the results which we state later in this chapter.

3.3.1 Examples of LS2W processes

The representation in equation (3.6) permits contributions at various scales within the vertical, horizontal and diagonal decomposition directions. Thus, we can construct stationary and non-stationary processes which possess features in one or more of these directions.

Example 3.1 *Possibly the simplest LS2W processes which we can construct consists of contributions made in only one decomposition direction. Let us therefore assume that no contributions are made by structure in either the horizontal or vertical directions. Thus, from (3.6), we have*

$$X_{\mathbf{r}}^{1,d} = \sum_{j=1}^J \sum_{\mathbf{u}} w_{j,\mathbf{u}}^d \psi_{j,\mathbf{u}}^d(\mathbf{r}) \xi_{j,\mathbf{u}}^d, \quad (3.13)$$

for some $J \in \mathbb{N}$. Let us also assume that the finest scale detail ($j = 1$) is the only scale at which a contribution is made to the process:

$$X_{\mathbf{r}}^{1,d} = \sum_{\mathbf{u}} w_{1,\mathbf{u}}^d \psi_{1,\mathbf{r}-\mathbf{u}}^d \xi_{1,\mathbf{u}}^d.$$

If, furthermore,

1. the ψ are non-decimated Haar wavelets;
2. $\xi_{1,\mathbf{u}}^d = \epsilon_{\mathbf{u}}$, where $\{\epsilon_{\mathbf{u}}\}$ is a purely random process with zero mean and variance σ^2 ;
3. and finally, that $w_{1,\mathbf{u}}^d = 1$ for all locations \mathbf{u} , which implies that $W_1^d(z) = 1$ for this example;

then the following process is obtained:

$$X_{r,s}^{1,d} = \frac{1}{2}(\epsilon_{r,s} \Leftrightarrow \epsilon_{r,s+1} \Leftrightarrow \epsilon_{r+1,s} + \epsilon_{r+1,s+1}). \quad (3.14)$$

The above is an example of a moving average field (Definition 3.3) of order (1,1) which uses the wavelets given in equation (3.3). A realisation of such a process is displayed in figure 3.2(a).

Example 3.2 *Scale 2 Haar wavelets can be used to form a LS2W process with “broader” features. By combining contributions from both the vertical and horizontal*

directions ($j = 2$), we obtain the following moving average field of order $(3,3)$, a realisation of which may be seen in figure 3.2(b):

$$\begin{aligned} X_{\mathbf{r}}^2 &= \sum_{\mathbf{u}} w_{w,\mathbf{u}}^h \psi_{2,\mathbf{u}}^h(\mathbf{r}) \xi_{2,\mathbf{u}}^h + \sum_{\mathbf{u}} w_{w,\mathbf{u}}^v \psi_{2,\mathbf{u}}^v(\mathbf{r}) \xi_{2,\mathbf{u}}^v, & (\text{set } w_{2,\mathbf{u}}^h = 1 = w_{2,\mathbf{u}}^v \forall \mathbf{u}); \\ &= (\epsilon_{r,s} + \epsilon_{r,s+1} + \epsilon_{r+1,s} + \epsilon_{r+1,s+1}) \\ &\Leftrightarrow (\epsilon_{r+2,s+2} + \epsilon_{r+2,s+3} + \epsilon_{r+3,s+2} + \epsilon_{r+3,s+3}). \end{aligned}$$

Here $\{\epsilon_{r,s}\}$ is a zero-mean random orthonormal process.

More generally a moving average (MA) field of order $(2^j \Leftrightarrow 1, 2^j \Leftrightarrow 1)$, $\{X_{\mathbf{r}}^j\}$, may be constructed using scale j discrete non-decimated Haar wavelets. Such processes fall into a class similar to that defined by Haining (1978). This class of processes may be defined as follows:

Definition 3.3

Let $\{X_{r,s}\}$ be a random field on a grid of dimension $R \times S$ and let $m, n \in \mathbb{Z}$. Then $\{X_{r,s}\}$ is said to be a moving average field of order (m,n) if

$$X_{r,s} = \sum_{p=0}^m \sum_{q=0}^n c_{p,q} \epsilon_{r+m-p, s+n-q}; \quad (3.15)$$

where $\{\epsilon_{p,q}\}$ is a random orthonormal process and the $\{c_{p,q}\}$ are constants.

See Moore (1988) or Cressie (1991) for further details.

As we have seen in the above examples, a special class of MA process can be generated using Haar wavelets at various directions and scales:

Definition 3.4

Let $c \in \mathbb{R}$. A Haar MA field of order j_0 , in direction l_0 , is defined to be the LS2W process $\{X_{\mathbf{r}}^{j_0, l_0}\}$ generated by the Haar wavelet family with the following condition on the amplitudes:

$$w_{j,\mathbf{u}}^l = \begin{cases} c & \text{for } j = j_0 \text{ when } l = l_0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.16)$$

Thus, for example, the process defined in Example 3.1 is a Haar MA field of order 1 in the diagonal decomposition direction.

Any two-dimensional MA field can be represented as a linear combination of Haar MA fields. This follows as a consequence of the property that any sequence in $l^2(\mathbb{Z}^2)$

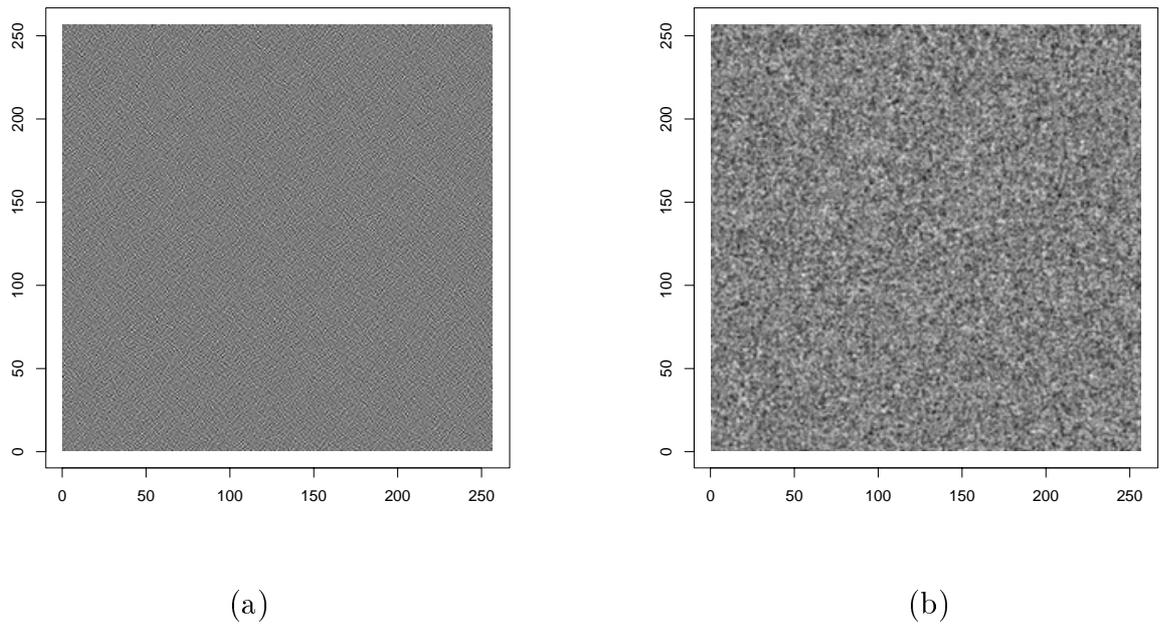


Figure 3.2: Haar moving average fields. (a) A 2D Haar MA process of order 1. (b) A 2D Haar MA process of order 2.

can be decomposed using Haar non-decimated wavelets (but not uniquely). Often, as in Examples 3.1 and 3.2, the representation is sparse. Note also, that other forms of MA fields may be constructed using non-decimated Daubechies wavelets. However, the real potential of the LS2W model lies in its ability to capture certain forms of non-stationary behaviour.

Example 3.3 *One of the simplest examples of a non-stationary LS2W process is given by the following:*

$$X_{\mathbf{r}} = \sum_{j=1}^4 \sum_{\mathbf{u}} w_{j,\mathbf{u}}^d \psi_{j,\mathbf{u}}^d \zeta_{j,\mathbf{u}}^d, \quad (3.17)$$

where $\{\psi_{j,\mathbf{u}}^l\}$ is the Haar family of wavelets and

$$w_{j,\mathbf{u}}^d = \begin{cases} 1 & \text{for } j = 1, \quad \mathbf{u} = [0, \dots, 2^{J-1} \Leftrightarrow 1] \times [0, \dots, 2^{J-1} \Leftrightarrow 1]; \\ 1 & \text{for } j = 2, \quad \mathbf{u} = [2^{J-1} \Leftrightarrow 1, \dots, 2^J \Leftrightarrow 1] \times [0, \dots, 2^{J-1} \Leftrightarrow 1]; \\ 1 & \text{for } j = 3, \quad \mathbf{u} = [0, \dots, 2^{J-1} \Leftrightarrow 1] \times [2^{J-1}, \dots, 2^J \Leftrightarrow 1]; \\ 1 & \text{for } j = 4, \quad \mathbf{u} = [2^{J-1} \Leftrightarrow 1, \dots, 2^J \Leftrightarrow 1] \times [2^{J-1}, \dots, 2^J \Leftrightarrow 1]; \\ 0 & \text{otherwise.} \end{cases}$$

(which implies that the $\{W_j^l(\mathbf{z})\}$ have an equivalent non-constant form for the four finest scales in the diagonal direction).

A realisation of this process is displayed in figure 3.3. Each quadrant of the process is stationary. However, the change in the covariance structure from one region to another is visible to the eye. We will revisit this particular process in the next section, when we consider the problem of measuring the local power contained within a LS2W process.

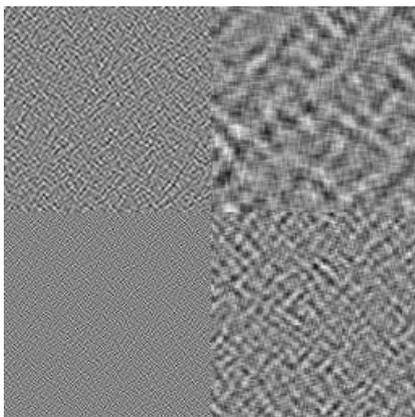


Figure 3.3: Realisation of a non-stationary LS2W process.

3.4 Measuring local power

As can be seen in Example 3.3, the covariance structure of many LS2W processes varies from one region to another. It is therefore important to be able to capture and quantify this local behaviour. In this section we propose a measure of the local power (i.e. contribution to variance) present in a LS2W process. The measure is founded upon the “amplitudes”, $W_j^l(\mathbf{z})$. This approach is therefore analogous to that of the traditional, stationary, approach whereby the spectrum of a stationary field is estimated by taking the square of the Fourier transform of the process. In the second half of this section, we consider the estimation of wavelet spectra via the non-decimated wavelet transform of a realisation.

The *local wavelet spectrum*, which we define below, measures the local power of an LS2W process at a specific (rescaled) location, $\mathbf{z} \in (0, 1)^2$, and scale, j , in a given direction l .

Definition 3.5

For $R, S \in \mathbb{N} \setminus 1$, we define the **local wavelet spectrum** (LWS) of a LS2W process, $\{X_{\mathbf{r}}\}$, to be given by

$$S_j^l(\mathbf{z}) \equiv |W_j^l(\mathbf{z})|^2, \quad (3.18)$$

for $\mathbf{z} \in (0, 1)^2$, $j \in 1, \dots, J$ and $l \in \{h, v, d\}$.

In other words, the LWS provide a form of location-direction-scale decomposition of the structure within a LS2W process. These channels tend to extract structure from the horizontal, vertical and diagonal directions respectively. As a consequence of Assumption 2 (pp. 51–52) which we make about LS2W processes, the LWS can also be measured (asymptotically) using the innovations $\{w_{j,\mathbf{u}}^l\}$. Note that the LWS is defined only for $\mathbf{z} \in (0, 1)^2$. This is because boundaries do not make any sense in this framework.

Property 3.1

Let $\mathbf{z} = (z_u, z_v)$. Assumption 2 means that

$$S_j^l(\mathbf{z}) = \lim_{R,S \rightarrow \infty} |w_{j,[z_u R],[z_v S]}^l|^2, \quad \text{for } \mathbf{z} \in (0, 1)^2, \quad (3.19)$$

and satisfies

$$\sum_l \sum_{j=1}^{\infty} S_j^l(\mathbf{z}) < \infty \quad \text{uniformly for } \mathbf{z} \in (0, 1)^2. \quad (3.20)$$

A plot of $S_j^l(\mathbf{z})$ for fixed direction, l , and scale, j , therefore provides a map of the contribution to structure made over the entire region. The following provides some examples of LWS for stationary and a non-stationary processes.

Example 3.4 Define the LWS of the order k Haar MA field in the decomposition direction m to be $\{S_j^l(\mathbf{z})\}$. Then, by Definition 3.5:

$$\begin{aligned} S_j^l(\mathbf{z}) &= |W_j^l(\mathbf{z})|^2; \\ &= \begin{cases} 1 & \text{for } j = k \text{ and } l = m, \\ 0 & \text{otherwise;} \end{cases} \\ &= \delta_{l,m} \delta_{j,k} \quad \forall \mathbf{z} \in (0, 1)^2. \end{aligned}$$

Thus, for the process considered in Example 3.1, $\{X_{\mathbf{r}}^{1,d}\}$, the LWS has the form

$$S_j^l(\mathbf{z}) = \begin{cases} 1 & \text{for } j = 1, l = d, \\ 0 & \text{otherwise} \end{cases} \quad \forall \mathbf{z} \in (0, 1)^2. \quad (3.21)$$

Similarly, the process in Example 3.2 has LWS of the form:

$$S_j^l(\mathbf{z}) = \begin{cases} 1 & \text{for } j = 2, l = h, \\ 1 & \text{for } j = 2, l = v, \\ 0 & \text{otherwise} \end{cases} \quad \forall \mathbf{z} \in (0, 1)^2. \quad (3.22)$$

However, for the non-stationary process considered in Example 3.3, the LWS structure will be

$$S_j^l(\mathbf{z}) = \begin{cases} 1 & \text{for } j = 1, l = d, z \in (0, 1/2)^2 \\ 1 & \text{for } j = 2, l = d, z \in (1/2, 1) \times (0, 1/2) \\ 1 & \text{for } j = 3, l = d, z \in (0, 1/2) \times (1/2, 1) \\ 1 & \text{for } j = 4, l = d, z \in (1/2, 1)^2 \\ 0 & \text{otherwise.} \end{cases} \quad (3.23)$$

An estimate of the LWS for realisations of this process is displayed in figure 3.5. Note how this estimate is able to identify pockets of localized activity which appear at various scales within the diagonal direction. See Example 3.6 for further details.

3.5 A.C. wavelets and the LWS representation

An important issue which we consider in this section is whether the LWS are uniquely defined, given the corresponding LS2W process. To establish such a result, we must firstly introduce a wavelet based quantity, discrete autocorrelation wavelets, and define the inner product matrix of these functions. Such wavelets are related to the autocorrelation shell proposed by Saito & Beylkin (1993) and have in the past been used to obtain a measure of the covariance structure within locally stationary wavelet processes. See the review of the work of Nason *et al.* (2000) in Section 2.7 for further details.

The following definition extends autocorrelation wavelets to two-dimensions.

Definition 3.6

Let $j \in \mathbb{N}$, $l \in \{v, h, d\}$ and $\boldsymbol{\tau}, \mathbf{v} \in \mathbb{Z}^2$ where $\boldsymbol{\tau} = (\tau_1, \tau_2)$ and $\mathbf{v} = (v_1, v_2)$. Then the **autocorrelation (a.c.) wavelet**, of a two-dimensional discrete wavelet family $\{\psi_{j,\mathbf{k}}^l\}$, is given by

$$\begin{aligned}\Psi_j^l(\boldsymbol{\tau}) &= \sum_{\mathbf{v}} \psi_{j,\mathbf{v}}^l(\mathbf{0}) \psi_{j,\mathbf{v}}^l(\boldsymbol{\tau}) \\ &= \sum_u \sum_v \psi_{j,(u,v)}^l \psi_{j,(u-\tau_1, v-\tau_2)}^l.\end{aligned}\quad (3.24)$$

As one might expect, the two-dimensional a.c. wavelets inherit the separable form of the discrete wavelets (cf. Definition 3.1). Moreover, as we demonstrate in Proposition 3.2, such a.c. wavelets are symmetric in $\boldsymbol{\tau}$ and are positive semi-definite.

Proposition 3.1

Let $\boldsymbol{\tau} = (\tau_1, \tau_2) \in \mathbb{Z}^2$. Then two-dimensional discrete autocorrelation wavelets have the following separable forms:

1. In the horizontal direction:

$$\Psi_j^h(\boldsymbol{\tau}) = \Phi_j(\tau_1) \Psi_j(\tau_2). \quad (3.25)$$

2. In the vertical direction:

$$\Psi_j^v(\boldsymbol{\tau}) = \Psi_j(\tau_1) \Phi_j(\tau_2). \quad (3.26)$$

3. Finally, in the diagonal direction:

$$\Psi_j^d(\boldsymbol{\tau}) = \Psi_j(\tau_1) \Psi_j(\tau_2). \quad (3.27)$$

It can also be shown that the two-dimensional discrete autocorrelation scaling function is given by $\Phi_j(\boldsymbol{\tau}) = \Phi_j(\tau_1) \Phi_j(\tau_2)$.

Proposition 3.2

Within any decomposition direction pair (j, l) , the two-dimensional discrete autocorrelation wavelets are symmetric in the following sense:

$$\Psi_j^l(\tau_1, \tau_2) = \begin{cases} \Psi_j^l(\Leftrightarrow\tau_1, \tau_2) \\ \Psi_j^l(\tau_1, \Leftrightarrow\tau_2). \end{cases} \quad (3.28)$$

Moreover, for any set of locations $\tau_1, \tau_2, \dots, \tau_n$ and all real k_1, k_2, \dots, k_n , the autocorrelation wavelets defined in Definition 3.6 are positive semi-definite functions in the sense that

$$\sum_{p=1}^n \sum_{q=1}^n \Psi_j^l(\tau_p \Leftrightarrow \tau_q) k_p k_q \geq 0. \quad (3.29)$$

As a consequence of (3.28), it naturally follows that $\Psi_j^l(\tau_1, \tau_2) = \Psi_j^l(\Leftrightarrow\tau_1, \Leftrightarrow\tau_2)$. We now consider an example which demonstrates the form of the two-dimensional discrete Haar a.c. wavelet.

Example 3.5 *Nason et al. (2000) showed that discrete (1-D) non-decimated Haar wavelets, $\Psi_j(\tau)$, can be expressed in terms of the continuous Haar a.c. wavelets, $\Psi_H(u)$, as follows:*

$$\Psi_j(\tau) = \begin{cases} \Psi_H\left(\frac{|\tau|}{2^j}\right) & \text{for } \tau = 1 \Leftrightarrow L_j, \dots, L_j \Leftrightarrow 1, \\ 0 & \text{otherwise.} \end{cases} \quad (3.30)$$

where

$$\Psi_H(u) = \begin{cases} 1 \Leftrightarrow 3|u| & \text{for } |u| \in [0, 1/2], \\ |u| \Leftrightarrow 1 & \text{for } |u| \in (1/2, 1]. \end{cases} \quad (3.31)$$

This result also holds for all other Daubechies' wavelets. Furthermore, it can be shown that the Haar a.c. father wavelet, $\Phi_j(\tau)$, is a sampled version of the continuous Haar a.c. wavelet

$$\Phi_H(u) = \int_{-\infty}^{\infty} \phi_H(x) \phi_H(x \Leftrightarrow u) dx = \begin{cases} 1 \Leftrightarrow |u| & \text{for } |u| \in [0, 1], \\ 0 & \text{otherwise.} \end{cases} \quad (3.32)$$

Thus, utilising the separable form of a.c. wavelets, it is easily shown that discrete two-dimensional Haar a.c. wavelets, $\Psi_j^l(\boldsymbol{\tau})$, are given by:

$$\Psi_j^l(\tau_x, \tau_y) = \Psi_H^l\left(\frac{|\tau_x|}{2^j}, \frac{|\tau_y|}{2^j}\right), \quad (3.33)$$

setting $\mathbf{u} = (u_1, u_2)$, where $\Psi_H^l(\mathbf{u})$ is a two-dimensional (continuous) Haar a.c. wavelet constructed from (3.31) and (3.32) in the same way as for the discrete case in Proposition 3.1:

Vertical direction:

$$\Psi_H^v(\mathbf{u}) = \begin{cases} (1 \Leftrightarrow 3|u_1|)(1 \Leftrightarrow |u_2|) & \text{for } |u_1| \in [0, 1/2] \\ & \text{and } |u_2| \in [0, 1], \\ (|u_1| \Leftrightarrow 1)(1 \Leftrightarrow |u_2|) & \text{for } |u_1| \in [1/2, 1] \\ & \text{and } |u_2| \in [0, 1]. \end{cases}$$

Horizontal direction:

$$\Psi_H^h(\mathbf{u}) = \begin{cases} (1 \Leftrightarrow 3|u_2|)(1 \Leftrightarrow |u_1|) & \text{for } |u_2| \in [0, 1/2] \text{ and } |u_1| \in [0, 1], \\ (|u_2| \Leftrightarrow 1)(1 \Leftrightarrow |u_1|) & \text{for } |u_2| \in [1/2, 1] \text{ and } |u_1| \in [0, 1]. \end{cases}$$

Diagonal direction:

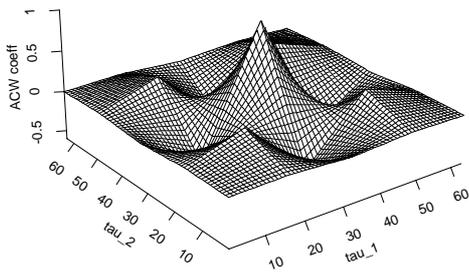
$$\Psi_H^d(\mathbf{u}) = \begin{cases} (1 \Leftrightarrow 3|u_1|)(1 \Leftrightarrow 3|u_2|) & \text{for } |u_1| \in [0, 1/2] \\ & \text{and } |u_2| \in [0, 1/2], \\ (|u_2| \Leftrightarrow 1)(1 \Leftrightarrow 3|u_1|) & \text{for } |u_1| \in [0, 1/2] \\ & \text{and } |u_2| \in [1/2, 1], \\ (|u_1| \Leftrightarrow 1)(1 \Leftrightarrow 3|u_2|) & \text{for } |u_1| \in [1/2, 1] \\ & \text{and } |u_2| \in [0, 1/2], \\ (|u_1| \Leftrightarrow 1)(|u_2| \Leftrightarrow 1) & \text{for } |u_1| \in [1/2, 1], \\ & \text{and } |u_2| \in [1/2, 1]. \end{cases}$$

Figure 3.4 below displays various examples of two-dimensional Haar autocorrelation wavelets. These and other autocorrelation wavelet families will be used in Section 3.7 to construct a measure of the autocovariance structure within LSW process.

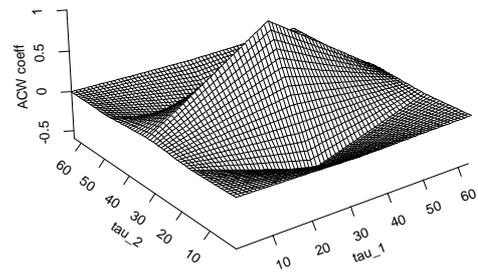
3.5.1 The inner product matrix of a.c. wavelets

The final quantity which we define is the discrete a.c. wavelet inner product matrix. Such matrices will prove useful in Section 3.6, when we consider the problem of estimating the LWS.

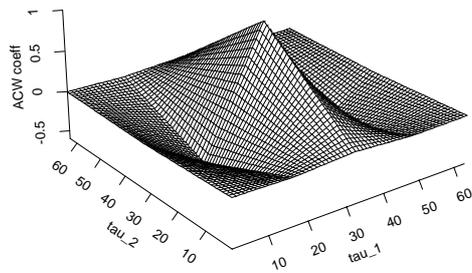
Instead of having two separate indices representing scale and direction (i.e. j and l), it is often convenient to combine both to provide a single index, η , each value of



(a)



(b)



(c)

Figure 3.4: Discrete Haar autocorrelation wavelets at scale 5 in the (a) diagonal, (b) vertical and (c) horizontal decomposition directions.

which represents a particular decomposition scale in a given direction. Explicitly, we code η as follows:

$$\eta(j, l) \equiv f(j) + g(l),$$

where

$$f(j) \equiv j \text{ and } g(l) \equiv \begin{cases} 0 & \text{when } l = v, \\ J & \text{when } l = h, \\ 2J & \text{when } l = d, \end{cases} \text{ for } j = 1, \dots, J.$$

For notational convenience, we shall henceforth represent $\eta(j, l)$ as simply η . The construction of η is such that its first J values refer to the set of scales in the vertical direction, the second set correspond to the horizontal direction whilst the final set refers to scales in the diagonal direction. The $(j, l) \Leftrightarrow \eta$ notation will be used interchangeably throughout the remainder of this chapter and Appendix A.

Further details about autocorrelation wavelets may be found in Chapter 5, where we consider the efficient construction of the inner product matrix of discrete a.c. wavelets. We define this matrix as follows:

Definition 3.7

We define the operator $A = (A_{\eta, \nu})_{\eta, \nu \geq 1}$ by

$$\begin{aligned} A_{\eta, \nu} &= \langle \Psi_\eta, \Psi_\nu \rangle \\ &= \sum_{\boldsymbol{\tau}} \Psi_\eta(\boldsymbol{\tau}) \Psi_\nu(\boldsymbol{\tau}). \end{aligned} \quad (3.34)$$

Further, we define the $3J$ dimensional matrix

$$A_J = (A_{\eta, \nu})_{\substack{\eta = 1, \dots, 3J \\ \nu = 1, \dots, 3J}} \quad (3.35)$$

where $J = \log_2(\min(R, S))$. Additionally, for each $J = J(R, S) = \log_2(\min(R, S))$ we define the vector

$$\kappa(\boldsymbol{\tau}) = \{\kappa_\eta(\boldsymbol{\tau})\}_{\eta=1, \dots, 3J}$$

via

$$\kappa(\boldsymbol{\tau}) = A_J^{-1} \Psi(\boldsymbol{\tau}) \quad (3.36)$$

where

$$\Psi(\boldsymbol{\tau}) = \{\Psi_\eta(\boldsymbol{\tau})\}_{\eta=1, \dots, 3J}.$$

A_J is formed in such a way that the rows (and columns) of A_J are ordered by scale within each decomposition direction. An equivalent construction is given by structuring the rows of A_J by direction within each decomposition scale. The computational expense of calculating either one of these matrices via a brute force approach is large. However, Eckley & Nason (2000) have shown that recursion schemes may be derived for the form defined in Definition 3.7 (see Chapter 5 for further details). These permit a more efficient construction of the matrix.

3.5.2 Uniqueness of the LWS

We are now nearly in a position to prove the uniqueness of the LWS representation, given that the underlying wavelet family is specified *a priori*. However, prior to this, we establish certain Fourier domain properties of the discrete wavelets. These will be required in the proof of the uniqueness of the LWS

Lemma 3.1

Let $\widehat{\psi}_j(\omega)$ be the Fourier transform of $\{\psi_{j,k}\}$ and $\widehat{\phi}_j(\omega)$ be the Fourier transform of $\{\phi_{j,k}\}$. Then it can be shown that

$$i) \widehat{\psi}_j(\omega) = 2^{j/2} m_1(2^{j-1}\omega) \prod_{k=0}^{j-2} m_0(2^k\omega) \quad (\text{result in Nason } et \text{ al. (2000)});$$

$$ii) \widehat{\phi}_j(\omega) = 2^{j/2} \prod_{k=0}^{j-1} m_0(2^k\omega);$$

where $m_0(\omega)$ and $m_1(\omega)$ are as defined in equations (2.12) and (2.20).

These results are simple consequences of the scaling relations between wavelets and father wavelets. However for completeness we have proved i) in the appendix.

Using the relationships detailed in Definition 3.1, the following expressions for the Fourier transforms of the $\psi_{j,\mathbf{u}}^l$ may be derived.

Corollary 3.1

Let $\boldsymbol{\omega} = (\omega_1, \omega_2)$ and

$$\widehat{\psi}_j^l(\boldsymbol{\omega}) = \int_{\mathbb{R}} \int_{\mathbb{R}} \psi_j^l(x_1, x_2) e^{i\omega_1 x_1} e^{i\omega_2 x_2} dx_1 dx_2$$

be the Fourier transform of the discrete autocorrelation wavelet at scale j in direction

l . Then, $\widehat{\psi}_j^v(\boldsymbol{\omega}) = \widehat{\psi}_j(\omega_1)\widehat{\phi}_j(\omega_2)$, $\widehat{\psi}_j^h(\boldsymbol{\omega}) = \widehat{\phi}_j(\omega_1)\widehat{\psi}_j(\omega_2)$ and $\widehat{\psi}_j^d(\boldsymbol{\omega}) = \widehat{\psi}_j(\omega_1)\widehat{\psi}_j(\omega_2)$.

The following theorem, valid for all Daubechies compactly supported wavelets, demonstrates that A is an invertible operator and that for each $J \in \mathbb{N}$, the norm of A_J^{-1} is bounded from above by a constant C_J . Moreover, the theorem establishes the uniqueness of the LWS given the corresponding LS2W process. These results are an extension of Theorem 1 from Nason *et al.* (2000). It should be noted that the first part of the proof follows directly from this earlier work. However due to the inclusion of an additional dimension, namely direction, the second part of the proof of this theorem requires careful consideration.

Theorem 3.1

For any compactly supported Daubechies wavelet, the family of discrete autocorrelation wavelets $\{\Psi_\eta\}_{\eta=1}^\infty$ is linearly independent. Hence,

- a) *the operator A is invertible (since all its eigenvalues are positive) and for each $J \in \mathbb{Z}^+$, the norm $\|A_J^{-1}\|$ is finite;*
- b) *and the LWS is uniquely defined given the corresponding LS2W process.*

3.6 Estimating the LWS

Having found a measure which provides a local direction-scale decomposition of power, it is natural to enquire how one can estimate this quantity, given the prior specification of the underlying wavelet family. The issue of what happens when one uses an alternative wavelet family to that which underlies the process is left as an avenue for future work.

Recall from stationary theory that an estimate of the spectral density function is given by the squared absolute value of the Fourier transform of a realisation of a second order stationary process. Analogously, the estimator which we propose for the LWS is founded upon the collection of squared empirical wavelet coefficients, for the coefficients which we consider here are real. Should complex wavelets be used, then the squared modulus of such coefficients should be adopted:

Definition 3.8

Let $\{X_r\}$ be a LS2W process as defined in Definition 3.2. The empirical wavelet

coefficients of the process are given by

$$d_{j,\mathbf{u}}^l \equiv \sum_{\mathbf{r}} X_{\mathbf{r}} \psi_{j,\mathbf{u}}^l(\mathbf{r}).$$

We are now in a position to define the local wavelet periodogram.

Definition 3.9

The **local wavelet periodogram** (LWP) of a LS2W process $\{X_{\mathbf{r}}\}$ is defined as

$$I_{j,\mathbf{u}}^l \equiv |d_{j,\mathbf{u}}^l|^2. \quad (3.37)$$

As we shall see in Theorem 3.2, it transpires that the LWP is a biased estimator of the LWS. However, the form of this bias suggests a transformation of the spectra which produces an asymptotically unbiased estimate of the LWS. In order that we may prove Theorems 3.2 and 3.3, we assume that the following conjecture about the operator, A , holds. Such an assumption is motivated by the work of Nason *et al.* (Theorem 2, 2000) who demonstrated that for the (one-dimensional) inner product matrix the conjecture holds for Haar and Shannon wavelets. Furthermore, they were able to provide strong evidence to suggest that the result holds for all Daubechies wavelets which, loosely speaking, lie between these two extremes.

Conjecture 3.1

For all Daubechies' compactly supported wavelets the minimum eigenvalue of the operator A , $\lambda_{\min}(A)$, is positive. Therefore $\|A^{-1}\| < \infty$, A is positive definite and has a bounded inverse.

It is important to note, however, that Theorem 3.1 guarantees that A_J is invertible and that the norm of A_J^{-1} is bounded.

We are now in a position to consider the expectation of the local wavelet periodogram.

Theorem 3.2

Let $\mathbf{z} = (z_1, z_2)$, $\mathbf{R} = (R, S)$ and $[\mathbf{zR}] = ([z_1R], [z_2S])$ where $R = 2^J$, $S = 2^K$ for some $J, K \in \mathbb{N}$. Further, assume that the $\{\xi_{\eta,\mathbf{r}}\}$ are Gaussian, then:

$$\mathbb{E}(I_{\eta,[\mathbf{zR}]}) = \sum_{\eta_1} A_{\eta\eta_1} S_{\eta_1}(\mathbf{z}) + O\left(\frac{1}{\min\{R, S\}}\right). \quad (3.38)$$

Thus the LWP estimate of the LWS at a given (j, l) -pair is a weighted sum of all LWS.

An example of this bias (i.e the right hand side of equation (3.38) minus $S_\eta(\mathbf{z})$) can be seen in Table 3.1, below. The table displays a selection of coefficients from A , highlighting the corruption which occurs within the estimate of the third finest scale LWS in the vertical decomposition direction, $I_{3,[\mathbf{z},\mathbf{R}]}$. Observe how the estimator is a mix of contributions from various directions and scales. Note in particular that power leaks across from the fourth and fifth scales within the diagonal decomposition direction.

Direction	Scale 1	Scale 2	Scale 3	Scale 4	Scale 5
Vertical	0.703	3.797	15.453	13.793	7.573
Horizontal	0.203	0.797	1.891	2.793	2.073
Diagonal	0.047	0.422	3.953	8.379	6.220

Table 3.1: Biased contributions, A_{η,η_1} , which enter into the LWP estimate of $S_{\eta=3}(\mathbf{z}) = S_3^v(\mathbf{z})$. The displayed coefficients are correct to three decimal places.

Clearly, without correction, the redundancy of the NDWT induces a spread of power into other directions and scales. However, if we denote the vector of periodograms, $\mathbf{I}(\mathbf{z}) = \{I_{\eta,[\mathbf{z},\mathbf{R}]}\}$, and define the vector of corrected LWPs to be given by $\mathbf{L}(\mathbf{z}) = A^{-1}\mathbf{L}(\mathbf{z})$, then we obtain an asymptotically unbiased estimator of the LWS:

$$\mathbb{E}(\mathbf{L}(\mathbf{z})) = \mathbf{S}(\mathbf{z}) + O\left(\frac{1}{\min\{R, S\}}\right). \quad (3.39)$$

This is a straightforward consequence of Proposition 4 of Nason *et al.* (2000).

The following definition will prove useful when considering the covariance structure of the wavelet periodogram.

Definition 3.10

Define

$$\alpha_{j_1, j_2}^{l_1, l_2}(\mathbf{u}_1, \mathbf{u}_2) = \sum_r \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}) \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{r}). \quad (3.40)$$

In effect, this is a form of ‘‘cross-correlation’’ between two wavelets of the same family at (possibly) different scales and directions, centred on different locations. Using this identity, we can explore the covariance structure of the (uncorrected) LWP.

Theorem 3.3

Assume that the $\{\xi_{\eta,\mathbf{r}}\}$ are again Gaussian. Then the covariance between $I_{j_1,\mathbf{p}}^{l_1}$ and $I_{j_2,\mathbf{q}}^{l_2}$ may be expressed as follows:

$$\text{Cov}(I_{j_1,\mathbf{p}}^{l_1}, I_{j_2,\mathbf{q}}^{l_2}) = 2 \left\{ \sum_{l_0} \sum_{j_0} \sum_{\mathbf{u}_0} (w_{j_0,\mathbf{u}_0}^{l_0})^2 \alpha_{j_1,j_0}^{l_1,l_0}(\mathbf{p}, \mathbf{u}_0) \alpha_{j_2,j_0}^{l_2,l_0}(\mathbf{q}, \mathbf{u}_0) \right\}^2.$$

Thus the correlation between these quantities decreases with increasing distance between location \mathbf{p} at scale-direction (j_1, l_1) and the location \mathbf{q} at (j_2, l_2) . In particular, when $j_1 = j_2$, the covariance is zero when $\|\mathbf{p} \Leftrightarrow \mathbf{q}\|$ exceeds the overlap of the corresponding wavelets support. Moreover

$$\begin{aligned} \text{Var}(I_{j,\mathbf{p}}^l) &= 2\mathbb{E}(I_{j,\mathbf{p}}^l)^2; \\ &= 2 \left(\sum_{\eta_1} A_{\eta_1} S_{\eta_1}([\mathbf{p}/\mathbf{R}]) \right)^2 + O\left(\frac{2^{j(\eta)}}{\min(R, S)}\right), \end{aligned} \quad (3.41)$$

where $j(\eta) \equiv \eta \Leftrightarrow \lfloor \frac{\eta-1}{J} \rfloor J$ simply denotes the scale element of $\eta(j, l)$.

The above demonstrates that the uncorrected LWP have asymptotically non-vanishing variance. Hence, by construction, the *corrected* LWP will also have an asymptotically non-vanishing variance, thus paralleling the traditional stationary case. Consequently, estimates of the LWS need to be smoothed to obtain consistency. See Section 3.6.1 for details of a wavelet-based scheme.

Example 3.6 Recall the non-stationary LS2W process which was considered in Example 3.3. This process was constructed in such a way that contributions to the covariance structure were only made in certain regions of the diagonal decomposition direction. Figure 3.5 displays a selection of the mean (corrected) LWP of 100 realisations of this process. Note how the scale 2 diagonal contribution is well localised as is the contribution at scale 4 in the diagonal direction. Some structure does however appear in the horizontal and vertical directions, primarily on the boundaries of the various covariance regions, see for example the plot of the scale 3 horizontal LWS estimate in figure 3.5.

To highlight the pitfalls of neglecting to correct by the inner product matrix, we include a plot of the mean (uncorrected) LWP of the same 100 realizations of this process for scales 1 to 4 (see figure 3.6). Note how power has seemingly increased in

both scales 2 and 4, even in the vertical direction which contributed no structure within the model construction.

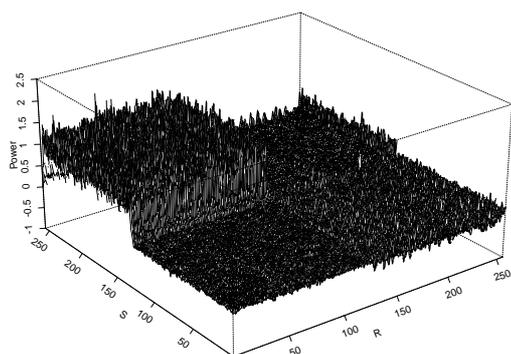
The corrected LWP will be used later, in Chapter 4, when we consider the application of this measure to various texture analysis problems.

3.6.1 Smoothing the local wavelet periodogram

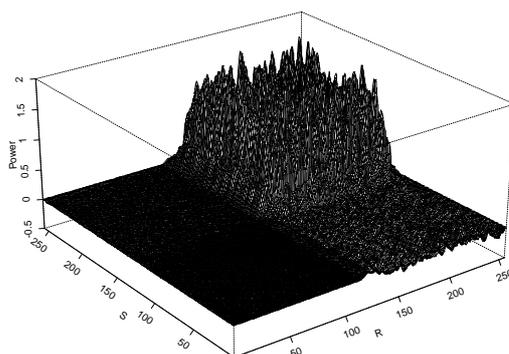
In a further parallel to the traditional stationary approach, Theorem 3.3 demonstrates that the LWP is not a consistent estimator of the LWS (see Priestley (pp. 301–302, 1981) for a brief summary of consistency). To attain a consistent estimate of the LWS we must therefore smooth the LWP. Several smoothing approaches could be used in this instance, for example kernel smoothing or a moving average approach. However bearing in mind that many images, including the textured images of Chapter 4, are characterised by edges, it would appear prudent to use a smoothing scheme which has the ability to deal efficiently with such features. It is for this reason that we propose to smooth the LWP, $I_{\eta, \mathbf{z}}$, as a function of \mathbf{z} using either DWT shrinkage, paralleling the approach proposed by von Sachs, Nason & Kroisandt (1997), or the translation invariant denoising approach of Coifman & Donoho (1995).

Assuming that the innovations $\{\xi_{j, \mathbf{u}}^l\}$ are Gaussian it follows that, upon squaring, each element of the wavelet periodogram has a χ^2 -distribution. Correcting, to obtain an asymptotically unbiased estimate of the LWS (as suggested by Theorem 3.2), leads to a complex correlated distribution for the LWP. Thus, we follow Nason *et al.* (2000) and suggest firstly performing wavelet shrinkage of the χ^2 -distributed periodogram prior to correction by A^{-1} . A detailed description of how one may smooth using an orthonormal second-stage wavelet basis $\{\tilde{\psi}_{l, m}\}$ is provided by von Sachs *et al.* (1997). Briefly, smoothing is performed by implementing a non-linear thresholding of the raw (uncorrected) periodogram, $I_{\eta}(\mathbf{z})$, and then inverting the smoothed transformation to obtain the estimate $\tilde{I}_{\eta}(\mathbf{z})$.

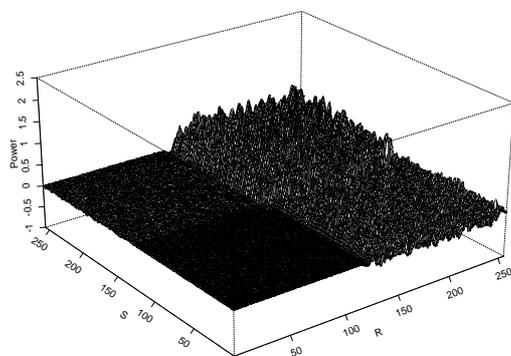
Whilst wavelets are well-suited to representing point-like discontinuities, a potential problem exists when smoothing images with line or curve-like edges. Often, the wavelet transform of such images contains several large wavelet coefficients along the entire length of an edge. Furthermore, as Starck, Candès & Donoho (2000) observe, “in a



Scale 2, Diagonal direction

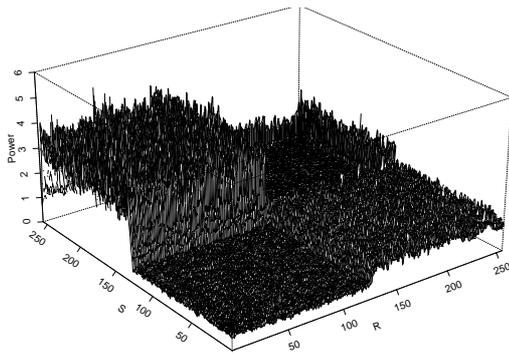


Scale 4, Diagonal direction

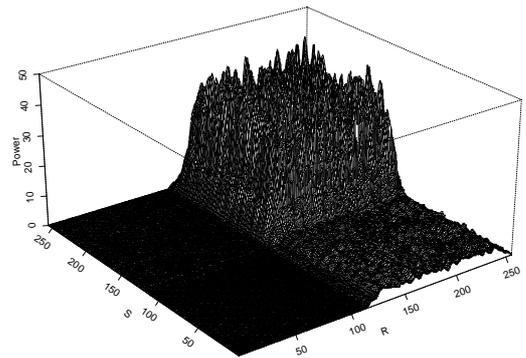


Scale 3, Horizontal direction

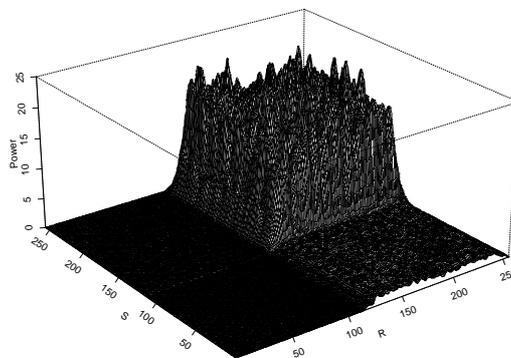
Figure 3.5: Mean (corrected) local wavelet spectra of 100 realisations of the non-stationary process displayed in figure 3.3.



Scale 2, Diagonal direction



Scale 4, Diagonal direction



Scale 4, Vertical direction

Figure 3.6: Mean (uncorrected) local wavelet spectra of 100 realisations of the non-stationary process displayed in figure 3.3.

map of the large wavelet coefficients [of an image] one sees the edges of the images repeated scale after scale". Thus many coefficients are required to reconstruct the edges in an image.

In recent years, Candès and Donoho have proposed two new expansion forms: namely Ridgelets and Curvelets (see, for example, Candès (1999), Candès & Donoho (1999a, b)). A ridgelet expansion of an image results in a sparse representation of both smooth functions and perfectly straight edges, whilst curvelets provide an efficient representation of smooth curve discontinuities. Recent empirical investigations by Starck *et al.* (2000) and Do & Vetterli (2000, 2001) indicate that these new transforms outperform traditional wavelet methods in the denoising of images with smooth curve discontinuities. It would therefore appear that, for certain LWS forms, there is strong motivation for considering curvelet-based smoothing techniques. This is left as an avenue for future research.

3.7 Capturing the autocovariance structure

In Sections 3.4 – 3.6 we proposed and studied various properties of the local wavelet spectrum, a quantity which measures the *local* power within a LS2W process at a given scale and direction. From traditional models of processes which lie on a regular grid, it is known that the autocovariance of a (second order) stationary process, $\{X_{\mathbf{u}}\}$, may be represented by the following:

$$C_X(\boldsymbol{\tau}) = \int f(\boldsymbol{\omega}) e^{i\boldsymbol{\omega}\boldsymbol{\tau}} d\boldsymbol{\omega}, \quad (3.42)$$

where $f(\boldsymbol{\omega})$ is the spectrum of the process. Thus it seems natural to enquire whether an equivalent relationship exists between the LWS and a measure which captures the local autocovariance structure of a LS2W process.

The measure which we propose below is based on a time series analogue introduced by Nason *et al.* (2000). The measure is constructed using the autocorrelation wavelets which we introduced in Section 3.5. The remainder of the section highlights various theoretical properties of the local autocovariance measure.

3.7.1 Local autocovariance

Definition 3.11

The **local autocovariance** (LACV), $C(\mathbf{z}, \boldsymbol{\tau})$, of a given LS2W process with LWS $\{S_j^l(\mathbf{z})\}$ is defined to be

$$C(\mathbf{z}, \boldsymbol{\tau}) = \sum_l \sum_{j=1}^{\infty} S_j^l(\mathbf{z}) \Psi_j^l(\boldsymbol{\tau}) \quad (3.43)$$

where $\boldsymbol{\tau} \in \mathbb{Z}^2$ and $\mathbf{z} \in (0, 1)^2$.

By Proposition 3.2, it follows that the local autocovariance $C(\mathbf{z}, \boldsymbol{\tau})$ is symmetric in $\boldsymbol{\tau}$. Frequently, the LACV provides an efficient representation of the second order structure within a process. The following example demonstrates the sparsity with which Haar MA fields may be represented by this measure.

Example 3.7 The stationary process $X_{\mathbf{r}}^{1,d}$ given in Example 3.1 has autocovariance

$$\begin{aligned} C_{X^1}(\tau_1, \tau_2) = \sigma^2 & \left[\delta_{\tau_1,0} \delta_{\tau_2,0} \Leftrightarrow \frac{1}{2} \delta_{\tau_1,0} \delta_{\tau_2,1} \Leftrightarrow \frac{1}{2} \delta_{\tau_1,1} \delta_{\tau_2,0} \right. \\ & \Leftrightarrow \frac{1}{2} \delta_{\tau_1,0} \delta_{\tau_2,-1} \Leftrightarrow \frac{1}{2} \delta_{\tau_1,-1} \delta_{\tau_2,0} + \frac{1}{4} \delta_{\tau_1,1} \delta_{\tau_2,1} \\ & \left. + \frac{1}{4} \delta_{\tau_1,-1} \delta_{\tau_2,-1} + \frac{1}{4} \delta_{\tau_1,-1} \delta_{\tau_2,1} + \frac{1}{4} \delta_{\tau_1,0} \delta_{\tau_2,1} \right] \end{aligned}$$

This is precisely the autocorrelation wavelet $\Psi_1^d(\tau_1, \tau_2)$. Thus $X^1(r, s)$ has an extremely sparse representation in terms of the autocorrelation wavelets:

$$\frac{1}{\sigma^2} C_{X^1}(\tau_1, \tau_2) = \Psi_1^d(\tau_1, \tau_2);$$

i.e. equation (3.43) with $S_j^l(\mathbf{z}) = 1$ for $(j = 1, l = d)$ and zero otherwise. Furthermore, as one would expect for a stationary process, there is no dependence on \mathbf{z} in this expression.

Similar sparse representations occur if we replace Haar wavelets by other compactly supported, two dimensional wavelets and concentrate only on one or two decomposition aspects, e.g. $W_j^l(\mathbf{z}) = 1$ for $j = 1, l = d$ and zero elsewhere.

Whilst the above example illustrates the form of the local autocovariance for a simple stationary case, Definition 3.11 refers to the more general LS2W case. The following proposition demonstrates the manner in which the autocovariance of a LS2W process, $C_{\mathbf{R}}$, tends asymptotically to the defined $C(\mathbf{z}, \boldsymbol{\tau})$ which has the autocorrelation

wavelet representation given by (3.43). This result extends that of Nason *et al.* (Proposition 1, 2000).

Proposition 3.3

Suppose we have a LS2W process as defined in Definition 3.2. Then as $R, S \rightarrow \infty$,

$$|C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| = O\left(\frac{1}{\min\{R, S\}}\right) \quad (3.44)$$

uniformly in $\boldsymbol{\tau} \in \mathbb{Z} \times \mathbb{Z}$ and $\mathbf{z} = (z_R, z_S) \in (0, 1)^2$.

Note If the process $\{X_{\mathbf{r}; \mathbf{R}}\}$ is *stationary*, then the dependence of S_j^l on \mathbf{z} disappears. In other words for a stationary process, S_j^l is constant over \mathbf{z} for any direction/scale pair (l, j) . This, in turn, implies that the dependence of C on \mathbf{z} disappears when the process is stationary. Thus, for stationary processes, the representation in equation (3.6) changes from being local to global. The shift-equivariance of the non-decimated wavelet transform is able to cope with this change in the nature of the representation.

3.7.2 Invertibility of the autocovariance representation

In the classical analysis of stationary processes which lie on a regular grid, the representation given by equation (3.42) for the autocovariance is invertible. More precisely, the spectrum is also the Fourier transform of the autocovariance. Thus it is natural to ask whether the LWS can be considered to be an “inverse transform” of the LACV? As a consequence of Theorem 3.1 we are able to prove the following straightforward extension of Nason *et al.* (Proposition 2, 2000), demonstrating that the LWS can indeed be considered to be such an inverse transform.

Lemma 3.2

Assume that Conjecture 3.1 holds. Then the inverse formula of

$$C(\mathbf{z}, \boldsymbol{\tau}) = \sum_{\eta} S_{\eta}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau})$$

is given by

$$S_j^l(\mathbf{z}) = \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\boldsymbol{\tau}} C(\mathbf{z}, \boldsymbol{\tau}) \Psi_{\eta_1}(\boldsymbol{\tau}). \quad (3.45)$$

It is important to understand that even though for each finite R and S , the process representation in (3.6) cannot be unique, the representation of the LACV given by (3.43), as an asymptotically defined quantity *is* unique.

The final result which we consider extends Proposition 3 of Nason *et al.* (2000) to two-dimensions, focusing on the ability of LS2W processes to represent second-order stationary processes:

Proposition 3.4

Assume that Conjecture 3.1 holds. Then for any family of discrete Daubechies' wavelets, $\{\psi_j\}$:

- a) *any zero mean, second-order stationary process, $\{X_{\mathbf{r}}\}$, with absolutely summable autocovariance $\sum_{\boldsymbol{\tau}} |C_X(\boldsymbol{\tau})|$ is a LS2W processes with LWS, $\{S_j^l(\mathbf{z})\}$.*
- b) *Moreover, any LS2W process $\{X_{r,s}\}$ with location-independent LWS, $\{S_j^l(\mathbf{z})\}$, fulfilling the additional assumption*

$$\sum_{l_j} 2^{2j} S_j^l(\mathbf{z}) < \infty$$

is a stationary process with absolutely summable autocovariance $C_X(\boldsymbol{\tau})$.

3.8 Summary

In this chapter, we have introduced a wavelet-based model of locally stationary random fields which lie on a regular grid. The LS2W model provides a local decomposition of power into various direction-scale pairs. To quantify the local power within LS2W processes we introduced the local wavelet spectrum (LWS), a spatially localised wavelet analogue of the Fourier-based spectrum which is used with second-order stationary processes. We have also been able to prove that given the corresponding LS2W process, the LWS is uniquely defined.

An estimator of the LWS was proposed in Section 3.6. Although this initial estimator was shown to be biased (an artefact of the inherent redundancy of the non-decimated wavelet transform), it gave rise to an asymptotically unbiased estimator after correction. This estimator will be used in Chapter 4, where we consider the application of this modelling approach to texture analysis. Finally, in Section 3.7, we introduced

a measure of the local autocovariance (LACV) structure within LS2W processes and were able to demonstrate that the LWS are a form of “inverse transform” of the LACV.

We have not considered the problem of estimating the LACV in this chapter. Conceptually, the estimation of this quantity seems simple:

Suppose we observe a process on a grid of dimension $R \times S$, where $R = 2^J$ and $S = 2^K$ for some $J, K \in \mathbb{N}$. Naïvely, the LACV may be estimated using

$$\hat{C}_{J_0}(\mathbf{z}, \boldsymbol{\tau}) = \sum_l \sum_{j=1}^{J_0} I_j^l([\mathbf{z}\mathbf{R}])\Psi_j(\boldsymbol{\tau}), \quad (3.46)$$

where $J_0 = \log_2(\min\{R, S\})$.

This measure is similar to that proposed by Nason *et al.* (2000) in the context of time series analysis. To date, neither estimator has been studied in detail. In Chapter 6 we consider the properties of the one-dimensional version of this estimator in the hope that we can relate this quantity to its stationary counterpart and, for example, consider issues of bias.

A module of routines for the (unbiased) estimation of the LWS has been implemented in S-Plus as an add-on to `WaveThresh`. A summary of the suite’s key routines may be found in Appendix C.

Chapter 4

Texture analysis

4.1 Introduction

In this chapter, we consider the application of the LS2W modelling approach to gray-scale texture analysis, the issue of analysing coloured images being left as an avenue for future research. Broadly speaking, texture is the visual character of an image region whose structure is, in some sense, regular: for example the appearance of a woven material. The human visual system is well adapted to the recognition of texture, to the extent that upon looking at figure 4.1(a) for example, we can easily identify that it is an image of tree bark.

Texture frequently possesses structure on many different scales. Thus, when modelling the structure of a textured image, an attempt should be made to incorporate this multiscale reality. A model, such as that afforded by the LS2W approach, which provides a multiscale decomposition of the covariance structure of a textured image would therefore appear desirable.

The content of this chapter is divided into two halves. The first half reviews the “statistical” approach to texture analysis: in Section 4.2, we consider the definition of texture and some of its many properties, together with a brief summary of various texture-based applications. A review of several traditional approaches to texture analysis is provided in Section 4.3, whilst recent wavelet-based approaches are considered in Section 4.4. The second half of the chapter focuses on the application of our LS2W approach to texture analysis. Section 4.5 provides details of the approach which we adopt to various texture discrimination and classification problems, together

with an example based on images taken from the standard Brodatz (1966) collection. Finally in Section 4.6 we investigate the potential of our approach to various texture analysis problems encountered by an industrial collaborator, comparing our results with Fourier and various multiscale-based texture measures.

4.2 What is texture?

The Oxford English Dictionary (Second Edition) provides the following entry under texture:

“texture n. [ad. L. *textūra* a weaving see TEXT n. and -URE. So F. *texture* (16th c. in Godef. *Compl.*.)] 1. a. The process or art of weaving. *Obs.* ... 2. b. *transf.* Any natural structure having an appearance or consistence as if woven; ... 3. The character of a textile fabric, as to its being fine, coarse, close, loose, plain, twilled, ribbed, diapered, etc., resulting from the way in which it is woven. ... 4. In extended use: The constitution, structure, or substance of anything with regard to its constituents or formative elements. a. Of organic bodies and their parts ... b. Of inorganic substances, as stones, soil etc.: Physical (not chemical) constitution; the structure or minute moulding (of a substance).”

It is a term which we use frequently in our everyday lives and, invariably, is used in the context of a physical quality of an object which appeals greatly to our senses, particularly vision and touch. Yet texture has no precise definition. Indeed, such is the difficulty in defining texture that Coggins (1982) was able to compile a collection of different definitions of texture from the computer vision literature (see Tuceryan & Jain (1999) for examples).

Haralick, Shanmugam & Dinstein (1973) write that texture “is an innate property of virtually all surfaces ... It contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment.” In what follows, we consider texture to be the collection of properties which represent the surface structure of an object. Figure 4.1 provides examples of textured images taken from the Brodatz (1966) image library — a collection of images which has become a

standard test set in the literature. Typically, textured images are not uniform. For example, the bark image (figure 4.1(a)) contains variations within its pixel intensities which form a pattern which we identify as being a visual texture. These patterns are induced by physical properties, for example roughness and edges, and reflective differences from the colour of the surface.

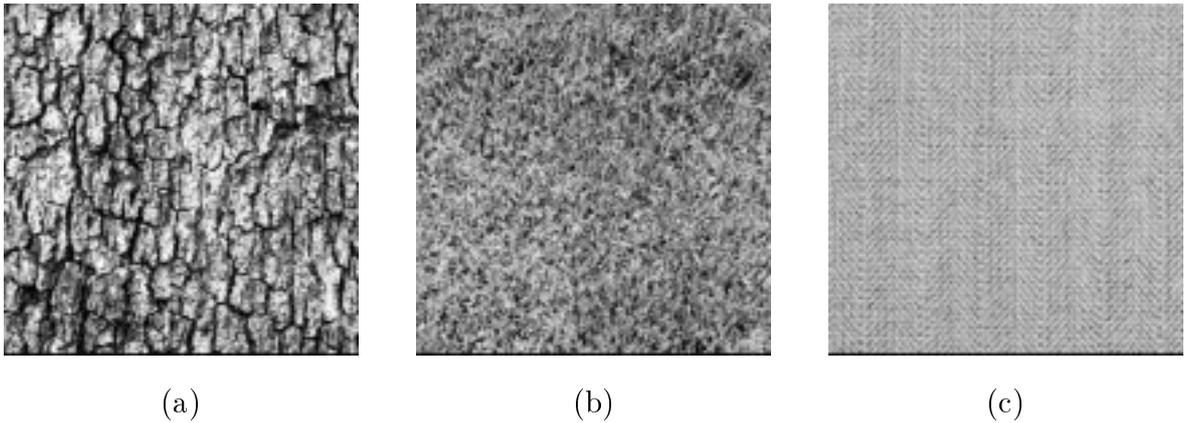


Figure 4.1: Sample images of (a) Bark, (b) Grass and (c) Herringbone Weave taken from the Brodatz (1966) image library. Images obtained from the USC-SIPI Image Database (<http://sipi.usc.edu/services/database/Database.html>).

A more refined view of texture is provided by Haralick (1979), who argues that texture can be considered as

“an organized area phenomena. When it is decomposable, it has two basic dimensions ... The first dimension is for describing primitives out of which the image is composed, and the second dimension is for the description of the spatial dependence or interaction between the primitives of an image texture.”

A texture primitive is simply a collection of pixels which form a basic element of a textured image, for example a blade of grass in figure 4.1(b). Thus, in light of Haralick’s comment, it is evident that some form of scale dependency is required for texture description. For example, in the herringbone weave of figure 4.1(c), there are potentially two primitive levels: the first (coarsest) level corresponding to the “bone” structure within the image, whilst the finer texture of the material strands constitutes an alternative level.

The development and application of techniques which harness the information contained within textured images are studied in texture analysis. Broadly speaking, the research in this field may be divided into two categories; namely the development of feature extraction methods and the development applications which use these extracted features.

Extracted features are frequently applied in one of the following activities:

Texture Discrimination: Discrimination of textured images of *known* classes based on extracted features. By discriminating effectively and understanding the features used, one can begin to appreciate the differences between the various classes.

Texture Classification: Similar to discrimination. The task is to determine to which of a finite number of classes a textured image I (of *unknown* class) belongs. Consequently, prior knowledge of the available classes is required, together with an effective texture measure.

Texture Segmentation: Involves the decomposition of a (possibly) multi-textured region (for example figure 4.2) into its constituent elements and identifying the boundaries between the various regions. Frequently, the number and type of textures involved are unknown, making this problem particularly challenging.

Texture Synthesis: This consists of the synthetic generation of a texture based on statistical parameters. See Wechsler (1980) or Iversen & Lønnestad (1992) for further details.

These activities have been used in several fields, including automated inspection (Jain, Farrokhnia & Alman (1990)), medical image analysis (Lundervold (1992)), remote sensing (Haralick *et al.* (1973), Lee & Philpot (1991)) and document processing (Jain & Farrokhnia (1991)) to name but a few. Tuceryan & Jain (1999) provide a stimulating review of many other applications.

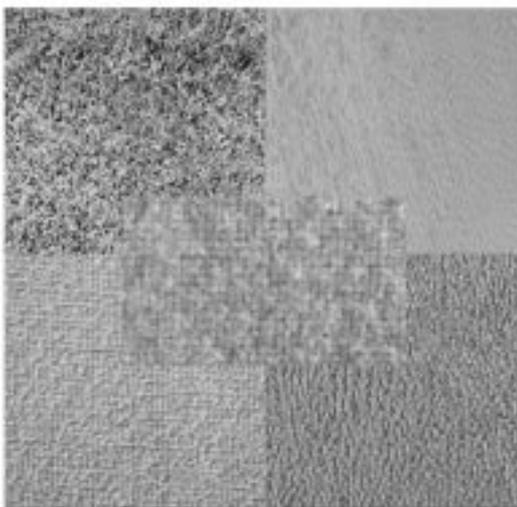


Figure 4.2: An example of a multi-textured region (images obtained from the Brodatz collection of the USC-SIPI Image Database).

Feature extraction techniques developed to date have largely fallen into one of two categories. The most commonly used approach is statistical and consists of computing different measures which describe a texture in a form suitable for statistical pattern recognition. A summary of several such techniques is provided below. In Section 4.3 we review some of the more established feature extraction techniques prior to considering several recently proposed wavelet-based approaches in Section 4.4. It should be noted that there exists an alternative approach to texture analysis, termed “syntactic”. This approach is based upon a grammatic or symbolic representation of the primitive relationship structure within a texture (see Haralick (1979), Wechsler (1980) or Sonka, Hlavac & Boyle (1999) for further details).

4.3 Statistical approaches to texture description

Below, we summarise some of the more established statistical techniques used in the literature. Although the approach is termed statistical, the issue of obtaining suitable measures from textured images has not received much attention in the statistics literature. Comprehensive reviews of the statistical approach to texture analysis are provided by Haralick (1979, 1986), Tuceryan & Jain (1999) and Tomita and Tsuji (Chapter 2, 1990). Reed & du Buf (1992) review feature extraction techniques for

unsupervised applications (i.e. segmentation) whilst Randen & Husøy (1999) provide a comparative review of various filtering-based approaches to feature extraction.

4.3.1 Fourier-based approaches

Perhaps the most familiar statistical technique used in the field of texture analysis is that which is based upon the autocorrelation function,

$$\rho(r, s) = \frac{RS}{(R \Leftrightarrow r)(S \Leftrightarrow s)} \frac{\sum_{i=1}^{R-r} \sum_{j=1}^{S-s} x_{i,j} x_{i+r,j+s}}{\sum_{i=1}^R \sum_{j=1}^S x_{i,j}^2},$$

for $r, s \in \{1, \dots, R\} \times \{1, \dots, S\}$. Here r and s denote the lag difference in the i, j directions respectively whilst R and S denote the dimensions of the image.

Haralick (1979) draws attention to an experiment conducted by Kaizer (1955). The aim of this experiment was to establish whether there was any form of relationship between the texture which photo-interpreters view in an image and the autocorrelation function of that image. Using several aerial photographs of an Arctic region (see for example figure 1 of Haralick (1979)) Kaizer assumed isotropy and calculated estimates of the autocorrelation functions. The results of Kaizer's research are discussed by Haralick (1979, 1986). What is particularly interesting is Kaizer's discovery that a relatively flat background, indicative of a fine texture, was interpreted by subjects as being fine *or* coarsely textured. This highlights a fundamental characteristic of texture: for a given smooth surface there exists a scale such that when the image is examined at that scale, it has no texture. As resolution increases, the grading of that texture moves gradually to a fine texture and then a coarse texture. Thus scale, and the appreciation of the scale in question, is an important feature of texture.

Another, related quantity, used for texture feature extraction is the spectrum of an image (i.e. the squared modulus of the image's Fourier transform). This provides a frequency descriptor of the information contained within an image. Average values of energy within ring or wedge functions of frequency (see figure 4.3) can be used to provide features relating to coarseness and directionality respectively. For example a high energy in the low frequencies is characteristic of a coarse texture, whilst a large contribution in the high frequencies is indicative of a fine texture (see, for example, Weszka, Dyer & Rosenfeld (1976) for further details).

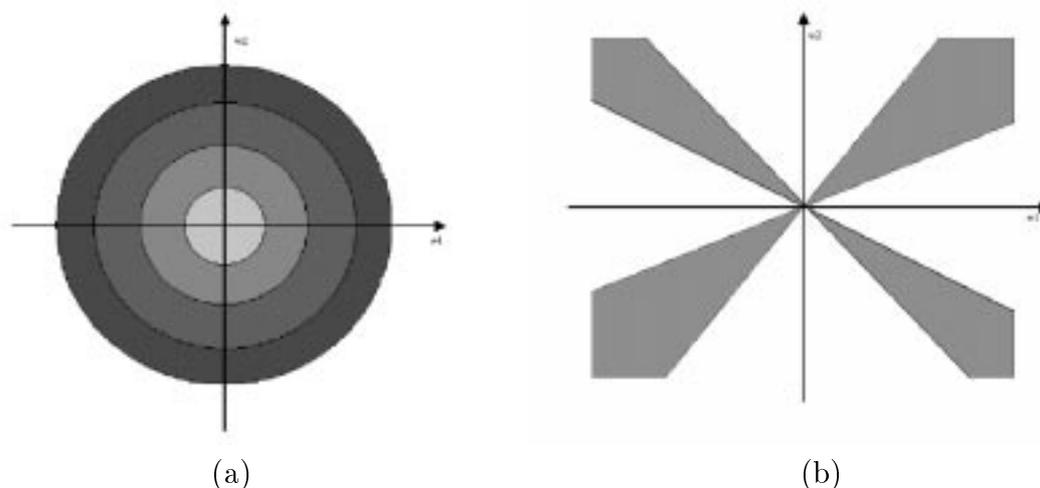


Figure 4.3: Frequency-based texture statistics: measures can be formed by evaluating average energy values in (a) frequency rings or (b) wedges.

It has been demonstrated that features based on the Fourier power spectrum are outperformed by measures based on gray level co-occurrence statistics, which we describe in Section 4.3.3. Statistics based on spatial gray-level differences have also been shown to outperform the Fourier approach (Weszka *et al.* (1976) and Connors & Harlow (1980)).

4.3.2 Textural edgeness

An alternative approach to quantifying texture by spatial frequency is to consider texture in terms of edgeness per unit area (Davies & Mitchie (1980)). A microedge may be detected by comparing the (local) values of properties obtained from pairs of non-overlapping neighbourhoods bordering the resolution cell. Conversely, a macro edge may be detected using large neighbourhoods. Sutton & Hall (1972) propose an alternative approach, using the gradient as a function of distance between pixels. Tomita & Tsuji (1990) propose several other texture properties which may be derived from statistics based upon edge distributions. These include coarseness, contrast and directivity.

4.3.3 Gray tone co-occurrence matrices

Haralick *et al.* (1973) present a general procedure for extracting textural properties based upon the co-occurrence matrix of an image (defined below). This approach is founded upon the use of the repeated occurrence of a gray level configuration within an image as a texture descriptor: a configuration varies rapidly with distance within a fine texture but slowly with coarse textures. More precisely, the method is formulated as follows:

Consider a region of interest, \mathbf{R} , of dimension $M \times N$ within an image. Further, let $P_{d,\phi}(a, b)$ be a matrix which measures the number of occurrences with which two pixels, of gray levels a and b respectively, appear in \mathbf{R} separated by a distance d in direction ϕ . Haralick *et al.* (1973) argue that $P_{d,\phi}(a, b)$ describes an occurrence of some gray-level configuration. Various measures such as energy, entropy, contrast and correlation may be derived from the co-occurrence matrices, these features subsequently being used for texture classification etc. Such matrices may also be defined to be symmetric (see Sonka *et al.* (Section 14.1.2, 1999)). Although co-occurrence matrices provide good results in practice, the approach is computationally expensive.

4.3.4 Other statistical approaches

A popular feature extractor, proposed by Laws (1979), assesses average gray-level, edges, spots, ripples and waves. The measures are constructed from three simple vectors:

$$\begin{aligned}
 L_3 &= [1, 2, 1] && \text{a measure of average;} \\
 E_3 &= [\leftrightarrow 1, 0, 1] && \text{which calculates first differences (i.e. edges);} \\
 \text{and } S_3 &= [\leftrightarrow 1, 2, \leftrightarrow 1] && \text{which calculates spots.}
 \end{aligned}$$

Taking suitable convolutions of these vectors with themselves and each other results in five vectors:

$$\left. \begin{aligned} L_5 &= [1, 4, 6, 4, 1], \\ E_5 &= [\leftrightarrow 1, \leftrightarrow 2, 0, 2, 1], \\ S_5 &= [\leftrightarrow 1, 0, 2, 0, \leftrightarrow 1], \\ R_5 &= [1, \leftrightarrow 4, 6, \leftrightarrow 4, 1] \\ \text{and } W_5 &= [\leftrightarrow 1, 2, 0, \leftrightarrow 2, \leftrightarrow 1]. \end{aligned} \right\} \quad (4.1)$$

The tensor products of the vectors displayed in equation (4.1) produce the two-dimensional filters which are commonly referred to as *Laws' masks*. For example, one such mask is given by:

$$L_5 \otimes E_5 = \begin{bmatrix} \leftrightarrow 1 & \leftrightarrow 2 & 0 & 2 & 1 \\ \leftrightarrow 4 & \leftrightarrow 8 & 0 & 8 & 4 \\ \leftrightarrow 6 & \leftrightarrow 12 & 0 & 12 & 6 \\ \leftrightarrow 4 & \leftrightarrow 8 & 0 & 8 & 4 \\ \leftrightarrow 1 & \leftrightarrow 2 & 0 & 2 & 1 \end{bmatrix}. \quad (4.2)$$

These filters can be used to formulate feature vectors of textured images.

The filters proposed by Laws are an example of a local linear transform (LLT). Other examples of such transforms include the discrete sine, cosine and Hadamard transforms. With several potential LLTs available for any given problem, Unser (1986) considers the problem of transform selection for a given application: the “optimal” transform being chosen either by minimising an entropy criterion or maximising an energy criterion such that the transform produces uncorrelated variables. Unser (1986) demonstrates that the “optimal” transform (with respect to the energy criterion) when trying to find a representation which best describes the local texture properties of an image, is the Karhunen-Loève transform. However, as Unser & Eden (1989) note, in practice it is often preferable to use sub-optimal linear transforms as optimal transforms are texture dependent.

In an experiment to determine how the LLT approach compares with the co-occurrence matrix method, Unser (1986) concludes that it is “almost as efficient”: i.e. the classification rates are comparable. Moreover, due to the computational efficiency of LLTs and their connection to psychological findings about the visual system, Unser

(1986) argues that they may be preferable to the co-occurrence approach. Indeed, Unser (1995) states that the LLT can “be used as the reference method for a single resolution analysis”.

Many other statistical approaches have been used by texture analysts, for example:

1. Autoregressive models (Deguchi & Morishita, 1978)
2. Markov chain models (Cross & Jain, 1983; Qian & Titterington, 1991))
3. Mathematical morphology (Serra & Verchery, 1973)
4. Fractal based methods (Pentland, 1984)
5. The texture transform (Simaan, 1990)
6. Gray level run length (Galloway, 1975)

See Haralick (1986) or Sonka *et al.* (1999) for further details.

None of the above feature extraction techniques can be described as providing a multiscale analysis of the structure within a textured image. However, as we noted in Section 4.2, scale-structure is an inherent feature of many textured images. Thus, when trying to differentiate or classify between various textured images, the inclusion of scale information seems highly desirable. In an attempt to harness and quantify such scale structure, recent research has focused upon the use of multiscale techniques for texture analysis. We proceed to consider such methods in the next section.

4.4 Wavelets for texture analysis

4.4.1 An introduction to wavelets for texture analysis

Recent psycho-visual research has indicated that the human and mammalian visual systems process images in a multiscale manner, preserving both local and global information (see Daugman (1990), Reed & Wechsler (1990) or Field (1999) for example). A review of such models is provided by Mallat (Chapter 5, 1999) . Such findings have provided a strong motivation for the development of texture analysis techniques founded upon multiscale methods.

Initial multiscale approaches to texture analysis were based upon Gabor functions, see for example Turner (1986), Bovic, Clark & Geisler (1990) and Dunn & Higgins (1995). However, Unser (1995) provides compelling arguments *against* such an approach, highlighting potential disadvantages including computational intensity and lack of filter orthogonality. Similar issues have also been raised by Chang & Kuo (1993). Conversely, discrete wavelet transforms can be computed efficiently and, in many cases, are orthogonal. Consider, for example the DWT or the “best basis” discrete wavelet packet transform: these may be calculated in order $O(N)$ and $O(N \log(N))$ operations respectively and are both orthogonal transforms. Coupled with the multiscale nature of wavelet transforms, these features have prompted the application of wavelets to texture analysis.

Below, we review several recently proposed techniques, paying particular attention to the work of Unser (1995). Further reviews of wavelets for texture analysis are provided by Sonka *et al.* (1999) and Scheunders *et al.* (1998).

4.4.2 The discrete wavelet transform

The use of the discrete wavelet transform for texture analysis was first suggested by Mallat (1989b). Noticing that wavelet representations could be interpreted as a form of basic primitive decomposition, Mallat was able to relate the discrete wavelet transform to the texton theory of Julesz (1981). This transform is also appealing as it is well localised and permits a decomposition into three different directions: vertical, horizontal and diagonal. However, it should be noted that one is not necessarily constrained to these three directions. Mallat (1989b) draws attention to alternative constructions, using non-separable wavelet bases such as those considered by Meyer (1992), which permit as many decomposition orientations as one desires.

4.4.3 Discrete wavelet packet transforms

Chang & Kuo (1993) reason that a potential disadvantage of using the DWT for texture analysis is that it focuses on the progressive analysis of the *low*-frequency smooths. They argue that although the discrete wavelet transform affords a multiscale image decomposition, the most significant information contained within a textured image is

often located within the *middle* frequencies. Thus, the DWT may not be suitable in many cases as the low frequency region “may not necessarily contain significant information”. In other words, it does not provide a suitably refined partition of the middle frequencies.

To combat this problem, Chang & Kuo (1993) suggest the use of the “tree structured” or wavelet packet transform, detecting “significant” frequency channels which are subsequently further decomposed, an approach which ensures an orthogonal transformation of the data. The measure used to decide which branches to decompose is based upon the normalised energy

$$e_j^l = \frac{1}{RS} \sum_{r=1}^R \sum_{s=1}^S |p_{j,(r,s)}^l| \quad \text{for } l = 1, 2, 3, 4;$$

where $p_{j,(r,s)}^1 = d_{j,(r,s)}^h$, $p_{j,(r,s)}^2 = d_{j,(r,s)}^v$, $p_{j,(r,s)}^3 = d_{j,(r,s)}^d$ and $p_{j,(r,s)}^4 = c_{j,(r,s)}$ – the level j horizontal, vertical, diagonal details and smooth respectively. This measure is also used by Chang & Kuo (1993) in the training and testing of various texture classification algorithms.

The application of the discrete wavelet packet transform to texture analysis is also considered by Saito & Coifman (1995) and Laine & Fan (1993). Saito & Coifman (1995) adopt the “Best Basis” approach of Coifman & Wickerhauser (1992), thus ensuring an orthogonal multiscale representation of the texture data. Energy based measures are then used to train and test two different classifiers: a classification tree and linear discriminant analysis-based classifier. Laine & Fan (1993) adopt a slightly different approach, evaluating a *complete* wavelet packet transform (WPT) of a textured region of interest. They then proceed to compare the classification performance of

1. the complete (i.e. redundant) WPT against that of the DWT;
2. an orthogonal WPT, based on a single given level, against the redundant WPT;
3. two different measures when constructing a vector of statistics used for pattern recognition. The first is an energy measure, based upon the sum of squared detail (or smooth) coefficients within a given packet image, whilst the second provides a measure of entropy: $H(x) = -\sum_{(r,s)} |x_{r,s}|^2 \log(|x_{r,s}|^2)$.

Their results indicate that, in comparison to the DWT, a “finer discrimination may be more strongly supported by additional subsets of wavelet packets (redundancy)”

and that classification based upon the wavelet packet energy measure was marginally superior to that based upon the entropy measure. They also observe that the vector of statistics based upon a redundant representation, using *all* level 1, 2 and 3 packets, results in poorer classification rates than those based upon an orthogonal representation (for example the complete set of level 3 wavelet packets). They reason that “this suggests that redundancy may increase uncertainty ... for the classifier employed in our study”. It is perhaps more reasonable to suppose that these higher misclassification rates were due to the vector entries being correlated and/or the so called “curse of dimensionality”.

4.4.4 Non-decimated discrete wavelet transforms

The lack of translation equivariance (TE) of the DWT is a well-known phenomenon which we considered in Section 2.5. Put simply, the consequence of non-TE is that a simple integer shift of the input signal frequently results in a non-trivial change in the DWT of the signal. This is clearly undesirable, particularly when considering images with multiple texture features such as figure 4.2, for upon translation, markedly different features could be returned at texture boundaries. To remedy this issue, Unser (1995) proposes the use of the Discrete Wavelet Frame (DWF), a form of non-decimated wavelet transform, for texture analysis. Like the NDWT proposed by Nason & Silverman (1995) and others, the DWF is “similar” to computing the DWT for all circular shifts (in 1-D) of the input signal (see Unser (1995) for further construction details).

Unser (1995) adopts the following approach to texture characterisation: Given a textured image, R , of dimension N^2 for $N = 2^J$, $J \in \mathbb{N}$, we decompose R with respect to the DWF, giving a $3J + 1$ component vector:

$$\begin{aligned} \mathbf{y}_R(r, s) &= (y_i(r, s))_{i=1, \dots, 3J+1}; \\ &= [s_0(r, s) \quad d_1^h(r, s) \quad d_1^v(r, s) \quad d_1^d(r, s) \quad \dots \quad d_{J-1}^d(r, s)]. \end{aligned}$$

A concise texture representation is afforded by the “channel variances”, $\text{Var}(y_i)$. These are estimated by

$$v_i = \frac{1}{n^2} \sum_{r=1}^N \sum_{s=1}^N y_i^2(r, s). \quad (4.3)$$

Using the v_i as classifier inputs for a test involving several Brodatz images, Unser (1995) observes that the DWF approach outperforms the DWT. Furthermore, the DWF compares favourably with the *local linear transform* (LLT) proposed by Unser (1986) (see Section 4.3.4). This is of particular interest, for as we observed earlier, the LLT regularly outperforms other standard single resolution methods, such as the co-occurrence and correlation approaches. Perhaps, most interestingly of all, Unser (1995) concludes that “increasing the number of vanishing moments . . . of the underlying basis functions does not seem to have any real advantage for texture analysis and discrimination”.

4.4.5 Other wavelet-based approaches to texture analysis

The recent “Wavelets for Texture Analysis” initiative at the University of Antwerp (<http://www.ruca.ua.ac.be/visielab/wta/wta.html>) has generated several novel texture analysis techniques. Van de Wouwer, Scheunders & Van Dyck (1999a) consider the application of the discrete (undecimated) wavelet transform to texture analysis, introducing two new sets of features. Their first set is based upon the experimental observation made by Mallat (p. 686, 1989) that the histogram of gray scales within a given detail image may be modelled by the family of exponentials:

$$h(u) = K \exp^{-(|u|/\alpha)^\beta} .$$

This is, a reflected Weibull distribution, with scale parameter α and shape parameter β . Estimates of these parameters were subsequently used as inputs to the texture classification algorithm.

The second set of measures proposed by Van de Wouwer *et al.* (1999a) is motivated by the co-occurrence approach of Haralick *et al.* (1973). Instead of calculating co-occurrence matrices for the image, Van de Wouwer *et al.* (1999a) calculate co-occurrence matrices of the wavelet detail images. Measures including inertia, total energy and entropy of the detail image are evaluated from the co-occurrence matrices for inclusion in the feature vector. Van de Wouwer *et al.* (1999b) consider the problem of classifying coloured texture images, building upon the “energy signature” approach of Laine & Fan (1993) and Unser (1995), whilst Van de Wouwer (Chapter 5, 1998) tackles the problem of rotation-invariant feature extraction, using a non-separable wavelet

transform.

To overcome problems of translation invariance and poor directional selectivity within the DWT, novel multiscale transforms such as the non-decimated wavelet packet transform, dual-tree complex wavelet transform (Kingsbury, 1999) and steerable pyramid (Simoncelli & Freeman, 1995) have been used for various texture analysis tasks. See Laine & Fan (1996), de Rivaz & Kingsbury (1999) or Portilla & Simoncelli (2000) for example.

Finally, recent research by members of the Digital Signal Processing group at Rice University has focused upon hidden Markov tree modelling of the structure within wavelet transforms and its application to various image and texture based applications: for example Crouse, Nowak & Baraniuk (1998), Romberg, Choi & Baraniuk (2000), Venkatachalam, Choi & Baraniuk (2000) and Choi *et al.* (2000). Such models can capture the key features of many real world images, for example the persistent nature of discontinuities in the wavelet domain. However, the application of such approaches can be computationally expensive (see Romberg *et al.* (2000)). To combat such expense, it is often convenient to reduce the number of model parameters by assuming that, within any given scale, the parameters are constant over location.

4.4.6 Concluding remarks

The measures used by Unser (1995) for texture classification are similar to those which are considered in Section 4.5 when we apply the LS2W model to various texture analysis problems. However, our approach differs from many detailed in Sections 4.4.2 – 4.4.5 in one key respect: few are model-based. It is therefore sometimes difficult to associate a meaning with the quantities used for texture feature extraction. Moreover, no direct consideration can be made as to whether the redundancy of various approaches, such as those proposed by Laine & Fan (1993, 1996) and Unser (1995), causes power to leak across decomposition scales and directions. Such leakage could nullify any meaning one might attempt to attach to the measures. Additionally, the lack of any formal modelling procedure makes the task of texture synthesis using these approaches very difficult.

4.5 The LS2W model and texture analysis

We now consider applying the LS2W modelling approach to texture analysis. Our approach is based upon the observation that textures possess structure at different scales. It seems natural, therefore, to conjecture that many textures may be discriminated on the basis of their local (scale-based) spectral structure. To achieve this, some form of summary measure of the structure contained within a textured image is required.

The work of Chapter 3 allows us to formalise the analysis of textured images within a stochastic framework, demonstrating that in the case of the redundant non-decimated wavelet transform, power can leak across both scale and direction. This bias may be corrected (asymptotically) by applying the inverse of the inner product matrix of discrete autocorrelation wavelets to the array of raw local wavelet periodograms (LWPs), as in equation (4.4). This estimate of the local wavelet spectrum (LWS) provides a form of location, direction-scale decomposition of the covariance structure within a LS2W process. To obtain a statistically consistent estimate of the LWS, we must also smooth the LWP.

Given a textured image, T_I , of dimension $2^J \times 2^J$, the collection of local wavelet spatial periodograms, $\{\mathbf{L}(\mathbf{z})\}$, forms an array of dimension $3J \times 2^J \times 2^J$. As a first step to investigating the potential of the LS2W approach to texture analysis, we consider the following statistic, one of many which could be based upon the (smoothed) local wavelet periodogram:

$$\mathbf{t}(T_I) = \sum_{\mathbf{z}} \hat{\mathbf{L}}(z) = \sum_{\mathbf{z}} A_J^{-1} \tilde{\mathbf{I}}([\mathbf{z}\mathbf{R}]). \quad (4.4)$$

Any given element, $\{\mathbf{t}(T_I)_\eta\}_{\eta(j,l)}$, provides a measure of the contribution made to the overall local variance structure at scale j within direction l . This measure is similar to the ‘‘channel-variance’’ proposed by Unser (1995). However, whilst Unser’s feature set is motivated by the conservation of energy within a tight wavelet frame, no consideration is made of how the redundancy of the DWF can affect estimates of local spectral features.

In the remainder of this chapter, we consider the application of the LS2W approach to various texture problems. To start with, we consider its potential in discriminating and classifying between various standard Brodatz textures, comparing our results

with alternative measures based upon the DWT, (uncorrected) NDWT and the Fourier spectrum. Then in Section 4.6, we consider the application of the LS2W modelling approach to various texture analysis problems encountered by an industrial collaborator. Again, we will compare our results with those of other approaches. It should be noted that due to the small size of the textures considered, we focused on building discriminant models using sub-images of dimension 128×128 . Such sub-images provide information at a variety of fine and medium scales whilst also allowing several samples to be taken from the original images. Should larger images become available, then naturally it would be interesting to investigate the effect of using larger sub-images in the analysis of such textures. This is left as an avenue for future work.

4.5.1 The Brodatz experiment

The Brodatz (1966) texture collection has become a standard test set within the texture analysis community. Thus, it seems natural to commence our investigation with this dataset. We focus attention on the six 640×640 images displayed in figure 4.4. These figures were supplied by T. Randen and may be downloaded from <http://www.ux.his.no/tranden>. As can be seen, the textures have many different features: some, such as figure4.4(5) are characterised by fine scale structure whilst others possess coarser features. However, each image is in some sense “stationary” as all exhibit some form of regularity within their structure.

The first experiment which we consider attempts to discriminate between the six textures displayed in figure 4.4. To the eye, these textures appear very different from one another. Hence we would hope that our LS2W-based approach is able to discriminate effectively between these different texture types. Fifty sub-images, S_i , of dimension 128×128 were randomly sampled from the upper half of each image. The feature vector, $\{\mathbf{t}(S_i)\}_{i=1,\dots,300}$, was evaluated for each sub-image and stored ready for input into a discrimination algorithm. This procedure was implemented in S-Plus, using the WaveThresh and LS2W software suites, making extensive use of the LS2W-function `cddews`. See Appendix C or <http://www.stats.bris.ac.uk/~maiae/LS2W> for further details.

In this initial analysis, the Haar family of wavelets was used to estimate the LWS, the resulting spectra being smoothed using the Daubechies Extremal Phase (N=4)

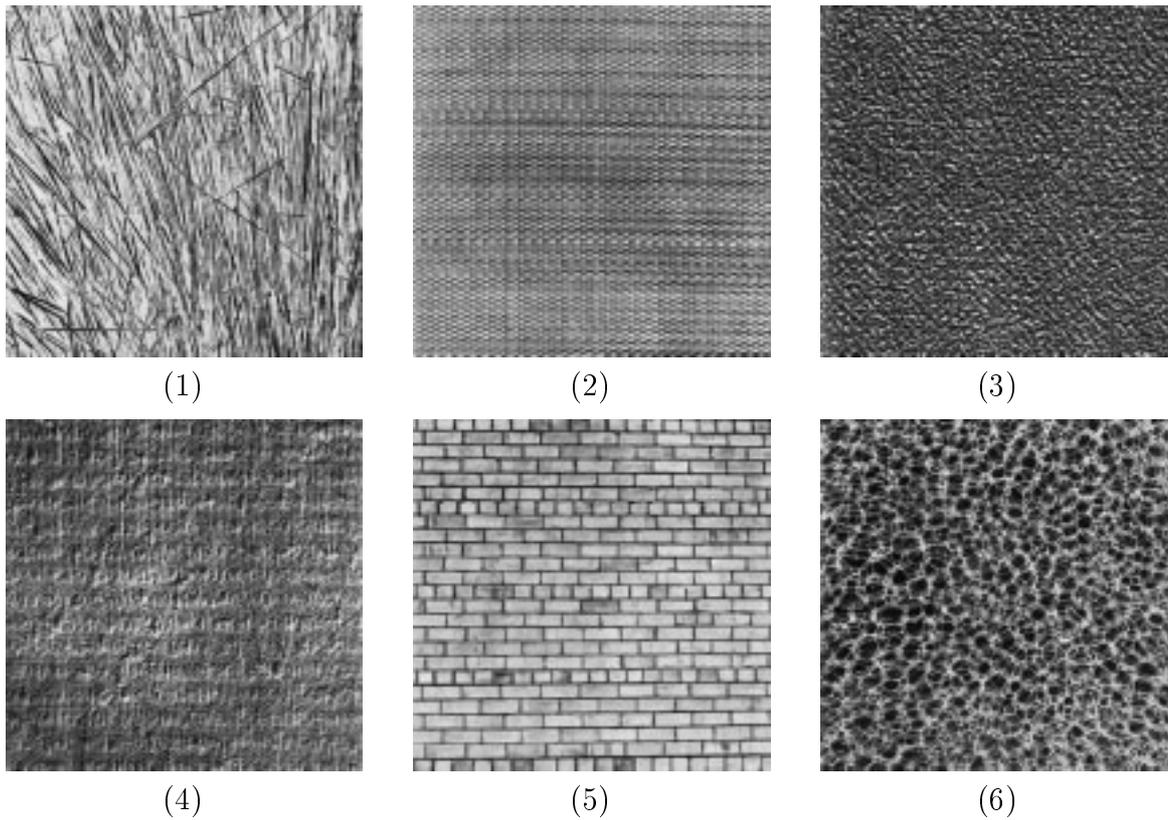


Figure 4.4: Brodatz images used to test the LS2W model's ability to discriminate and classify between different textures. Images provided by T. Randen, Schlumberger Stavanger Research.

wavelet with a soft, locally stationary universal threshold as proposed by Nason *et al.* (Theorem 4, 2000). The variance estimator used in this case was the sample variance. Fisher's linear discriminant analysis (LDA) was used as an exploratory discrimination method (see Mardia, Kent & Bibby (1979) or Ripley (1996) for comprehensive reviews of this technique). The results of this analysis may be seen in figure 4.5. The different texture types can be discriminated quite easily, with each texture class being well localised within the discriminant plane. Note however that texture types 3 and 4, though discriminable, are situated close together. This feature is also present in the linear discriminant plot of texture features based upon the (uncorrected and unsmoothed) Haar non-decimated wavelet transform (figure 4.6). The vector of feature statistics was formed by summing the squared non-decimated wavelet coefficients over location within each direction-scale pair.

Figures 4.7 and 4.8 display plots of the first two discriminant variables based on

texture measures created using the Haar discrete wavelet transform and the Fourier transform respectively. In the case of the discrete wavelet transform, each element of the feature vector was formed by summing the squared detail coefficients within a given direction-scale pair. The Fourier feature set was created by summing elements within frequency rings having a depth of 10 frequency units (see figure 4.3(a)). Figure 4.7 displays a good level of discrimination between all texture classes. In contrast, the Fourier-based measures have difficulty in discriminating between texture classes 3, 4 and 6. As several of these images are characterised by edges (i.e. discontinuities), we would not expect the Fourier approach to discriminate as efficiently between the various classes.

The second experiment which we consider attempts to classify a test set of three hundred sub-images of dimension 128×128 into one of the six Brodatz texture classes. Fifty sub-images were randomly sampled from the lower half of each image in figure 4.4. Features were calculated for each test sub-image using the LS2W, (uncorrected) non-decimated wavelet transform, discrete wavelet transform and Fourier approach detailed above. To start with, sub-images were assigned to a texture class according to the following simple rule:

1. Perform a LDA on the training set of all six texture classes.
2. Then for each test sub-image, calculate the LDA-transformed feature vector and assign \tilde{S}_i to the class whose mean is closest (in the Euclidean sense).

The results of this experiment are displayed below:

Method	Percentage Correctly Classified
LS2W model	83.3
Uncorrected NDWT	83.3
DWT	91.0
Fourier	73.3

Table 4.1: Percentage of Brodatz textures classified correctly using various feature extraction techniques. Wavelet based features were generated using the Haar wavelet and the LS2W model was smoothed using Daubechies Extremal Phase (N=4) wavelets.

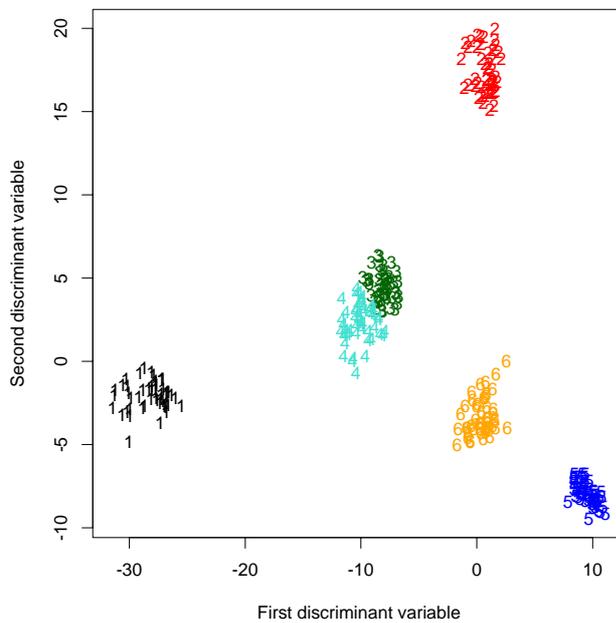


Figure 4.5: Plot of the first two linear discriminant axes for LS2W-based features obtained from the Brodatz textures displayed in figure 4.4. Texture classes 1 through 6 refer to sub-images sampled from figure 4.4(1) – (6) respectively.

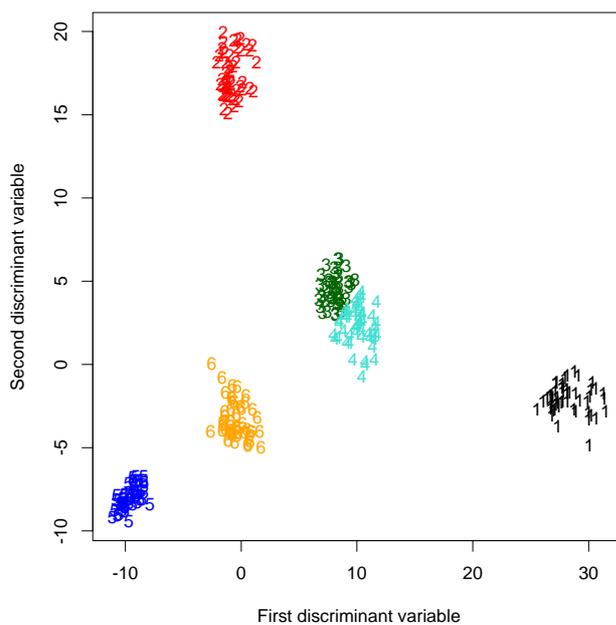


Figure 4.6: LDA plots for measures of the Brodatz textures based on the (uncorrected) non-decimated wavelet transform.

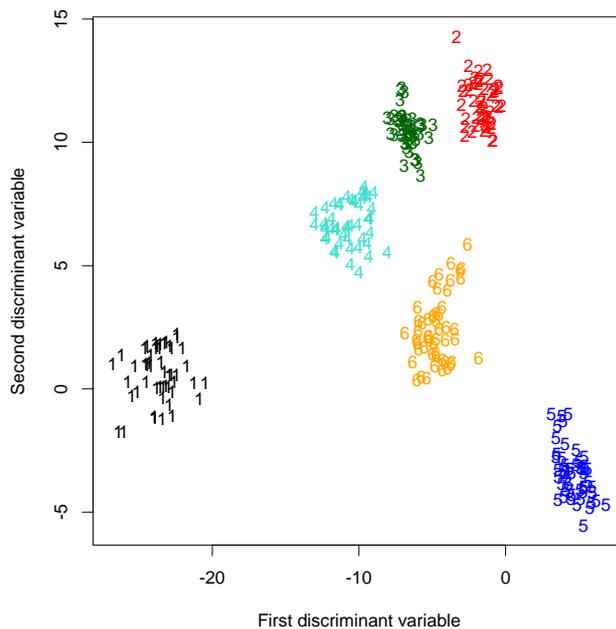


Figure 4.7: LDA plots for measures of the Brodatz textures based on the discrete wavelet transform.

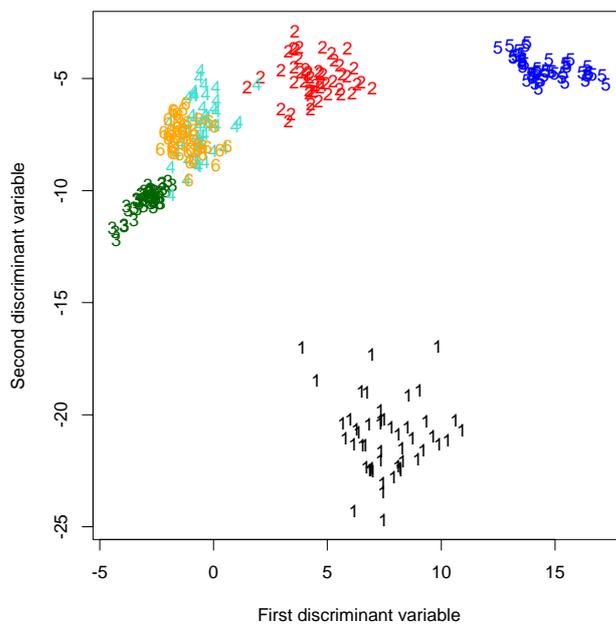


Figure 4.8: LDA plots for measures of the Brodatz textures based on the Fourier transform.

As we would expect from looking at the LDA plots in figures 4.5 – 4.8, these errors are invariably due to texture type 4 being misclassified as type 3. Looking at figure 4.4 this is perhaps unsurprising, the structure in both images being reasonably similar. It is also interesting to note that the multiscale approaches produce more accurate classification results. However, we would expect this as these images are characterised by edges.

Noticing that it is difficult to discriminate between, for example, texture types 3 and 4 in the LDA plots associated with the multiscale approaches (figures 4.5 – 4.7), one might in practice consider a two-stage scheme in an attempt to improve classification performance. Such an approach is described below:

LS2W, NDWT and DWT approach: A LDA is performed on the training set of all six texture classes. If the LDA-transformed feature vector of a test sub-image, \tilde{S}_i , is closest (in the Euclidean sense) to the mean of class 1, 2, 5 or 6, then the sub-image is assigned to that class. Otherwise, a LDA is performed on the training set of texture classes 3 and 4 *only*. The sub-image is then assigned to whichever texture the LDA-transformed feature vector is closest to.

Fourier approach: Recall from figure 4.8 that only texture classes 1, 2 and 5 are easily discriminable using the Fourier-based feature vector. Thus the following two-stage classification scheme, founded on the first two linear discriminant variables of a test set, was adopted when using Fourier-based feature vectors:

- i) Perform a LDA on the training set of all six texture classes.
- ii) If the LDA-transformed feature vector of \tilde{S}_i is closest (in the Euclidean sense) to the mean of class 1, 2 or 5 then assign the sub-image to that class.
- iii) Otherwise, perform a LDA on the training set of texture classes 3, 4 and 6.
- iv) Assign \tilde{S}_i to the class to which the LDA-transformed feature set is closest.

The results obtained may be seen in table 4.2.

Clearly this modified scheme, which takes into consideration the discriminant structure of this specific data set, produces superior results. It is evident from the above

Method	Percentage Correctly Classified
LS2W model	100
Uncorrected NDWT	100
DWT	98.7
Fourier	92.3

Table 4.2: Percentage of Brodatz textures classified correctly using various feature extraction techniques in conjunction with a two-stage classification scheme. Wavelet based features were generated using the Haar wavelet and the LS2W model was smoothed using Daubechies Extremal Phase (N=4) wavelets.

that the wavelet-based approaches achieved improved classification rates compared with the Fourier-based approach. However, with seemingly perfect classification rates returned by the LS2W and (uncorrected) non-decimated wavelet transform methods, one must ask how realistic is it to judge a classifier on the basis of its ability to discriminate between bricks and bubbles? As we shall see in the next section, the classification problems which are encountered in reality are frequently far more subtle.

4.6 Applications of the LS2W model

4.6.1 Exploratory analysis of piled material images

The following texture analysis problem arises from work with an industrial collaborator. Six samples of identical material were buffed to varying degrees in an attempt to simulate different levels of garment wear. The effect of this buffing is to induce pilling, a building up of fibrous balls on the surface of the material. As can be seen in figure 4.9, certain materials have a very fine level of pilling (for example figure 4.9(1)) whilst others are heavily pilled (figure 4.9(6)). Each image is 1024×1024 pixels in size. Unfortunately, physical scale was not considered to be an issue in the original application which generated this data. Consequently no record was made of the size of the textiles contained in these images. In future work, such records will be collected.

Interestingly, some of these samples are very difficult to discriminate between visually. To investigate the ability of LS2W approach to discriminate between these different textures, fifty sub-images of dimension 128×128 were randomly sampled from

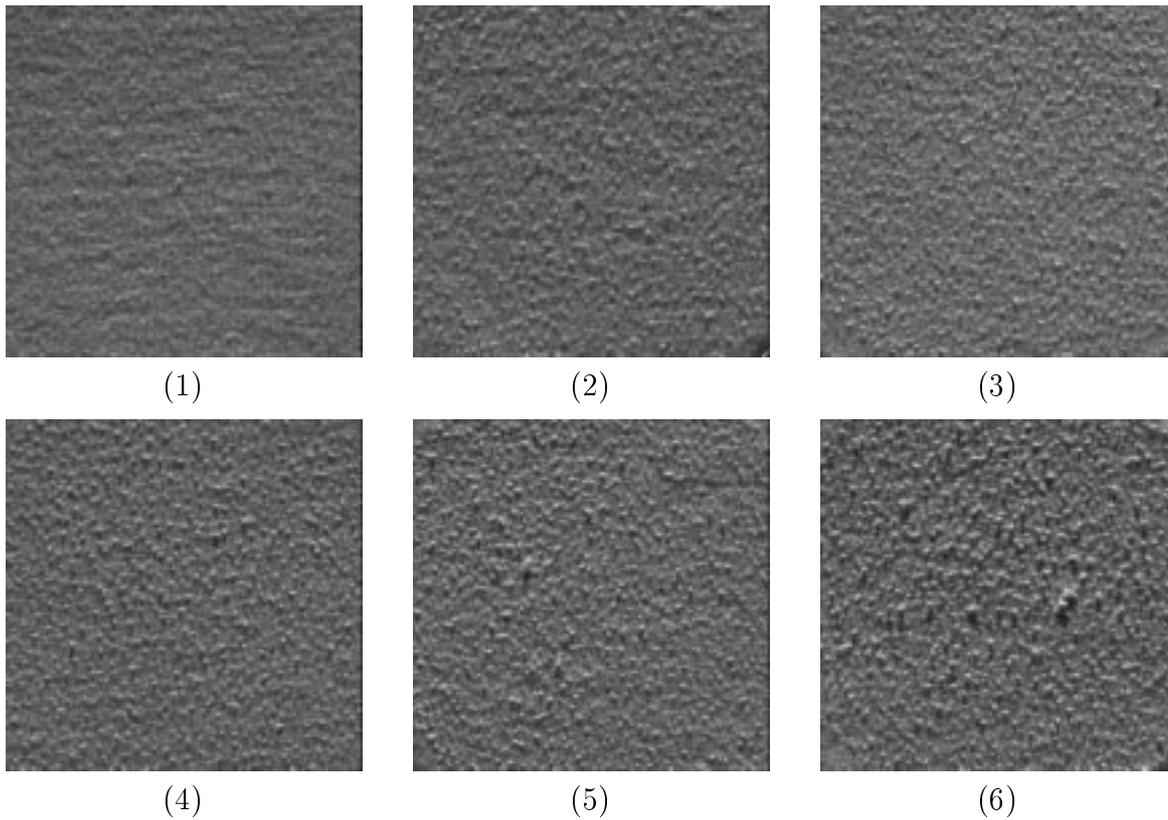


Figure 4.9: Images of materials pilled to varying degrees. Image (1) contains a fine pill material, whilst image (6) contains heavy pilling. Images provided by Unilever Research.

the left hand half of each image. For each sub-image, the following classes of feature sets were evaluated:

1. LS2W measures, smoothed using the Daubechies Least Asymmetric (N=6) wavelet;
2. Uncorrected non-decimated wavelet transform;
3. Discrete wavelet transform;
4. Fourier transform features using rings of 10 frequency units.

This linear partition of the frequency space (item 4) was thought to be reasonable for this initial study, being neither particularly fine nor coarse. Other choices of partition could consist of a fine linear partition of the space or a logarithmic partition, thus mimicking the division performed by wavelets. Daubechies Extremal Phase (N=3) wavelets were used for all wavelet-based measures.

Figure 4.10 displays a plot of the first two linear discriminant axes for the LS2W feature set. Note how the different pill levels span the plane: heaviest pill on the left and lightest pills on the right. The different classes are reasonably well separated, the analysis even being able to separate pill levels 5 and 6, two images which appear very similar to the eye. However, it should be noted that pill levels 3 and 4 overlap. Figure 4.11 displays a plot of the first and third discriminant variables for this feature set, whilst figure 4.12 displays the first and fourth discriminant variables. Observe how the quality of separation between the various texture classes diminishes as we proceed from the second through to the fourth discriminant variable. Note in particular how the separation between pill levels 5 and 6 disappears by figure 4.12.

Figures 4.13 – 4.15 display the plots of the first two linear discriminant axes for the feature sets obtained from the non-decimated wavelet, discrete wavelet and Fourier transforms respectively. With the exception of figure 4.15, pill levels 3 and 4 again overlap. By contrast, the Fourier based features used in figure 4.15 have difficulty in discriminating between texture classes 1, 2 and 4, although classes 3, 5 and 6 are well separated.

The weighting of the different contributions to the linear discriminant variables may provide us with a useful insight into which scales permit discrimination between the various texture types. Table 4.3 shows the weighting of the various contributions to the first and second linear discriminant variables formed from the LS2W training set. Many scales contribute, the vertical scale structure having a particularly strong influence especially on the second discriminant variable. This would appear to correspond with the direction of the weave within the material.

Classification

With such subtle differences between the images displayed in figure 4.9, it is interesting to see whether the various feature extraction schemes can provide measures which permit reasonable classification rates. To this end, a test set of fifty sub-images of dimension 128×128 were randomly sampled from the right half of each pill image. The LS2W, uncorrected non-decimated wavelet transform, discrete wavelet transform and Fourier “spectral ring” feature sets were evaluated for each sub-image. These feature vectors were subsequently used to classify the sub-images to a pill class.

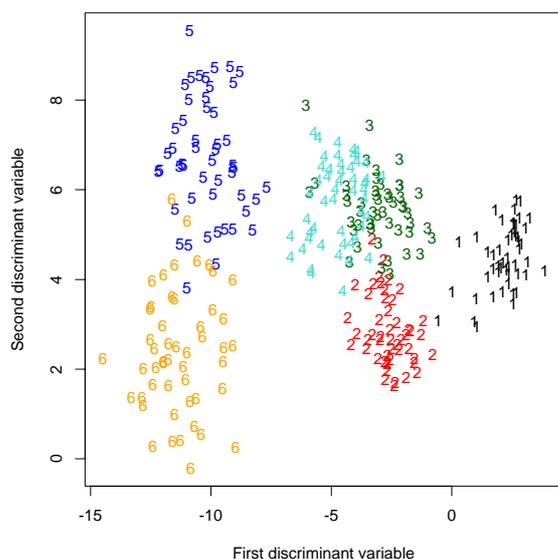


Figure 4.10: LDA plots for measures of the Pill images based on the LS2W model. Sub-images were analysed using the non-decimated Daubechies Extremal Phase ($N=3$) wavelet transform, smoothed using the Daubechies Least Asymmetric ($N=6$) wavelet and corrected.

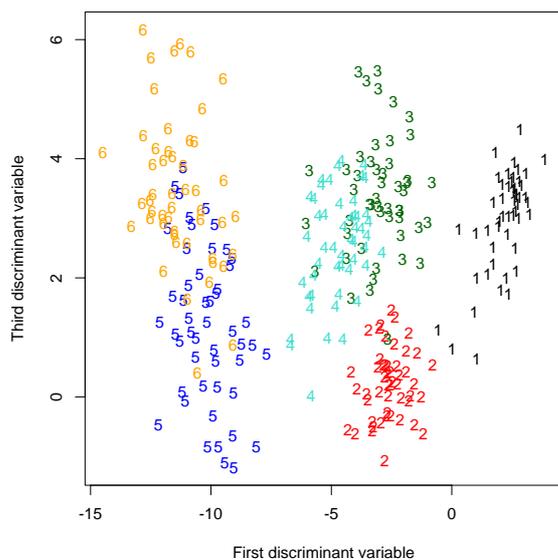


Figure 4.11: LDA plot of the first and third discriminant variables for the Pill images based on the LS2W model.

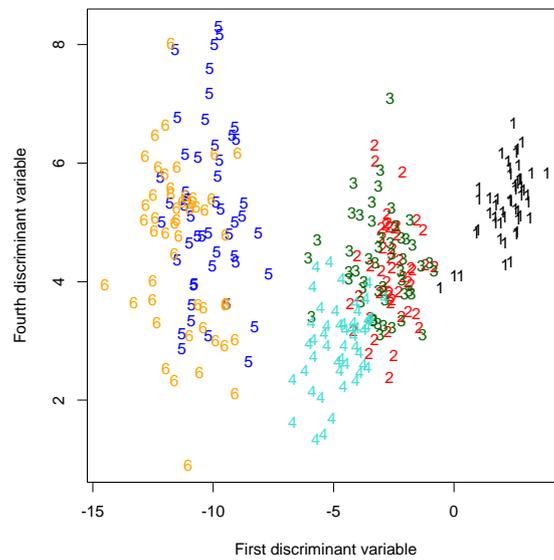


Figure 4.12: LDA plot of the first and fourth discriminant variables for the Pill images based on the LS2W model.

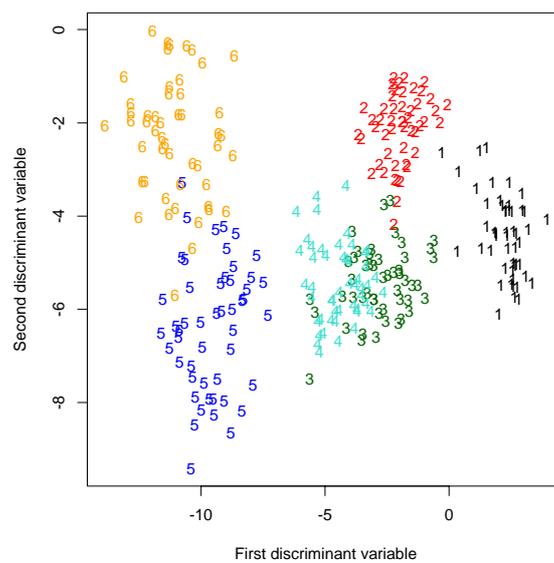


Figure 4.13: LDA plots for measures of the Pill images based on the uncorrected, non-decimated wavelet transform.

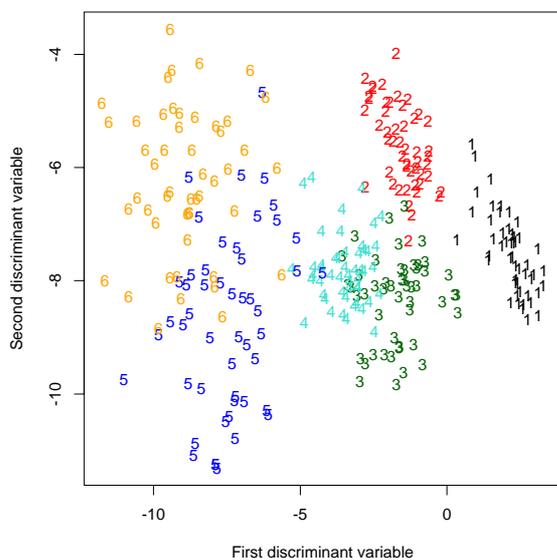


Figure 4.14: LDA plots for measures of the Pill images based on the discrete wavelet transform.

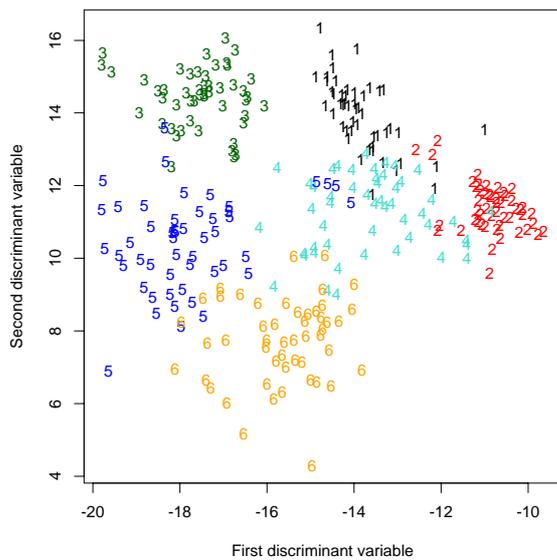


Figure 4.15: LDA plots for measures of the Pill images based on the Fourier transform.

Direction/Scale Pairs	First Discriminant Variable	Second Discriminant Variable
(1,v)	6.6	-33.3
(2,v)	-9.6	-7.6
(3,v)	3.0	-98.4
(4,v)	-7.6	5.9
(5,v)	2.5	4.0
(6,v)	-12.4	16.3
(7,v)	-4.7	3.4
(1,h)	2.4	-0.9
(2,h)	-2.7	10.7
(3,h)	-1.0	0.2
(4,h)	-1.9	-0.5
(5,h)	-2.0	0.7
(6,h)	-0.1	0.3
(7,h)	-0.7	-1.1
(1,d)	-0.7	-0.6
(2,d)	0.6	-0.1
(3,d)	0.3	2.5
(4,d)	-0.1	0.4
(5,d)	-0.6	-0.3
(6,d)	1.0	-4.1
(7,d)	-0.6	-3.5

Table 4.3: The first and second discriminant variables for the LS2W-based linear discriminant analysis model of the pilled images, constructed using the Daubechies Extremal Phase (N=3) wavelets ($\times 10^{-6}$).

Method	Percentage Correctly Classified
LS2W model	57.7
Uncorrected NDWT	51.7
DWT	54
Fourier	66

Table 4.4: Percentage of Pill textures classified correctly using a single step classification algorithm. Wavelet based features were generated using the Daubechies Extremal Phases (N=3) wavelet and the LS2W model was smoothed using Daubechies Least Asymmetric (N=4) wavelets.

To begin with, a single stage classification scheme based upon the first four linear discriminant variables was performed. Each sub-image was classified using a minimum (Euclidean) distance rule. In the event that the distances between a sub-image and two (or more) texture classes were equal, the sub-image was deemed to be unclassified. The results of this approach are displayed in Table 4.4. As can be seen, barely half the sub-images are classified correctly when using the multiscale methods — the LS2W approach achieving the best results of the three. Note however, that the Fourier approach classifies approximately two thirds of the sub-images correctly. These comparatively poor misclassification rates are not particularly surprising for texture classes 2, 3 and 4 are not well separated in the linear discriminant plots of figures 4.10 – 4.15.

In practice, it is likely that one would adopt a sequential classification approach, akin to that described in Section 4.5.1 when discriminating between such visually similar textures. This is the next experiment which we consider. We focus on using simply the first two linear discriminant variables since these display the best separation between the various texture classes. The results of this experiment are displayed in Table 4.5. The most notable feature of these results is that no method achieves a perfect (or even near perfect) classification rate. This is understandable, for even the human eye can find it difficult to discriminate between some of these pill images. Sub-images from pill levels 3, 4 and 5 were by far the most frequently misclassified samples.

Of the three multiscale approaches, the LS2W approach faired best of all with 70.7% of sub-images classified correctly. However yet again, the Fourier transform achieved

Method	Percentage Correctly Classified
LS2W model	70.7
Uncorrected NDWT	66.7
DWT	65.7
Fourier	72.3

Table 4.5: Percentage of Pill textures classified correctly using various feature extraction techniques. Wavelet based features were generated using the Daubechies Extremal Phases (N=3) wavelet and the LS2W model was smoothed using Daubechies Least Asymmetric (N=4) wavelets.

the best classification rate (72.3%) with this dataset. This result is not surprising, for these images have a regular form. Consequently their spectral properties in the wavelet domain will also be regular, thus implying that the underlying process is, in some sense, stationary. Hence we would expect these textures to be well-discriminated by Fourier features.

4.6.2 Exploratory analysis of hair images

The final problem which we consider is the discrimination and classification of two images of straight hair, displayed in figure 4.16. The images arise from research by an industrial collaborator. Figure 4.16(a) depicts an image of hair which was given a treatment which we call Treatment A, whilst figure 4.16(b) depicts an image of the same head of hair having received a second treatment, Treatment B. Each image is 576×768 pixels in size. Unfortunately, physical scale was not considered to be an issue in the original application which generated this data. Consequently no record was made of the size of these images. In future work, such records will be collected.

Clearly, both images are affected by a variable light condition — the left hand side of both images appear to be shaded somewhat. To counteract this effect, we could pre-process the image in some way: a popular image processing technique being histogram equalisation (see Sonka *et al.* (1999) for further details). Unfortunately, this approach has a tendency to emphasise sudden changes in structure (i.e. discontinuities) whilst diminishing the effect of subtle differences. Consequently, we do not pursue this avenue at this time. Instead, we deal with the images in their original format.

Figure 4.17 displays LDA plots for various feature sets obtained from fifty 128×128 sub-images which were randomly sampled from the *upper* half of each image in figure 4.16. The three wavelet-based LDA plots display a good level of discrimination between the two texture types whilst figure 4.17(d), based upon the Fourier transform, does not achieve such a clear distinction between the two texture types.

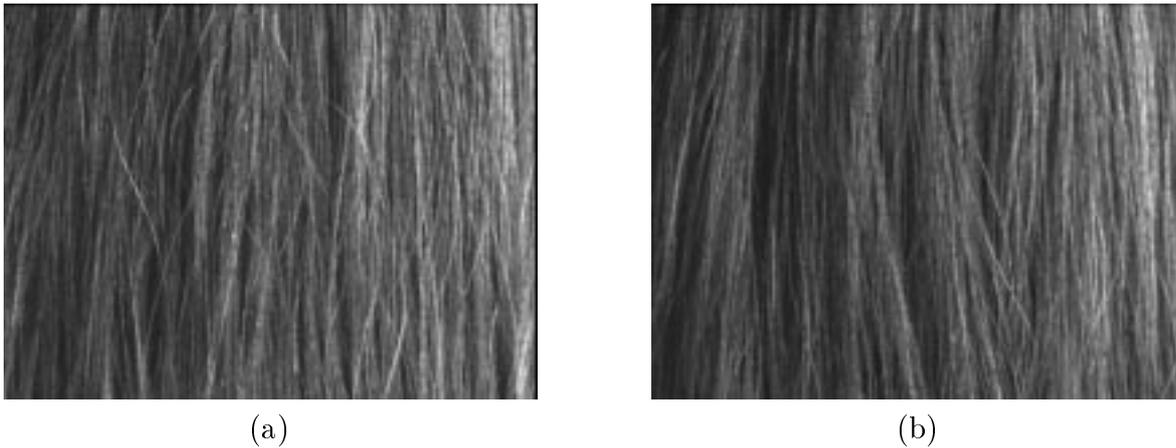


Figure 4.16: Images of straight hair. Image (a) depicts hair which was washed and dried according to treatment A, whilst image (b) depicts hair which was washed and dried according to treatment B. Images provided by Unilever Research.

Table 4.6 provides the weightings of the various direction-scale contributions to the LDA plot in figure 4.17(a). Note how the greatest influence arises from the vertical scale coefficients, as one might expect when comparing two images of near vertical straight hair.

Direction/Scale Pairs	First Discriminant Variable ($\times 10^{-4}$)
(1,v)	-3.103
(2,v)	-1.679
(3,v)	14.980
(4,v)	2.826
(6,v)	2.723
all other pairs	< 1

Table 4.6: The first discriminant variable for the LS2W-based linear discriminant analysis model of the hair images.

In an attempt to ascertain the potential of the LS2W approach to classify hair images, fifty further sub-images of dimension 128×128 were randomly sampled from

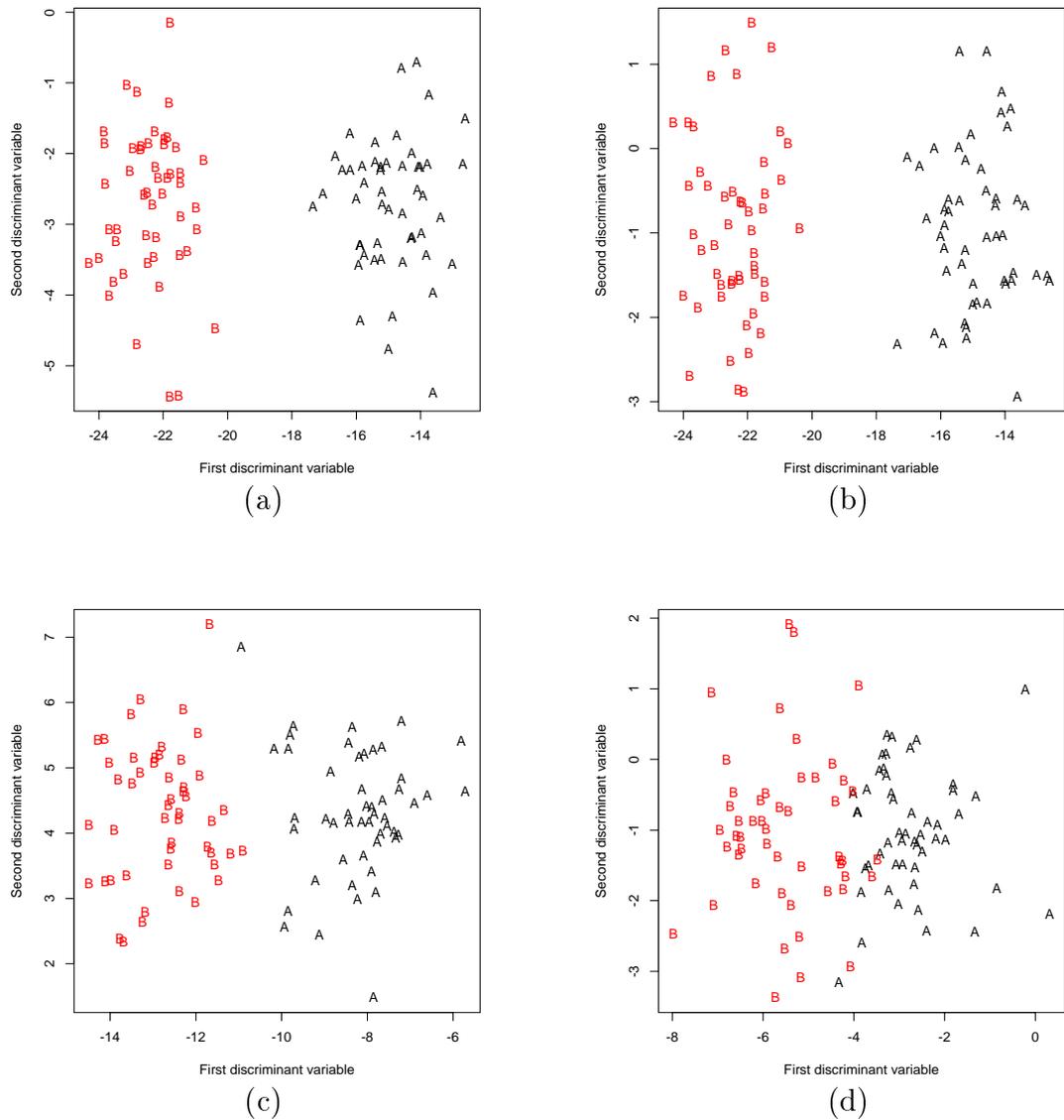


Figure 4.17: LDA plots for measures of the Hair images based on the LS2W model. Plot (a) is based on LS2W-derived features, whilst plots (b) and (c) were formed using non-decimated and discrete wavelet transform features respectively. The Daubechies Extremal Phase ($N=4$) wavelet transform was used in each case, the LS2W-spectra being smoothed using Daubechies Least Asymmetric ($N=6$) wavelets. Plot (d) was formed using Fourier-based spectral rings having a depth of 10 pixels.

the *lower* half of each image in figure 4.16. LS2W, non-decimated wavelet, discrete wavelet and Fourier-based features were evaluated for each sub-image, the resulting vector of statistics being transformed, using the appropriate first linear discriminant. As in Sections 4.5.1 and 4.6.1, each sub-image was assigned to the texture class whose mean was closest to the (transformed) feature co-ordinate.

Classification is far from perfect for this data set. As can be seen in Table 4.7, the three wavelet-based classifiers achieve higher classification rates than the Fourier-based approach. However, approximately 30% of sub-images were misclassified by even the best approach. Removing the variable lighting from the original images, using histogram equalisation would, with the exception of the Fourier-set, appear to have little effect on the rate of misclassification.

Method	Percentage Correctly Classified (Original)	Percentage Correctly Classified (Histogram equal.)
LS2W model	71	68
Uncorrected NDWT	71	68
DWT	73	72
Fourier	60	69

Table 4.7: Percentage of hair sub-images classified correctly using various feature extraction techniques. Wavelet based features were generated using the Daubechies Extremal Phases (N=4) wavelet, the LS2W model being smoothed using Daubechies Least Asymmetric (N=4) wavelets.

Note how in both analyses the LS2W and uncorrected NDWT approaches achieve identical classification rates. This result is not surprising, for there exists good separation within the LDA plots associated with both these feature sets (see figures 4.17 (a) and (b)). The reason for this similarity in results is that the original images have a very regular form. In other words they are reasonably stationary and so their local wavelet spectra will be constant across location. Hence, correction by A^{-1} will not greatly affect the analysis. In Section 4.6.3 we will consider a non-stationary texture analysis example which demonstrates the potential of the LS2W approach.

Table 4.8, below displays the results of a similar classification experiment using the Haar wavelet transform. Note how the results are comparable with those which were

achieved with the Daubechies Extremal Phase (N=4) wavelet. This is in line with the findings of Unser (1995) who noted that increasing the number of vanishing moments of the underlying wavelets does not significantly affect classification.

Method	Percentage Correctly Classified
LS2W model	68
Uncorrected NDWT	68
DWT	66

Table 4.8: Percentage of hair sub-images classified correctly using various feature extraction techniques. Wavelet based features were generated using the Haar wavelet, the LS2W model being smoothed using Haar wavelets.

4.6.3 Non-stationary texture classification

Recall that the power of the LS2W modelling approach lies in its ability to analyse images whose covariance structure is *locally* stationary. In other words, it is well suited to the analysis of images whose covariance structure is *globally* non-stationary, but *stationary* within a local region. Crucially, the LS2W approach is able to correct artefacts which arise as a consequence of the inherent redundancy of the NDWT, the transform used in the estimation of the spectral structure of an image. The result of this correction is that we are able to reduce the effect of power spreading across scales and directions. This is in stark contrast to using the squared detail coefficients of the NDWT, see for example figures 3.5 and 3.6. It is therefore anticipated that our modelling approach will fare well when classifying between non-stationary textures. To investigate it's potential, we consider the following simulated problem:

Suppose a certain tile making process generates two texture types, T1 and T2 (see figure 4.18). T1 represents a desirable tile type whilst T2 is deemed to be a spoiled tile. The task therefore is to find an approach which is able to achieve a high rate of correct classification.

Two classification approaches are considered, the first being based upon our LS2W model whilst the second uses the NDWT. The LS2W classification approach

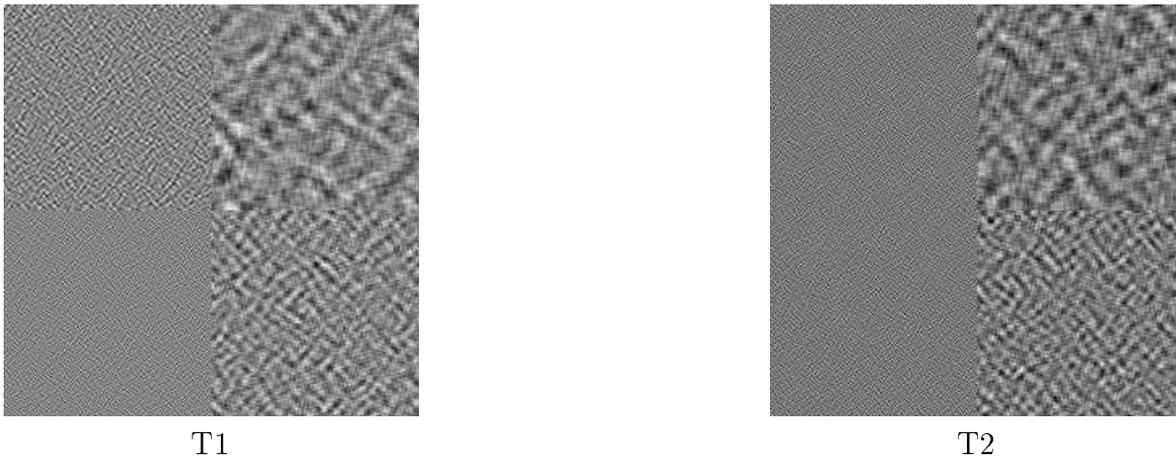


Figure 4.18: Simulated examples of non-stationary textures.

is structured as follows: for each of 25 realisations of tile types T1 and T2, calculate the local wavelet periodogram using the Haar transform, smoothing each periodogram using Daubechies Extremal Phase (N=4) wavelets. Then calculate the mean local wavelet periodogram structure within each tile type, thus obtaining two spectral models, $\tilde{\mathbf{I}}_{T1}$ and $\tilde{\mathbf{I}}_{T2}$, of each tile's local wavelet spectral structure

A further 50 realisations of each tile type, $\{T_i\}_{i=1,\dots,100}$, are then used as a test set for classification purposes. For each test case, calculate the LWP, again using the squared detail coefficients of the Haar NDWT smoothed using the Daubechies Extremal Phase (N=4) wavelets. A tile T_i is then assigned to type T1 if

$$\sum_{j,l,\mathbf{u}} \left(\tilde{I}_{j,l,\mathbf{u};T1} \Leftrightarrow \tilde{I}_{j,l,\mathbf{u};T_i} \right)^2 < \sum_{j,l,\mathbf{u}} \left(\tilde{I}_{j,l,\mathbf{u};T2} \Leftrightarrow \tilde{I}_{j,l,\mathbf{u};T_i} \right)^2 .$$

and type T2 if

$$\sum_{j,l,\mathbf{u}} \left(\tilde{I}_{j,l,\mathbf{u};T1} \Leftrightarrow \tilde{I}_{j,l,\mathbf{u};T_i} \right)^2 > \sum_{j,l,\mathbf{u}} \left(\tilde{I}_{j,l,\mathbf{u};T2} \Leftrightarrow \tilde{I}_{j,l,\mathbf{u};T_i} \right)^2 .$$

An equivalent approach is adopted using the squared detail coefficients of an unsmoothed, Haar non-decimated wavelet transform of the realisations.

The results of this experiment are displayed in Table 4.9. Recall that in the examples given in Sections 4.6.1 and 4.6.2 the LS2W and (uncorrected) NDWT approaches yielded similar classification rates. This was due to the original images being stationary. In this case however, the difference between the two approaches becomes quite apparent, with the NDWT method only able to classify 62% of tiles correctly. The reason for this is that the inherent redundancy of the NDWT causes

Method	Percentage Correctly Classified
LS2W model	100
Uncorrected NDWT	62

Table 4.9: Percentage of tile types classified correctly using the LS2W and NDWT-based classification approaches.

power to leak across directions and into lower scales, thus making discrimination between the two tile types on the basis of their detail coefficients difficult. The LS2W approach corrects for this leakage and therefore attains a higher classification rate.

4.7 Summary

The application of the LS2W modelling approach to texture analysis has been considered in this chapter. Having reviewed recent wavelet-based texture measures, we proposed and investigated a direction/scale measure of texture power based on the LWP proposed in Chapter 3. This measure was compared, with varying degrees of success, against alternative texture measures (non-decimated wavelet transform, discrete wavelet transform and Fourier) for various texture problems. In the case of the standard Brodatz textures, the LS2W-approach fared well.

We then proceeded to consider more realistic problems. Discrimination between the various Pill and Hair images was good. However, when attempting to classify sample textures obtained from these real-world problems, no one approach excelled. As one would expect with *stationary* images, the LS2W model and uncorrected non-decimated wavelet transform features obtained comparable results. However, when considering the application of these approaches to non-stationary texture classification, the uncorrected NDWT approach was inferior to the LS2W model — the higher classification rates of the latter being due to its ability to correct the power leakage which is induced by the redundancy of the NDWT.

Chapter 5

The inner product matrix of discrete autocorrelation wavelets

5.1 Introduction

This chapter considers the construction of the inner product matrix of discrete autocorrelation wavelets. Such wavelets have recently been used in the statistical analysis of locally-stationary time series, consisting of N ($= 2^J$) data points, which possess a variance structure which changes slowly over time (see Nason *et al.* (2000) for example). Broadly, this means that on close-range inspection, such a series would appear to be stationary (see Nason & von Sachs 1999). Thus if one can collect sufficient information in the region of local stationarity, one can obtain a sensible estimate of their statistical properties.

Direct, brute force, construction of discrete autocorrelation wavelets proves computationally taxing when dealing with large datasets. However a well-known recursion, which we discuss in Section 5.2, permits an economic construction of the autocorrelation wavelets. The order of the recursion is $\mathcal{O}(2^J)$, compared with $\mathcal{O}(2^{2J})$ when using the brute-force approach. This chapter shows that the autocorrelation wavelet recursion scheme can be exploited to derive an efficient scheme for the construction of the inner product matrix of the autocorrelation wavelets, rather than adopt the slow brute-force computations used by Nason *et al.* (2000).

The method proposed in Section 5.3 relates neighbouring elements of the inner product matrix lying on a given diagonal (figure 5.1 illustrates our scheme). A recursive

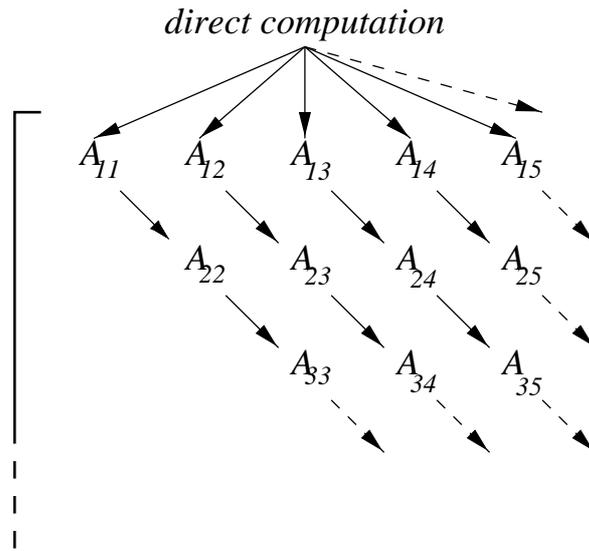


Figure 5.1: Recursive scheme for (symmetric) inner product matrix A calculation. All diagonal elements are obtained recursively: $A_{j+1,k+1}$ from $A_{j,k}$. The top row is populated through direct computation.

scheme for the calculation of those (initialising) elements which lie upon the top row of the inner product matrix is proposed in Section 5.4. The computational efficiency of the various schemes considered in Section 5.2 and 5.3 are compared to the brute-force alternatives in Table 5.1. Finally, motivated by the work of Chapters 3 and 4, we propose a recursive construction for the inner product matrix of (separable) two-dimensional discrete autocorrelation wavelets. Proofs are presented in Appendix B.

	Brute force	Recursive
Autocorrelation wavelets	2^{2J} (N^2)	2^J (N)
Inner Product matrix	$J2^J$ ($N \log N$)	J^3 ($\log N$) ³

Table 5.1: Order of computations required.

5.2 Autocorrelation wavelets and time series

From our review of the work of Nason *et al.* (2000) in Section 2.7, it is evident that the inner product matrix of autocorrelation wavelets plays an important role in the

estimation of EWS. More precisely, the corrected EWS estimator is *asymptotically* unbiased. Thus it is desirable to analyse the longest possible time series. Such series require the use of larger inner product matrices. Hence this chapter focuses on the development of an efficient scheme for the generation of such matrices.

We begin by highlighting two useful recursive properties of autocorrelation wavelets which permit an efficient construction scheme. The first property shows the relationship between the finest scale autocorrelation wavelets and the discrete autocorrelation father wavelet:

Property 5.1

Using equation 5.1.34 of Daubechies (1992), it is easily shown that the discrete autocorrelation wavelets at scale 1 are related to the discrete autocorrelation father wavelets via the relationship $\Psi_1(\tau) = (\Leftrightarrow 1)^\tau \Phi_1(\tau)$.

The next property illustrates a two-scale relationship between autocorrelation wavelets. This permits a more efficient method for discrete autocorrelation wavelet computation than the brute force application of (2.34).

Property 5.2

Let $\tau \in \mathbb{Z}$. Then the discrete autocorrelation wavelet at scale $j + 1$ is related to that at scale $j \in \mathbb{N}$ by the following:

$$\Psi_{j+1}(2\tau) = \Psi_j(\tau) \tag{5.1}$$

and

$$\Psi_{j+1}(2\tau + 1) = \sum_{p=\max\{\frac{-L_1}{2}, 1-L_j+\tau\}}^{\min\{\frac{L_1}{2}-1, L_j+\tau-1\}} \Phi_1(2p + 1) \Psi_j(\tau \Leftrightarrow p). \tag{5.2}$$

In other words, the autocorrelation wavelet at any scale can be recursively obtained from the autocorrelation wavelet at the previous finer scale, using knowledge only of Φ_1 , the scale $j = 1$ autocorrelation father wavelet. A similar two-scale scheme can also be found for discrete autocorrelation *father* wavelets.

It is easily shown that, whereas construction of the complete set of $\{\Psi_j(\tau)\}_{j=1, \dots, J}$ takes $\mathcal{O}(2^{2J})$ operations via a brute force approach, one can construct the same family in $\mathcal{O}(2^J)$ operations using Property 5.2 (see Eckley & Nason (2000) for further details).

5.2.1 Inner product matrix: brute force construction

Recall from Definition 2.7 that Nason *et al.* (2000) define the J -dimensional inner product matrix of autocorrelation wavelets to be given by

$$A_J = (A_{j,k})_{j,k \in \{1, \dots, J\}}$$

where

$$A_{j,k} = \sum_{\tau=1-\min\{L_j, L_k\}}^{\min\{L_j, L_k\}-1} \Psi_j(\tau) \Psi_k(\tau).$$

It can be shown that for all $j \in \mathbb{N}$, $\Psi_j(0) = 1$. Using this identity, together with the symmetry of the discrete autocorrelation wavelets, it is easily seen that:

$$A_{j,k} = 1 + 2 \sum_{\tau=1}^{\min\{L_j, L_k\}-1} \Psi_j(\tau) \Psi_k(\tau). \quad (5.3)$$

Consequently, for any suitable values of j and k , direct computation of the inner product, $A_{j,k}$, takes $\min\{L_j, L_k\} + 1$ operations, given that $\Psi_j(\tau)$ and $\Psi_k(\tau)$ have already been evaluated. However, as A_J is symmetric, only its upper (or lower) triangle need be computed. Hence, for $k \geq j$, computation of any given element in this upper triangle requires

$$\min\{L_j, L_k\} + 1 = L_j + 1 \quad \text{operations.}$$

Thus, brute force construction of A_J takes

$$\sum_{j=1}^J \sum_{l=j}^J (L_j + 1),$$

i.e. $\mathcal{O}(J^2)$ operations.

5.3 Inner product matrix: recursive construction

The previous section motivated the need for an efficient method of constructing discrete autocorrelation wavelet inner product matrices. A natural approach to finding such a method would be to apply the results of Property 5.2 to find recursive structures within such a matrix. This is the approach adopted below. We begin by obtaining a relationship which relates elements of the inner product matrix which lie upon the

leading diagonal. This helps motivate the work of Section 5.3.3, in which we derive an analogous relationship between neighbouring elements on any given diagonal. Sections 5.3.2 and 5.3.4 consider the computational efficiency for the leading and other diagonals respectively.

5.3.1 The leading diagonal

This section obtains a relationship which connects neighbouring elements of the leading diagonal. In particular, it enables efficient computation of $A_{k,k}$ from $A_{1,1}$ for any $k \in \mathbb{N}$. Some of the ideas in this section and leading to Proposition 5.4 are joint work with Dr. G. P. Nason. We start by defining a few key quantities:

Definition 5.1

For $r \in \mathbb{Z}$, let $l_r = \Leftrightarrow L_1/2 \Leftrightarrow \min\{0, r\}$ and $u_r = L_1/2 \Leftrightarrow 1 \Leftrightarrow \max\{0, r\}$. Then we define

$$Q_r = \sum_{p=l_r}^{u_r} \Phi_1(2p+1)\Phi_1(2(p+r)+1). \quad (5.4)$$

It is easily shown that Q_r is symmetric about $r = 0$ and has support $[1 \Leftrightarrow L_1, L_1 \Leftrightarrow 1]$. As a consequence of this symmetry, we need only evaluate $\{Q_r\}_{r \in \{0, \dots, L_1-1\}}$. Evaluation of this set takes

$$\text{Ops}(Q) = \frac{L_1}{2}(L_1 + 1) \text{ operations.}$$

The second quantity is based upon the autocorrelation wavelets at level $j \in \mathbb{N}$.

Definition 5.2

Let $j \in \mathbb{N}$, $l_n = 1 \Leftrightarrow L_j + \max\{0, n\}$ and $u_n = L_j \Leftrightarrow 1 + \min\{0, n\}$. Then define

$$P_{j,n} = \sum_{k=l_n}^{u_n} \Psi_j(k)\Psi_j(k \Leftrightarrow n). \quad (5.5)$$

Property 5.3

Clearly $P_{j,n}$ is symmetric about $n = 0$ and has support $[2(1 \Leftrightarrow L_j), \dots, 2(L_j \Leftrightarrow 1)]$.

Additionally $P_{j,0} = A_{j,j}$.

Using the two-scale relationship of discrete autocorrelation wavelets, we can recursively construct the $P_{j,n}$ as follows:

Proposition 5.1

Let $p \in \mathbb{Z}$ and $j \in \mathbb{N}$ and let

$$\begin{aligned} u_e &= \min\{L_1 \Leftrightarrow 1, 2(L_{j-1} \Leftrightarrow 1) \Leftrightarrow p\}, & l_e &= \max\{1 \Leftrightarrow L_1, 2(1 \Leftrightarrow L_{j-1}) \Leftrightarrow p\}, \\ u_{o1} &= \min\left\{\frac{L_1}{2} \Leftrightarrow 1, 2(L_{j-1} \Leftrightarrow 1) \Leftrightarrow p\right\}, & l_{o1} &= \max\left\{\Leftrightarrow \frac{L_1}{2}, 2(1 \Leftrightarrow L_{j-1}) \Leftrightarrow p\right\}, \\ u_{o2} &= \min\left\{\frac{L_1}{2} \Leftrightarrow 1, p \Leftrightarrow 2(1 \Leftrightarrow L_{j-1})\right\} & \text{and } l_{o2} &= \max\left\{\Leftrightarrow \frac{L_1}{2}, p \Leftrightarrow 2(L_{j-1} \Leftrightarrow 1)\right\}. \end{aligned}$$

Then,

$$P_{j,2p} = P_{j-1,p} + \sum_{q=l_e}^{u_e} P_{j-1,p+q} Q_q \quad (5.6)$$

and

$$P_{j,2p+1} = \sum_{r=l_{o1}}^{u_{o1}} \Phi_1(2r+1) P_{j-1,p+r} + \sum_{r=l_{o2}}^{u_{o2}} \Phi_1(2r+1) P_{j-1,p-r}. \quad (5.7)$$

Thus the $P_{j,n}$ may be calculated using knowledge of only Φ_1 and $P_{k,n}$ at finer scales k .

The results of Property 5.2, together with the above identities, permits the derivation of a recursive relationship between neighbouring elements which lie along the leading diagonal of the inner product matrix.

Proposition 5.2

Let $j \in \mathbb{N}$. Then the $(j+1, j+1)^{th}$ element of the inner product matrix is related to the $(j, j)^{th}$ element by the following recursive relation:

$$\begin{aligned} A_{j+1,j+1} &= A_{j,j} + \sum_{r=1-L_1}^{L_1-1} P_{j,r} Q_r \\ &= A_{j,j}(1 + Q_0) + 2 \sum_{r=1}^{L_1-1} P_{j,r} Q_r. \end{aligned} \quad (5.8)$$

In other words, the elements which lie on the leading diagonal of the inner product matrix, A_J , can be recursively obtained using only knowledge of $A_{1,1}$ and Φ_1 .

5.3.2 Computational effort of calculating the leading diagonal recursively

Initialising values

We start our analysis of the computational effort required to calculate the leading diagonal using the scheme outlined in Section 5.3.1 by considering the cost of computing

the initialising values $\{P_{1,n}\}$. By definition,

$$P_{1,n} = \sum_{k=1-L_1+\max\{0,n\}}^{L_1-1+\min\{0,n\}} \Psi_1(k)\Psi_1(k \Leftrightarrow n). \quad (5.9)$$

However, as $P_{1,n}$ is an even function in n , it suffices to calculate it for $n \in [0, \dots, 2(L_1 \Leftrightarrow 1)]$. Thus, from (5.9), it follows that direct evaluation of the $\{P_{1,n}\}_{n \in [0, \dots, 2(L_1-1)]}$, using pre-computed values of $\{\Psi_j(\tau)\}$, takes

$$\begin{aligned} \text{Ops}(P_{\text{init}}) &= \sum_{n=0}^{2(L_1-1)} 2(L_1 \Leftrightarrow 1) \Leftrightarrow n; \\ &= (L_1 \Leftrightarrow 1)(2L_1 \Leftrightarrow 1) \quad \text{operations.} \end{aligned}$$

In other words, for any given wavelet family, it is an $\mathcal{O}(1)$ operation.

Evaluation of the $P_{j,2p}$

Recall from Proposition 5.1 that

$$P_{j,2p} = P_{j-1,p} + \sum_{q=l_e}^{u_e} P_{j-1,p+q} Q_q. \quad (5.10)$$

Moreover, by Property 5.3, $P_{j,2p}$ is symmetric about $2p = 0$. It therefore suffices to consider the evaluation of $P_{j,2p}$ for $p > 0$.

The construction of (5.10) is such that to calculate $P_{j,2p}$, for any given $j \in \mathbb{N}, p \in \mathbb{Z}$, one must perform the following number of operations:

$$\begin{aligned} \text{Ops}(P_{j,2p}) &= 1 + u_e \Leftrightarrow l_e, \\ &= 1 + \min\{L_1 \Leftrightarrow 1, 2(L_{j-1} \Leftrightarrow 1) \Leftrightarrow p\} \\ &\quad \Leftrightarrow \max\{1 \Leftrightarrow L_1, 2(1 \Leftrightarrow L_{j-1}) \Leftrightarrow p\}. \end{aligned} \quad (5.11)$$

However, as $p > 0$, it follows that $1 \Leftrightarrow L_1$ is always greater than $2(1 \Leftrightarrow L_{j-1}) \Leftrightarrow p$, for $j \in \mathbb{N} \setminus \{1\}$. Hence equation (5.11) can be simplified somewhat. However, it is important to note that the minimum term cannot be simplified, as there exist $p \in \mathbb{N}$ such that $2(L_{j-1} \Leftrightarrow 1) \Leftrightarrow p < L_1 \Leftrightarrow 1$. It therefore follows that the number of operations required to calculate $P_{j,2p}$ from $P_{j-1,\cdot}$, for $j, p \in \mathbb{N}$, is given by

$$\text{Ops}(P_{j,2p}) = L_1 + \min\{L_1 \Leftrightarrow 1, 2(L_{j-1} \Leftrightarrow 1) \Leftrightarrow p\}. \quad (5.12)$$

We now consider the values of p for which we wish to evaluate $P_{j,2p}$, for any given $j \in \mathbb{N}$. The recursive identity of Proposition 5.2,

$$A_{j+1,j+1} = A_{j,j}(1 + Q_0) + 2 \sum_{r=1}^{L_1-1} P_{j,r} Q_r, \quad (5.13)$$

requires only those values of $P_{j,r}$ such that $r \in [1, \dots, L_1 \Leftrightarrow 1]$. We can therefore conclude that the length of the filter associated with the wavelet determines the number of $P_{j,r}$ which need to be evaluated at any given level. Thus at first glance, it appears reasonable simply to calculate $P_{j,2p}$ for all $2p \in [1, L_1 \Leftrightarrow 1]$ for all levels $j' < j$. However, as we demonstrate in Example 5.1, the recursive form of (5.6) used to generate the $\{P_{j,2p}\}$ ensures that this is *not* possible.

Example 5.1 *Assume that $J \in \mathbb{N}$ is fixed and that all $\{P_{j,r}\}_{j=1,\dots,J-1}$ which are required for the construction of $\{P_{J,2p}\}$ have already been evaluated. As J is fixed, we know from (5.13) that it suffices to calculate $P_{J,2p}$ for $2p \in [1, \dots, L_1 \Leftrightarrow 1]$. However, for Daubechies' compactly supported wavelets, L_1 is even. Hence it suffices to calculate $P_{J,2p}$ for $p \in [1, \dots, L_1/2 \Leftrightarrow 1]$. By Proposition 5.1,*

$$P_{J,2p} = P_{J-1,p} + \sum_{q=l_e+p}^{u_e+p} P_{J-1,q} Q_{q-p}. \quad (5.14)$$

In other words, we need to know $P_{J-1,q}$ for

$$1 \Leftrightarrow L_1 + p \leq q \leq \min\{L_1 \Leftrightarrow 1, 2(L_{J-1} \Leftrightarrow 1) \Leftrightarrow p\} + p. \quad (5.15)$$

However,

$$\begin{aligned} \min_{p \in \{1, \dots, L_1/2-1\}} \{L_1 \Leftrightarrow 1, 2(L_{J-1} \Leftrightarrow 1) \Leftrightarrow p\} &= \min\{L_1 \Leftrightarrow 1, 2L_{J-1} \Leftrightarrow L_1/2 \Leftrightarrow 1\}; \\ &= L_1 \Leftrightarrow 1. \end{aligned}$$

Hence (5.15) reduces to:

$$1 \Leftrightarrow L_1 + p \leq q \leq L_1 \Leftrightarrow 1 + p, \quad (5.16)$$

for $p \in \{1, \dots, L_1/2 \Leftrightarrow 1\}$. Thus, we conclude that to calculate $\{P_{J,2p}\}_{1,\dots,L_1/2-1}$, $P_{J-1,q}$ is required for $q \in [0, \dots, L_1 + L_1/2 \Leftrightarrow 2]$.

Clearly, wider and wider "intervals" of $P_{j,r}$ will be required as j decreases. However, it is important to note that one cannot construct an algorithm for the evaluation of

$\{P_{j,2p}\}_{j=1,\dots,J}$ for *all* Daubechies' compactly supported wavelets. Consider, for example, the situation at the end of Example 5.1: to construct $P_{J,2p}$ for $p \in \{1, \dots, L_1/2 \Leftrightarrow 1\}$, we need to be able to evaluate $P_{J-1,r}$ for $r \in \{0, \dots, L_1 + L_1/2 \Leftrightarrow 2\}$. However, as L_1 is even, $L_1 + L_1/2 \Leftrightarrow 2$ can be either odd *or* even valued, depending on the form of N_h .

In other words, if we want to know the precise number of $P_{J-1,2p}$ to evaluate, we must consider each wavelet family individually. Already, this approach seems unappealing. However, the situation is actually more complicated than this, for we also need to consider the situation for each scale (see Eckley & Nason (2000) for further details). Thus, we may conclude that if the algorithm is to be constructed such that only the *required* $\{P_{j,r}\}$ are evaluated not only do we need to construct routines for each individual wavelet family, but the situation for each individual scale must also be considered. This is most unappealing from an implementational perspective.

An alternative to deriving an algorithm for each individual wavelet family, is to devise an algorithm which, though maybe not as computationally efficient, can be used for *any* Daubechies wavelet. We propose such an algorithm below. Although this approach evaluates a slightly larger number of $P_{j,r}$ than is actually required, it is both easy to implement and, more importantly, is still efficient.

1. Fix $J \in \mathbb{N}$.
2. Calculate $P_{1,r}$ for all $r \in [0, \dots, 2(L_1 \Leftrightarrow 1)]$.
3. Then, for $j \in 2, \dots, J$, calculate $P_{j,r}$ for $r \in I_j$, where

$$I_j = \{0, \dots, \min\{2(L_j \Leftrightarrow 1), (J \Leftrightarrow j + 1)L_1\}\}, \quad (5.17)$$

setting $P_{j,r} = 0$ for any values of r such that $2(L_j \Leftrightarrow 1) < r \leq (J \Leftrightarrow j + 1)L_1$.

Adopting the above procedure permits the evaluation of a tractable upper bound for the number of operations required no matter what the choice of wavelet. Moreover, as $P_{j,r}$ is evaluated for values of $r \in I_j$, we know that we need only evaluate $\{P_{j,2p}\}$ for

$$p \in I_{j,\text{even}} = \{0, \dots, u_j \equiv \min\{L_j \Leftrightarrow 1, (J \Leftrightarrow j + 1)L_1/2\}\}.$$

Note that in general, u_j cannot be simplified to $(J \Leftrightarrow j + 1)L_1/2$, for if J is large whilst j is small, then $u_j = L_j \Leftrightarrow 1$. However, using the definition of L_j , it is easy to show

that the following holds:

$$u_j = (J \Leftrightarrow j + 1)L_1/2 \quad \text{iff} \quad \frac{N_h}{N_h \Leftrightarrow 1} \leq 2 \frac{2^j \Leftrightarrow 1}{(J \Leftrightarrow j + 1)}. \quad (5.18)$$

From the work of Eckley & Nason (2000), we know that

$$\text{Ops}\{P_{\text{even}}\} = \sum_{j=2}^J \sum_{p=0}^{u_j} \text{Ops}(P_{j,2p})$$

is, at the very most, an $\mathcal{O}(J^2)$ operation.

Evaluation of the $P_{j,2p+1}$

The situation for the calculation of the $P_{j,2p+1}$ is almost exactly the same as that discussed in the previous section. Following the same logic, it is clear that if we are to evaluate $\{P_{j,2p+1}\}$ for $2p+1 \in \{1, \dots, L_1\}$, then wider and wider intervals of $P_{j-1,r}$ are required as j decreases. Again, the width of any one of these intervals depends on the form of L_1 , the upper boundary being either odd or even. Thus, to obtain an upper bound on the number of operations required to calculate the relevant $\{P_{j,2p+1}\}$, we adopt the procedure outlined above, calculating $P_{j,r}$ for

$$r \in I_{\text{odd}} = \{0, \dots, \min \{2(L_j \Leftrightarrow 1), (J \Leftrightarrow j + 1)(L_1)\}\},$$

setting $P_{j,r} = 0$ for any values of r such that $2(L_j \Leftrightarrow 1) < r \leq (J \Leftrightarrow j + 1)(L_1)$. Note that as it is assumed that $r = 2p + 1 \in I$, it suffices to calculate the $\{P_{j,2p+1}\}$ for $p \in \{0, \dots, u_{o,j}\}$, where

$$u_{o,j} = \min \left\{ \frac{L_1}{2}(J \Leftrightarrow j + 1) \Leftrightarrow 1, L_j \Leftrightarrow 2 \right\}.$$

It can be shown that calculation of the $\{P_{j,2p+1}\}$ required by this algorithm takes, at most, $\text{Ops}(P_{\text{odd}}) = \mathcal{O}(J^2)$ operations (see Eckley & Nason (2000)) for further details.

Calculation of the leading diagonal

Having proposed algorithms for the calculation of the key identities and investigated the computational expense of using such algorithms, we are now in a position to consider the number of operations required to calculate the leading diagonal of a given inner product matrix of discrete autocorrelation wavelets.

Suppose that $A_{1,1}$, $\{P_{j,r}\}$ and the $\{Q_r\}$ have already been calculated. Then from (5.8), it follows that the calculation of $A_{j+1,j+1}$, for any level $j \in 1, \dots, J \Leftrightarrow 1$, takes L_1 operations. Thus the construction of the leading diagonal of the inner product matrix via the schemes proposed in Section 5.3.1 takes

$$\begin{aligned} \text{Ops(Leading Diagonal)} &= \text{Ops}(A_{1,1}) + \text{Ops}(A_{j+1,j+1})_{j=1,\dots,J-1} + \text{Ops}(P_{\text{odd}}) \\ &\quad + \text{Ops}(P_{\text{even}}) + \text{Ops}(P_{\text{init}}) + \text{Ops}(Q) \\ &= JL_1 + 1 + \text{Ops}(P_{\text{odd}}) + \text{Ops}(P_{\text{even}}) + \text{Ops}(P_{\text{init}}) + \text{Ops}(Q) \end{aligned}$$

operations. In other words it is, at most, an $\mathcal{O}(J^2)$ operation. In contrast, the brute force approach adopted by Nason *et al.* would take $\mathcal{O}(2^J)$ operations.

5.3.3 General diagonal case

The results of Section 5.3.1, particularly Proposition 5.2, suggest that a recursive relationship may exist for those entries which lie on other diagonals. Prior to establishing such a recursion, we define the following identities.

Definition 5.3

Let $j, k \in \mathbb{N}$ with $k \geq j$, $l_r = \max\{1 \Leftrightarrow L_j, 1 \Leftrightarrow L_k + r\}$ and $u_r = \min\{L_j \Leftrightarrow 1, L_k + r \Leftrightarrow 1\}$. Furthermore, suppose that $r \in \mathbb{Z}$. Then define,

$$T_{j,k,r} = \sum_{l=l_r}^{u_r} \Psi_j(l)\Psi_k(l \Leftrightarrow r). \quad (5.19)$$

$T_{j,k,r}$ plays an analogous role here to that of $P_{j,r}$ in the leading diagonal case. However $T_{j,k,r}$ does not (generally) share the properties as $P_{j,r}$. For example, the support of $T_{j,k,r}$ is $[2 \Leftrightarrow L_j \Leftrightarrow L_k, L_k + L_j \Leftrightarrow 2]$. Furthermore, $T_{j,k,r}$ is not usually symmetric in r , though it is easily shown that

$$T_{j,k,-r} = T_{k,j,r}. \quad (5.20)$$

As one might expect, a special case exists when $k = j$, for $T_{j,j,r} = P_{j,r}$ and is consequently symmetric in r by Property 5.3. The following proposition establishes an efficient, recursive, approach for the construction of the $T_{j,k,r}$.

Proposition 5.3

Suppose that $j, k \in \mathbb{N}$ with $k \geq j$, $j \neq 1$, $p \in \mathbb{Z}$ and set

$$\begin{aligned} l_{e,p} &= \max \{1 \Leftrightarrow L_1, 2 \Leftrightarrow L_{j-1} \Leftrightarrow L_{k-1} \Leftrightarrow p\}, \\ u_{e,p} &= \min \{L_1 \Leftrightarrow 1, L_{k-1} + L_{j-1} \Leftrightarrow 2 \Leftrightarrow p\}, \\ l_{o1,p} &= \max \left\{ \Leftrightarrow \frac{L_1}{2}, 2 \Leftrightarrow L_{j-1} \Leftrightarrow L_{k-1} \Leftrightarrow p \right\}, \\ u_{o1,p} &= \min \left\{ \frac{L_1}{2} \Leftrightarrow 1, L_{k-1} + L_{j-1} \Leftrightarrow 2 \Leftrightarrow p \right\}, \\ l_{o2,p} &= \max \left\{ \Leftrightarrow \frac{L_1}{2}, 2 + p \Leftrightarrow L_{j-1} \Leftrightarrow L_{k-1} \right\} \\ \text{and } u_{o2,p} &= \min \left\{ \frac{L_1}{2} \Leftrightarrow 1, L_{j-1} + L_{k-1} + p \Leftrightarrow 2 \right\}. \end{aligned}$$

Then it can be shown that

$$T_{j,k,2p} = T_{j-1,k-1,p} + \sum_{m=l_{e,p}}^{u_{e,p}} T_{j-1,k-1,p+m} Q_m \quad (5.21)$$

and

$$T_{j,k,(2p+1)} = \sum_{r=l_{o1,p}}^{u_{o1,p}} \Phi_1(2r+1) T_{j-1,k-1,p+r} + \sum_{r=l_{o2,p}}^{u_{o2,p}} \Phi_1(2r+1) T_{j-1,k-1,p-r}. \quad (5.22)$$

Combining knowledge of the $\{T_{j,k,r}\}$, together with the $\{Q_r\}$ and the top row of the inner product matrix, the following recursive algorithm for the calculation of those elements lying on the diagonals of the inner product matrix may be derived.

Proposition 5.4

Let $k \geq j$. Then using Q_r and $T_{j,k,r}$ as defined above, the following recursive scheme can be derived to calculate those elements which lie on a diagonal of the inner product matrix A :

$$A_{j,k} = A_{j-1,k-1} + \sum_{r=1-L_1}^{L_1-1} T_{j-1,k-1,r} Q_r. \quad (5.23)$$

5.3.4 Efficiency of adopting the recursive approach for other diagonals

By reasoning in a manner similar to that of Section 5.3.2, it becomes evident that if one wishes to develop an algorithm which evaluates *only* those $\{T_{j,k,r}\}$ required by the recursion proposed in Proposition 5.4, then the algorithm must

- a) consider each wavelet family separately,
- b) and consider the situation for each scale individually.

Such an algorithm seems both inelegant and laborious to implement. Thus we propose an alternative approach which, although evaluates a slightly larger number of $T_{j,k,r}$ than is actually required, proves much simpler to implement.

Details of the proposed algorithm are given below, together with an analysis of the computational expense. Note that as the recursive construction of the $A_{j,j}$ has already been discussed in length and as we only need calculate $A_{j,k}$ for $k \geq j$, we will assume throughout that $k > j$. We commence by finding the number of computations required to calculate the initialising $\{T_{j,k,r}\}$. Then we consider the number of operations required to calculate the $T_{j,k,r}$ for even and odd-values of r respectively.

Calculation of the initialising values

A natural place to start considering the efficiency of this recursive approach is to evaluate the number of computations required to enumerate the $\{T_{j,k,r}\}$ when $j = 1$. In this case, the defining equation, (5.19), reduces to

$$T_{1,k,r} = \sum_{l=\max\{1-L_1, 1-L_k+r\}}^{\min\{L_1-1, L_k+r-1\}} \Psi_1(l)\Psi_k(l \Leftrightarrow r), \quad (5.24)$$

This is compactly supported on $[2 \Leftrightarrow L_1 \Leftrightarrow L_k, \dots, L_1 + L_k \Leftrightarrow 2]$.

The $\{T_{1,k,r}\}$ can be calculated directly, using the values of $\Psi_j(\tau)$ obtained from the recursion scheme detailed in Proposition 5.2. However, following arguments similar to those proposed in Section 5.3.2, it is evident that a complicated, case by case algorithm is required for the evaluation of the precise number of $\{T_{1,k,r}\}$ required for the recursive construction of the $\{A_{j,k}\}$. A much simpler algorithm may be developed if we are willing to evaluate a slightly larger number of $\{T_{1,k,r}\}$ than that which is required for the construction of A_J . This algorithm possesses the additional benefit of being suitable for all wavelet families. An outline of the algorithm is provided below. Note how the width of the interval is dependent on k — this is a consequence of the assumption that $k > j$.

1. Fix $J \in \mathbb{N}$.

2. For $k \in \mathbb{N} \setminus 1$, set

$$l_{\text{init},k} = \max\{\Leftrightarrow L_1(J \Leftrightarrow k + 1), 2 \Leftrightarrow L_1 \Leftrightarrow L_k\}$$

and

$$u_{\text{init},k} = \min\{L_1(J \Leftrightarrow k + 1), L_1 + L_k \Leftrightarrow 2\}.$$

3. Calculate $T_{1,k,r}$ for all

$$r \in I_k = [l_{\text{init},k}, \dots, u_{\text{init},k}],$$

setting $T_{1,k,r} = 0$ for any values of r such that $L_1 + L_k \Leftrightarrow 2 \leq r \leq L_1(J \Leftrightarrow k + 1)$
or $\Leftrightarrow L_1(J \Leftrightarrow k + 1) \leq r \leq 2 \Leftrightarrow L_1 \Leftrightarrow L_k$.

It can be shown that construction of the initialising $T_{1,k,r}$ via the above scheme takes at most $\text{Ops}(T_{\text{init}}) = \mathcal{O}(J^2)$ operations (see Eckley & Nason (2000) for further details).

Evaluation of the $T_{j,k,2p}$

Recall from Proposition 5.3, that

$$T_{j,k,2p} = T_{j-1,k-1,p} + \sum_{m=l_{e,p}}^{u_{e,p}} T_{j-1,k-1,p+m} Q_m. \quad (5.25)$$

Assuming that all relevant $\{T_{j-1,k-1,r}\}$ have been evaluated, it follows from (5.25) that evaluation of $T_{j,k,2p}$ for any given $j, k, \in \mathbb{N}$ and $p \in \mathbb{Z}$ takes

$$\text{Ops}(T_{j,k,2p}) = 1 + u_{e,p} \Leftrightarrow l_{e,p} \quad \text{operations.} \quad (5.26)$$

As in earlier sections, it is important to observe that for any given $j, k \in \mathbb{Z}$, it is not necessarily the case that one must evaluate $T_{j,k,2p}$ for all $2p \in [2 \Leftrightarrow L_j \Leftrightarrow L_k, L_j + L_k \Leftrightarrow 2]$. However, as with the initialising values, $\{T_{1,k,r}\}$, the exact construction would require a cumbersome case by case algorithm. This is both inelegant and complicated to implement. We therefore propose the following algorithmic approach for the construction of the $\{T_{j,k,2p}\}$:

1. Fix $J \in \mathbb{N}$.

2. Then for $j, k \in \mathbb{N} \setminus 1$, set

$$l_{j,k}^e = \max\{\Leftrightarrow L_1(J \Leftrightarrow k + 1), 2 \Leftrightarrow L_j \Leftrightarrow L_k\} \quad (5.27)$$

and

$$u_{j,k}^e = \min\{L_1(J \Leftrightarrow k + 1), L_j \Leftrightarrow L_k \Leftrightarrow 2\}. \quad (5.28)$$

3. Evaluate $T_{j,k,2p}$ for all

$$p \in I_{j,k}^e = \left\{ \frac{l_{j,k}^e}{2}, \dots, \frac{u_{j,k}^e}{2} \right\}.$$

setting $T_{j,k,2p} = 0$ for any $p \in \{\Leftrightarrow L_1(J \Leftrightarrow k + 1), \dots, L_1(J \Leftrightarrow k + 1)\}$ such that $p \notin I_{j,k}^e$.

The advantage of using the above scheme is that it may be used with all wavelet families, although it evaluates a slightly larger number of $T_{j,k,2p}$ than is actually required.

Adopting the above procedure, it follows that the construction of the required $\{T_{j,k,2p}\}$ takes

$$\sum_{p=l_{j,k}^e/2}^{u_{j,k}^e/2} \text{Ops}(T_{j,k,2p}) \quad \text{operations.} \quad (5.29)$$

Thus, the construction of the complete suite of $\{T_{j,k,2p}\}_{\substack{j=2, \dots, J-1 \\ k=j+1, \dots, J}}$ for all relevant p takes

$$\begin{aligned} \text{Ops}(T_{\text{even}}) &= \sum_{j=2}^{J-1} \sum_{k=j+1}^J \sum_{p=l_{j,k}^e/2}^{u_{j,k}^e/2} \text{Ops}(T_{j,k,2p}) = \sum_{j=2}^{J-1} \sum_{k=j+1}^J \sum_{p=l_{j,k}^e/2}^{u_{j,k}^e/2} 1 + u_{e,p} \Leftrightarrow l_{e,p} \\ &\leq \sum_{j=2}^{J-1} \sum_{k=j+1}^J (2L_1 + 1)(u_{j,k}^e/2 \Leftrightarrow l_{j,k}^e/2 + 1) \\ &\leq (2L_1 + 1) \sum_{j=2}^{J-1} \sum_{k=j+1}^J L_1(J \Leftrightarrow k + 1). \end{aligned}$$

In other words it is, at worst, an $\mathcal{O}(J^3)$ operation.

Evaluation of the $T_{j,k,2p+1}$

Recall from Proposition 5.3 that

$$T_{j,k,2p+1} = \sum_{r=l_{o1,p}}^{u_{o1,p}} \Phi_1(2r + 1)T_{j-1,k-1,p+r} + \sum_{r=l_{o2,p}}^{u_{o2,p}} \Phi_1(2r + 1)T_{j-1,k-1,p-r}. \quad (5.30)$$

It therefore follows that the construction of $T_{j,k,2p+1}$ for any $j, k \in \mathbb{N} \setminus 1$ and $2p + 1 \in [2 \Leftrightarrow L_j \Leftrightarrow L_k, L_j + L_k \Leftrightarrow 2]$ takes

$$\text{Ops}(T_{j,k,2p+1}) = u_{o1,p} \Leftrightarrow l_{o1,p} + u_{o2,p} \Leftrightarrow l_{o2,p} + 1 \quad \text{operations.} \quad (5.31)$$

Furthermore, using arguments similar to those given in previous sections, it transpires that the development of an algorithm which calculates precisely those $T_{j,k,2p+1}$ required by the recursion schemes proposed here is both complicated and time-consuming. We therefore propose that the following algorithm, which is valid for all Daubechies' wavelets, be adopted when constructing the $T_{j,k,2p+1}$:

1. Fix $J \in \mathbb{N}$.

2. For $j, k \in \mathbb{N} \setminus 1$, set

$$l_{j,k}^o = \frac{1}{2} \max\{\Leftrightarrow L_1(J \Leftrightarrow k + 1), 2 \Leftrightarrow L_j \Leftrightarrow L_k\} \quad (5.32)$$

$$u_{j,k}^o = \frac{1}{2} \min\{L_1(J \Leftrightarrow k + 1), L_j + L_k \Leftrightarrow 2 \Leftrightarrow 1\}. \quad (5.33)$$

3. Calculate $T_{j,k,2p+1}$ for all

$$2p + 1 \in I_{j,k}^o = [l_{j,k}^o, \dots, u_{j,k}^o],$$

setting $T_{j,k,2p+1} = 0$ for any values of

$$2p + 1 \in [\Leftrightarrow L_1(J \Leftrightarrow k + 1), \dots, L_1(J \Leftrightarrow k + 1)]$$

such that $2p + 1 \notin I_{j,k}^o$.

Adopting the above procedure, it follows that the construction of the required $\{T_{j,k,2p+1}\}$, for all $j, k \in \mathbb{N} \setminus 1$, takes

$$\begin{aligned} \text{Ops}(T_{\text{odd}}) &= \sum_{j=2}^{J-1} \sum_{k=j+1}^J \sum_{p=l_{j,k}^o}^{u_{j,k}^o} \text{Ops}(T_{j,k,2p+1}) \leq \sum_{j=2}^{J-1} \sum_{k=j+1}^J \sum_{p=l_{j,k}^o}^{u_{j,k}^o} (2L_1 \Leftrightarrow 1) \\ &\leq \sum_{j=2}^{J-1} \sum_{k=j+1}^J (2L_1 \Leftrightarrow 1)[L_1(J \Leftrightarrow k + 1) \Leftrightarrow 1]. \end{aligned}$$

In other words, construction of the required $\{T_{j,k,2p+1}\}$ is, at worst, an $\mathcal{O}(J^3)$ operation.

Evaluation of the $A_{j,k}$

We conclude our analysis by considering the number of operations required to form the $A_{j,k}$ using the recursive quantities defined in Section 5.3.3. Assuming that $j \neq 1$, Proposition 5.4 states that

$$A_{j+1,k+1} = A_{j,k} + \sum_{1-L_1}^{L_1-1} T_{j,k,r} Q_r. \quad (5.34)$$

It therefore follows that, given the prior enumeration of the $\{T_{j,k,r}\}$, calculation of any given $\{A_{j,k}\}_{\substack{j=2,\dots,J-1 \\ k=j+1,\dots,J}}$ takes

$$\text{Ops}(A_{j,k}) = 2L_1 \Leftrightarrow 1 \quad \text{operations.}$$

Thus the recursive construction, utilising the schemes outlined over the previous sections, of those elements which lie neither upon the leading diagonal nor upon the first row of the inner product matrix, takes

$$\text{Ops}(\text{Lead Diags}) = \text{Ops}(T_{\text{init}}) + \text{Ops}(T_{\text{even}}) + \text{Ops}(T_{\text{odd}}) + \sum_{j=2}^{J-1} \sum_{k=j+1}^J \text{Ops}(A_{j,k})$$

operations. In other words, it is at worst an $\mathcal{O}(J^3)$ operation. Conversely, direct enumeration of the $\{A_{j,k}\}$ via the brute force approach would take $\mathcal{O}(2^J)$ operations.

One question now remains, namely, how best to calculate the top row of A? Do we have to use a brute force approach, or does an efficient recursive method exist? We address this issue in the next section.

5.4 Constructing the inner product matrix top row

Given the flavour of the work in Section 5.3, it is natural to wonder whether the $\{A_{1,j+1}\}$ can be obtained recursively using knowledge of $A_{1,j}$ and hence seed the first row of the matrix. Furthermore, should such a method exist, is it more efficient than the brute force approach? To date, no recursion involving $A_{1,1}$ has been found. However, a certain form of recursive scheme can be developed which may be applied in the evaluation of the $A_{1,j}$. Such a scheme requires the construction of the following identity.

Definition 5.4

Define

$$R_{j,q}^l = \sum_{r=\max\{\lceil \frac{L_1-q}{2^{j-l+1}} \rceil, 1-L_j\}}^{\min\{\lfloor \frac{L_1-2-q}{2^{j-l+1}} \rfloor, L_j-1\}} \Psi_1(2^{j-l+1}r + q + 1) \Psi_j(r). \quad (5.35)$$

It is possible to show that the $\{R_{j,q}^l\}$ may be evaluated recursively via the following proposition.

Proposition 5.5

$R_{j,q}^l$ has the following recursive form:

$$R_{j,q}^l = R_{j-1,q}^{l-2} + \sum_p \Phi_1(2p+1)R_{j-1,2^{j-l+1}+2^{j-l+2}p+q}^{l-2}. \quad (5.36)$$

Furthermore, it can be shown that the support of $R_{j,q}^l$ is given by

$$[\Leftrightarrow L_1 + (1 \Leftrightarrow L_j)2^{j-l+1}, L_1 + (L_j \Leftrightarrow 1)2^{j-l+1} \Leftrightarrow 2].$$

Consequently, the support of $R_{j,2p}^j$ is

$$[2(1 \Leftrightarrow L_j) \Leftrightarrow L_1, L_1 + 2(L_j \Leftrightarrow 2)].$$

The identity given in Definition 5.4 may be used in the recursive computation of $A_{1,j+1}$ for $j \in \mathbb{N}$.

Proposition 5.6

The value of $A_{1,j+1}$ at any scale j can be obtained by calculating $R_{j,2p}^j$ recursively and using knowledge of Φ_1 , the scale $j = 1$ father autocorrelation wavelet. More precisely, the relationship can be expressed as follows:

$$A_{1,j+1} = 1 + \sum_{p=-L_1/2}^{L_1/2-1} \Phi_1(2p+1)R_{j,2p}^j. \quad (5.37)$$

We have therefore demonstrated that the top row of the inner product matrix may be constructed via a recursive relation. From this point, we can recursively compute all elements which exist on a given diagonal in the upper triangle of a matrix using the methods of previous sections. However, we must now ask how efficient is this recursive approach?

It is quite easy to see that, as we progress from one scale to the next, the $\{R_{j,q}^l\}$ required by one scale differ from that required by the next. Hence, in effect, we have to re-calculate the $R_{j,q}^l$ for each $A_{1,j}$. Already, this approach is beginning to look unattractive. Indeed, it seems that for the majority of entries along the top row of the inner product matrix, it is more efficient to calculate the $A_{1,j}$ directly via the interpolation rules of Property 5.2. Hence, we advocate that the $\{A_{1,k}\}$ be constructed directly from $\Psi_1(\tau)$ and $\Psi_k(\tau)$ which can be computed using the efficient $\mathcal{O}(2^J)$ algorithm.

From our earlier work, it is easy to see that, given the prior construction of the discrete autocorrelation wavelets, direct calculation of the $A_{1,j}$ for $k \in \{1, \dots, J\}$ would take

$$\sum_{j=1}^J (L_1 \leftrightarrow 1) = J(L_1 \leftrightarrow 1) \quad \text{operations.}$$

5.5 Extension to two-dimensions

Recursion schemes, similar to those outlined in Section 5.3, can be devised for the construction of either of the following inner product matrices:

- i) $B_J = (B_{j,k}) = (\langle \Phi_j, \Phi_k \rangle)$ - the inner product matrix of discrete autocorrelation father wavelets,
- ii) or $C_J = (\langle \Phi_j, \Psi_l \rangle)$;

though clearly, the latter is not symmetric. Furthermore, as we explain below, by taking suitable products of A_J , B_J and C_J , an inner product matrix of discrete (separable) two-dimensional autocorrelation wavelets may be constructed.

Recall from Definition 3.7 that we define the inner product matrix of discrete, two-dimensional autocorrelation wavelets $\{\Psi_j^h, \Psi_j^v, \Psi_j^d\}_{j=1, \dots, J}$ to be the $3J$ -dimensional matrix, D_J , the elements of which are constructed as follows:

$$D_{\eta, \nu} = \langle \Psi_\eta, \Psi_\nu \rangle,$$

where η, ν are coded as follows

$$\eta(j, l) \equiv f(j) + g(l)$$

with

$$f(j) \equiv j \quad \text{and} \quad g(l) \equiv \begin{cases} 0 & \text{when } l = v, \\ J & \text{when } l = h, \\ 2J & \text{when } l = d. \end{cases}$$

The construction of this matrix is such that its elements may be evaluated by using suitable combinations of other inner product matrices — namely A_J , B_J , and C_J . Explicitly, D_{3J} , can be expressed as follows:

$$D_J = \begin{bmatrix} A_J * B_J & C_J * C_J^T & A_J * C_J \\ C_J * C_J^T & A_J * B_J & A_J * C_J \\ A_J * C_J^T & A_J * C_J^T & A_J * A_J \end{bmatrix}$$

where $A * B$ denotes a component-wise multiplication of A and B . Thus recursion schemes of the form given in Section 5.3 can be used in the construction of the inner product matrix of two-dimensional discrete autocorrelation wavelets.

5.6 Concluding remarks

The efficient computation of the discrete autocorrelation wavelets' inner product matrix is vital for the (asymptotically) *unbiased* estimation of the evolutionary wavelet spectra of locally stationary wavelet processes. Collection of longer time series, for better estimation, requires computation of inner product matrices of larger and larger dimension and hence demonstrates the need for efficient methods of computation.

This chapter introduced an efficient recursive scheme for the construction of the inner product matrix of discrete autocorrelation wavelets. Our scheme constructs matrices using $\mathcal{O}(J^3) = \mathcal{O}((\log N)^3)$ operations, in comparison to the brute-force scheme which uses $\mathcal{O}(J2^J) = \mathcal{O}(N \log N)$ operations. If our new algorithm is used in conjunction with the recursive formulae for generating the autocorrelation wavelets, then the recursive methods require $\mathcal{O}(2^J) = \mathcal{O}(N)$ operations in contrast to the expensive $\mathcal{O}(2^{2J}) = \mathcal{O}(N^2)$ operations required by the brute force approach.

Chapter 6

Local autocovariance estimation

This chapter documents initial research that investigates the local autocovariance measure of LSW processes proposed by Nason *et al.* (2000). In Section 6.2 we consider the interpretation of the LACV estimator in terms of *stationary* time series statistics. Then, in Section 6.3, the application of the Haar LACV estimator to zero mean, second order stationary processes is considered. As these results constitute work in progress, we conclude by highlighting various avenues of future research.

6.1 Introduction

Recall from Section 2.7 that Nason *et al.* (2000) proposed the following as a measure of the local autocovariance structure within LSW processes:

$$C(z, \tau) = \sum_{j=1}^{\infty} S_j(z) \Psi_j(\tau), \quad \text{for } \tau \in \mathbb{Z}, z \in (0, 1). \quad (6.1)$$

In practice, we do not observe time series of infinite length. Rather, it is natural that the observed sequence be finite. This leads us to define the following measure:

Definition 6.1

Let $J \in \mathbb{N}$ and $\{X_t\}$ be a LSW process as defined in Definition 2.8. Then the **curtailed local autocovariance** (CLACV) to level J is defined to be

$$C_J(z, \tau) = \sum_{j=1}^J S_j(z) \Psi_j(\tau), \quad \text{for } \tau \in \mathbb{Z}, z \in (0, 1). \quad (6.2)$$

Defining

$$d_{j,k} = \sum_{t=0}^{T-1} X_t \psi_{j,k}(t) \quad (6.3)$$

to be the *empirical wavelet coefficients* of an LSW process, $\{X_t\}$, Nason *et al.* (2000) demonstrated that the vector of *corrected* wavelet periodograms $\mathbf{L}(z) = \{L_{j,[zT]}\}_{j=1,\dots,J}$, where $\mathbf{L}(z) = A_J^{-1}\mathbf{I}(z)$, is an asymptotically unbiased estimator of the EWS. Here $I_{j,[zT]} \equiv |d_{j,[zT]}|^2$. It therefore appears natural to estimate the CLACV by replacing the EWS, $S_j(z)$, in (6.2) by its estimate, $L_j(z)$, as follows:

Definition 6.2

Let $T = 2^J$ for some $J \in \mathbb{N}$ and $\{x_t\}_{t=0,\dots,T-1}$ be a realisation of a real-valued LSW process with associated wavelet family $\psi_{j,k}$. Then the **curtailed local autocovariance estimator** to level J is defined to be

$$\widehat{C}_J(z, \tau) = \sum_{j=1}^J L_j(z) \Psi_j(\tau). \quad (6.4)$$

As the corrected wavelet periodogram is an asymptotically unbiased estimator of the EWS, it follows that $\widehat{C}_J(z, \tau)$ is an asymptotically unbiased estimator of the CLACV. Below we study various analytic properties of the CLACV estimator, attempting to relate this quantity to existing estimators used in time series analysis. However, it should be noted that to obtain a consistent estimate of the LACV structure, this estimator would normally require some form of smoothing.

6.2 Interpreting the CLACV estimator

Recall from traditional time series analysis that the autocovariance function (acvf) of a zero-mean, second order stationary process $\{X_t\}$ is purely a function of the lag τ :

$$\text{Cov}(X_t, X_{t+\tau}) = R(\tau). \quad (6.5)$$

Given a realisation of the process, $\{x_t\}_{t=0,\dots,T-1}$, the acvf may be estimated using

$$\widehat{R}(\tau) = \frac{1}{T} \sum_{t=1}^{T-1-|\tau|} x_t x_{t+|\tau|}. \quad (6.6)$$

Although biased by the additive lag-dependent quantity $\frac{|\tau|}{N}R(\tau)$, this positive semi-definite function has, in general, a smaller mean square error than the alternative

estimator $\widehat{R}^*(\tau) = (N \Leftrightarrow |\tau|)^{-1} \sum_{t=1}^{T-|\tau|} x_t x_{t+|\tau|}$. See Parzen (1961) or Priestley (Section 5.3.3, 1981) for further details.

An interesting approach to interpreting the information contained within the CLACV estimator is to investigate whether this quantity may be expressed in terms of a measure similar to that of equation (6.6). In so doing we can, for example, more easily compute the bias of our estimator. With this approach in mind, we introduce the following identity which provides a measure of the degree of association between X_t and $X_{t+\tau}$ over specific regions of the series.

Definition 6.3

Let $T = 2^J$ for some $J \in \mathbb{N}$ and $\{x_t\}_{t=0, \dots, T-1}$ be a realisation of a LSW process with associated discrete non-decimated wavelet family $\{\psi_{j,k}(s)\}$. Then the lag τ **wavelet autocovariance estimate** (WAE) at location k within scale j , $\widehat{R}_{j,k}^W(\tau)$, is defined to be given by

$$\widehat{R}_{j,k}^W(\tau) = \sum_{t=0}^{T-1-\tau} \psi_{j,k}(t) \psi_{j,k}(t+\tau) x_t x_{t+\tau}. \quad (6.7)$$

The WAEs are a form of “windowed” acvf, the windowing being achieved by the inclusion of compactly supported discrete non-decimated wavelets. With the above definition in place, we may prove the following proposition which demonstrates that squared empirical wavelet coefficients, and hence the CLACV estimator, may be represented as a sum of these localised (stationary) autocovariance estimators.

Proposition 6.1

Let $T = 2^J$ for some $J \in \mathbb{N}$ and set $r = \lfloor zT \rfloor$. Further let $\{x_t\}_{t=0, \dots, T-1}$ denote a realisation of a LSW process based on the discrete non-decimated wavelet family $\{\psi_{j,k}(t)\}$. Then the squared empirical wavelet coefficient, $d_{j,k}^2$, may be expressed as

$$d_{j,k}^2 = \widehat{R}_{j,k}^W(0) + 2 \sum_{\tau=1}^{T-1} \widehat{R}_{j,k}^W(\tau).$$

Hence, the CLACV estimator may be represented as a weighted sum of WAEs:

$$\widehat{C}_J(z, \tau) = \sum_{j=1}^J \Psi_j(\tau) \sum_{l=1}^J A_{j,l}^{-1} \left\{ \widehat{R}_{l,r}^W(0) + 2 \sum_{\tau=1}^{T-1} \widehat{R}_{l,r}^W(\tau) \right\}. \quad (6.8)$$

In light of the above, we can interpret the CLACV estimator as being a weighted sum of “windowed” acvfs. These windows are of various lengths, dependent on the scale parameter, j , and the support of the $\{\psi_{j,k}\}$.

Proof of Proposition 6.1

Using equation (6.3), it follows that

$$\begin{aligned} d_{j,k}^2 &= \left(\sum_{t=0}^{T-1} X_t \psi_{j,k}(t) \right)^2 \\ &= \sum_{s=0}^{T-1} x_s \psi_{j,k}(s) \sum_{t=0}^{T-1} x_t \psi_{j,k}(t). \end{aligned} \quad (6.9)$$

Expanding (6.9) we obtain:

$$\begin{aligned} d_{j,k}^2 &= \sum_{s=0}^{T-1} x_s^2 \psi_{j,k}(s)^2 + 2 \left(\sum_{s=0}^{T-1-1} x_s x_{s+1} \psi_{j,k}(s) \psi_{j,k}(s+1) \right. \\ &\quad \left. + \sum_{s=0}^{T-1-2} x_s x_{s+2} \psi_{j,k}(s) \psi_{j,k}(s+2) + \dots + x_0 x_{T-1} \psi_{j,k}(0) \psi_{j,k}(T \Leftrightarrow 1) \right) \\ &= \sum_{s=0}^{T-1} x_s^2 \psi_{j,k}(s)^2 + 2 \left\{ \sum_{\tau=1}^{T-1} \sum_{s=0}^{T-1-\tau} x_s x_{s+\tau} \psi_{j,k}(s) \psi_{j,k}(s+\tau) \right\} \\ &= \widehat{R}_{j,k}^W(0) + 2 \sum_{\tau=1}^{T-1} \widehat{R}_{j,k}^W(\tau). \end{aligned} \quad (6.10)$$

Recalling that the CLACV estimate is given by

$$\begin{aligned} \widehat{C}_J(z, \tau) &= \sum_{j=1}^J \Psi_j(\tau) L_j(z) \\ &= \sum_{j=1}^J \Psi_j(\tau) \sum_{l=1}^J A_{j,l}^{-1} d_{l,r}^2, \end{aligned} \quad (6.11)$$

On substituting (6.10) into (6.11) we obtain the required result. \square

Proposition 6.1 considers the structure of the CLACV estimator for *any* given LSW process. Although appealing, the interpretation of $\widehat{R}_{j,k}^W(\tau)$ as a “windowed” acvf is not directly accessible to those unfamiliar with wavelets. Hence we provide the following result which focuses on the special case of a locally stationary Haar process. To achieve this, we introduce the following quantities which measure the (stationary) covariance structure on various windows of the series. Clearly these are related to $\widehat{R}_{j,k}^W$, but the precise form of the relation is not important here:

$$\begin{aligned} \widehat{R}_{l,r}^0(\tau) &= \sum_{s=r}^{2^l+r-1-\tau} x_s x_{s+\tau}, & \widehat{R}_{l,r}^1(\tau) &= \sum_{s=r+2^{l-1}-\tau}^{2^{l-1}+r-1} x_s x_{s+\tau}, \\ \widehat{R}_{l,r}^2(\tau) &= \sum_{s=r}^{2^{l-1}+r-1-\tau} x_s x_{s+\tau} \quad \text{and} \quad \widehat{R}_{l,r}^3(\tau) &= \sum_{s=r+2^{l-1}}^{2^l+r-1-\tau} x_s x_{s+\tau}. \end{aligned}$$

Using these quantities we can obtain a direct interpretation of the local autocovariance estimator, \widehat{C}_J , as a weighted sum of windowed versions of the classical acvf estimator.

Proposition 6.2

Let $\{x_t\}_{t=0, \dots, T=2^J-1}$ be a realisation of a LSW process founded upon the Haar wavelet family. Setting $r = \lfloor zT \rfloor$ and assuming that the NDWT is implemented using a periodic boundary condition, then the level J CLACV estimator may be expressed as follows:

$$\begin{aligned} \widehat{C}_J(z, \tau) = & 2^{-1} \sum_{j=1}^J \Psi_j(\tau) A_{j,1}^{-1} (x_{r+1}^2 + x_r^2 \Leftrightarrow 2x_r x_{r+1}) \\ & + \sum_{j=1}^J \sum_{l=2}^J 2^{-l} \Psi_j(\tau) A_{j,l}^{-1} \left\{ \widehat{R}_{l,r}^2(0) + \widehat{R}_{l,r}^3(0) \right. \\ & \left. + 2 \sum_{u=1}^{2^{l-1}-1} \left(\widehat{R}_{l,r}^2(u) + \widehat{R}_{l,r}^3(u) \Leftrightarrow \widehat{R}_{l,r}^1(u) \right) \Leftrightarrow 2 \sum_{u=2^{l-1}}^{2^l-1} \widehat{R}_{l,r}^0(u) \right\}. \end{aligned} \quad (6.12)$$

The above demonstrates that the CLACV estimator of locally stationary Haar processes may be interpreted as a weighted sum of (stationary) windowed acvf estimators. The contributions in equation (6.12) estimate the stationary structure on various windows of the series, these window widths being dependent on the lag, τ , and on scale, l . We can therefore relate the Haar CLACV estimator to the classical acvf estimator.

Proof of Proposition 6.2

In this proof we adopt the form of $d_{l,r}$ as implemented in `WaveThresh`, which we will use later in Section 6.3. This construction differs slightly from equation (6.3), consisting of a sign change and a shift in origin. The sign change is irrelevant as we are dealing with squared detail coefficients. The effect of the shift in origin is also negligible as the NDWT is translation-equivariant. Moreover, the principal motivation for this result is the investigation of the bias of this measure when analysing second-order stationary processes. In this case, the process's covariance structure is location-independent and hence an origin shift will not effect the subsequent analyses.

By (6.11) ,

$$\begin{aligned}\widehat{C}_J(z, \tau) &= \sum_{j=1}^J \sum_{l=1}^J \Psi_j(\tau) A_{j,l}^{-1} d_{l,r}^2, \quad \text{where } r = [zT] \\ &= 2^{-1} \sum_{j=1}^J \Psi_j(\tau) A_{j,1}^{-1} (x_r \Leftrightarrow x_{r+1})^2 + \sum_{j=1}^J \sum_{l=2}^J \Psi_j(\tau) A_{j,l}^{-1} d_{l,r}^2.\end{aligned}$$

Further,

$$\begin{aligned}d_{l,r}^2 &= \frac{1}{2^l} \left(\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \Leftrightarrow \sum_{t=r}^{2^{l-1}-1+r} x_t \right)^2 \\ &= \frac{1}{2^l} \left[\left(\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \right)^2 + \left(\sum_{t=r}^{2^{l-1}-1+r} x_t \right)^2 \Leftrightarrow 2 \sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \sum_{s=r}^{2^{l-1}-1+r} x_s \right]. \quad (6.13)\end{aligned}$$

Consider the last term of (6.13). This double sum extends over a lattice of points on a square. Normally we might consider, for example, summing first over rows and then adding the row sums. However in this case, we sum over *diagonals* and add the *diagonal* sums together:

$$\begin{aligned}\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \sum_{s=r}^{2^{l-1}-1+r} x_s &= \sum_{s=r}^{2^{l-1}+r-1} x_s x_{s+2^{l-1}} + \sum_{s=r}^{2^{l-1}+r-2} x_s x_{s+2^{l-1}+1} \\ &\quad + \sum_{s=r}^{2^{l-1}+r-3} x_s x_{s+2^{l-1}+2} + \dots + x_r x_{r+2^{l-1}} \\ &\quad + \sum_{s=r+1}^{2^{l-1}+r-1} x_s x_{s+2^{l-1}-1} + \sum_{s=r+2}^{2^{l-1}+r-1} x_s x_{s+2^{l-1}-2} \\ &\quad + \dots + x_{2^{l-1}+r-1} x_{2^{l-1}+r}\end{aligned}$$

In other words,

$$\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \sum_{s=r}^{2^{l-1}-1+r} x_s = \sum_{u=0}^{2^{l-1}-1} \sum_{s=r}^{2^{l-1}+r-1-u} x_s x_{s+2^{l-1}+u} + \sum_{u=1}^{2^{l-1}-1} \sum_{s=r+u}^{2^{l-1}+r-1} x_s x_{s+2^{l-1}-u}.$$

The terms on the right hand side of this last expression represent the summations of the upper and lower triangles of the lattice. Setting $m = u + 2^{l-1}$ and $m = u \Leftrightarrow 2^{l-1}$ in the first and second double sums respectively, it follows that

$$\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \sum_{s=r}^{2^{l-1}-1+r} x_s = \sum_{m=2^{l-1}}^{2^l-1} \sum_{s=r}^{2^{l-1}+r-1-m} x_s x_{s+m} + \sum_{m=1-2^{l-1}}^{-1} \sum_{s=r+m+2^{l-1}}^{2^{l-1}+r-1} x_s x_{s-m}.$$

Furthermore, by setting $u = \Leftrightarrow m$ in the second double sum of the above, we obtain

$$\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t \sum_{t=r}^{2^{l-1}-1+r} x_t = \sum_{u=2^{l-1}}^{2^l-1} \widehat{R}_{l,r}^0(u) + \sum_{u=1}^{2^{l-1}-1} \widehat{R}_{l,r}^1(u). \quad (6.14)$$

Next, consider the term $\left(\sum_{t=r}^{2^{l-1}-1+r} x_t\right)^2$. Again, we sum over the diagonals first and then add the diagonal sums:

$$\begin{aligned} \left(\sum_{t=r}^{2^{l-1}-1+r} x_t\right)^2 &= \sum_{s=r}^{2^{l-1}-1+r} x_s^2 + 2 \left\{ \sum_{s=r}^{2^{l-1}-1+r-1} x_s x_{s+1} \right. \\ &\quad \left. + \sum_{s=r}^{2^{l-1}-1+r-2} x_s x_{s+2} + \dots + x_r x_{r+2^{l-1}-1} \right\} \\ &= \sum_{s=r}^{2^{l-1}-1+r} x_s^2 + 2 \sum_{u=1}^{2^{l-1}-1} \sum_{s=r}^{2^{l-1}-1+r-u} x_s x_{s+u} \\ &= \widehat{R}_{l,r}^2(0) + 2 \sum_{u=1}^{2^{l-1}-1} \widehat{R}_{l,r}^2(u). \end{aligned} \quad (6.15)$$

Similarly, it may be shown that

$$\left(\sum_{t=2^{l-1}+r}^{2^l-1+r} x_t\right)^2 = \widehat{R}_{l,r}^3(0) + 2 \sum_{u=1}^{2^{l-1}-1} \widehat{R}_{l,r}^3(u). \quad (6.16)$$

On substituting equations (6.14), (6.15) and (6.16) into (6.13) we obtain the required result. \square

6.3 The CLACV and stationary processes

In this section we consider the bias of the Haar CLACV estimator when estimating the covariance structure of second order stationary time series. To this end, let $\{x_t\}_{t=0,\dots,2^j-1}$ be a realisation of a zero-mean, second order stationary process. Then it can easily be shown that the expected values of the identities used in Proposition 6.2 are given by

$$\left. \begin{aligned} \mathbb{E}(\widehat{R}_{l,r}^0(u)) &= (2^l \Leftrightarrow u)R(u), \\ \mathbb{E}(\widehat{R}_{l,r}^1(u)) &= uR(u), \\ \mathbb{E}(\widehat{R}_{l,r}^2(u)) &= (2^{l-1} \Leftrightarrow u)R(u), \\ \text{and } \mathbb{E}(\widehat{R}_{l,r}^3(u)) &= (2^{l-1} \Leftrightarrow u)R(u) \end{aligned} \right\} \quad (6.17)$$

where $R(u)$ is simply the acvf of the underlying stationary process.

Using Proposition 6.2 and equation (6.17), we can derive the expected value of the Haar CLACV estimator for second order stationary processes:

Proposition 6.3

Let $\{x_t\}_{t=0,\dots,2^J-1}$ be a realisation of a second order stationary process. Then the expected value of the CLACV estimator, based on Haar wavelets, is given by

$$\begin{aligned} \mathbb{E}(\widehat{C}_J(z, \tau)) &= R(0) \left\{ 3/2 \sum_{j=1}^J \sum_{l=2}^J \Psi_j(\tau) A_{j,l}^{-1} + \sum_{j=1}^J \Psi_j(\tau) A_{j,1}^{-1} \right\} \Leftrightarrow R(1) \sum_{j=1}^J \Psi_j(\tau) A_{j,1}^{-1} \\ &+ \sum_{j=1}^J \sum_{l=1}^J \Psi_j(\tau) 2^{-l} A_{j,l}^{-1} \left\{ 2 \sum_{u=1}^{2^{l-1}-1} (2^l \Leftrightarrow 3u) R(u) \right. \\ &\left. \Leftrightarrow 2 \sum_{u=2^{l-1}}^{2^l-1} (2^l \Leftrightarrow u) R(u) \right\}. \end{aligned} \quad (6.18)$$

It is evident from the above that the Haar-based CLACV estimator, $\widehat{C}_J(z, \tau)$, is biased by contributions from lags other than τ . These contributions are difficult to appreciate when looking at equation (6.18). Thus a MAPLE routine has been written to evaluate $\mathbb{E}(\widehat{C}_J(z, \tau))$ for this specific case involving Haar wavelets. Given a curtailing level, $J \in \mathbb{N}$, and a lag, $\tau \in \mathbb{Z}$, the code uses analytic expressions for $\Psi_j(\tau)$ and A_J , derived by Nason *et al.* (2000) to return exact coefficients for the constituent elements of (6.18). The relevant files for this program can be downloaded from <http://www.stats.bris.ac.uk/~maiae/LACV>.

The following example demonstrates the nature of the bias within the Haar CLACV estimator for a simple moving average (MA) process.

Example 6.1 Assume that $\{x_t\}_{t=0,\dots,127}$ is a realisation of a zero-mean MA(3) process whose true covariance structure is given by $R(\tau) = \gamma_0 \delta_{0,\tau} + \gamma_1 \delta_{1,\tau} + \gamma_2 \delta_{2,\tau} + \gamma_3 \delta_{3,\tau}$, where $\delta_{n,\tau}$ is the Kronecker delta. Then it can be shown that the expected values of the Haar-based estimator, $\widehat{C}_J(z, \tau)$, are such that (to two decimal places)

$$\mathbb{E}(\widehat{C}_7(z, \tau)) = \begin{cases} 1.00\gamma_0 \Leftrightarrow 0.01\gamma_1 \Leftrightarrow 0.01\gamma_2 \Leftrightarrow 0.01\gamma_3 & \text{for } \tau = 0, \\ 0.00\gamma_0 + 0.99\gamma_1 \Leftrightarrow 0.01\gamma_2 \Leftrightarrow 0.01\gamma_3 & \text{for } \tau = 1, \\ 0.00\gamma_0 \Leftrightarrow 0.01\gamma_1 + 0.81\gamma_2 + 0.35\gamma_3 & \text{for } \tau = 2, \\ 0.00\gamma_0 \Leftrightarrow 0.01\gamma_1 + 0.35\gamma_2 + 0.27\gamma_3 & \text{for } \tau = 3, \\ 0.00\gamma_0 \Leftrightarrow 0.01\gamma_1 \Leftrightarrow 0.11\gamma_2 + 0.19\gamma_3 & \text{for } \tau = 4. \end{cases} \quad (6.19)$$

Note how $\mathbb{E}(\widehat{C}_7(z, \tau))$ is able to provide a reliable estimate of the autocovariance structure at lags 0 and 1. For $\tau = 2$ or 3, the CLACV estimator is contaminated by contributions from other lags. Further, note how the contribution of γ_τ in $\mathbb{E}(\widehat{C}_7(z, \tau))$ decreases quickly as τ increases — this suggests that a component of the bias may be behaving like $|\tau|/N$ as in the classical case (an avenue for future work). If the length of the series increases to $4096 = 2^{12}$ observations then we obtain,

$$\mathbb{E}(\widehat{C}_{12}(z, \tau)) = \begin{cases} 1.00\gamma_0 \Leftrightarrow 0.00\gamma_1 \Leftrightarrow 0.00\gamma_2 \Leftrightarrow 0.00\gamma_3 & \text{for } \tau = 0, \\ 0.00\gamma_0 + 1.00\gamma_1 \Leftrightarrow 0.00\gamma_2 \Leftrightarrow 0.00\gamma_3 & \text{for } \tau = 1, \\ 0.00\gamma_0 \Leftrightarrow 0.00\gamma_1 + 0.82\gamma_2 + 0.36\gamma_3 & \text{for } \tau = 2, \\ 0.00\gamma_0 \Leftrightarrow 0.00\gamma_1 + 0.36\gamma_2 + 0.28\gamma_3 & \text{for } \tau = 3, \\ 0.00\gamma_0 \Leftrightarrow 0.00\gamma_1 \Leftrightarrow 0.10\gamma_2 + 0.20\gamma_3 & \text{for } \tau = 4. \end{cases} \quad (6.20)$$

Again, at lags 2, 3 and 4 the estimator is contaminated.

To demonstrate the effect of this contamination, consider the following situation: suppose that $\{x_t\}_{t=0, \dots, 127}$ is a realisation of

$$\begin{aligned} X_t &= 10 \times \frac{1}{\sqrt{2}}(\epsilon_t \Leftrightarrow \epsilon_{t-1}) + 40 \times \frac{1}{2}(\epsilon_t + \epsilon_{t-1} \Leftrightarrow \epsilon_{t-2} \Leftrightarrow \epsilon_{t-3}) \\ &= (20 + 10/\sqrt{2})\epsilon_t + (20 \Leftrightarrow 10/\sqrt{2})\epsilon_{t-1} + (\Leftrightarrow 20\epsilon_{t-2}) + (\Leftrightarrow 20\epsilon_{t-3}), \end{aligned} \quad (6.21)$$

where the $\{\epsilon_t\}$ are iid $N(0, 1)$ random variables. Then it is easily shown that the acvf of this process is given by

$$R(\tau) = \begin{cases} 1700 & \text{for } \tau = 0, \\ 491.4 & \text{for } \tau = 1, \\ \Leftrightarrow 800 & \text{for } \tau = 2, \\ \Leftrightarrow 541.4 & \text{for } \tau = 3, \\ 0 & \text{otherwise} \end{cases}$$

(correct to one decimal place). However, using equation (6.19), it can be seen that the expected values of the Haar-based CLACV estimator are given by

$$\mathbb{E}(\widehat{C}_7(z, \tau)) = \begin{cases} 1700 & \text{for } \tau = 0, \\ 491.4 & \text{for } \tau = 1, \\ \Leftrightarrow 850.7 & \text{for } \tau = 2, \\ \Leftrightarrow 438.7 & \text{for } \tau = 3, \\ \Leftrightarrow 28.6 & \text{for } \tau = 4. \end{cases} \quad (6.22)$$

Comparing equations (6.21) and (6.22) the effect of the inherent bias of $\widehat{C}_J(z, \tau)$ is clear.

An independent check to verify these results can be achieved by simulating realisations of such processes and computing the CLACV estimator directly using `WaveThresh`. To this end, 1000 simulations of the process given by equation (6.21) were generated, each realisation consisting of 128 data points.

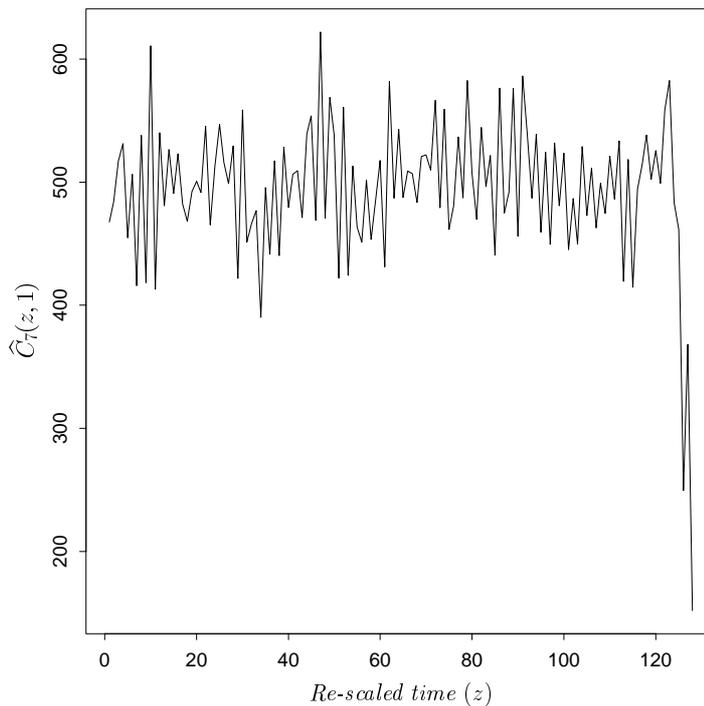


Figure 6.1: Mean of the CLACV estimate, $\widehat{C}_7(z, 1)$, for 1000 simulations of the MA(3) process.

The Haar CLACV estimate was evaluated for each simulation. Figure 6.1 displays the mean of the 1000 estimates of $\widehat{C}_7(z, 1)$. Note how the estimate fluctuates around 491, as we would expect from the results of equation (6.22). However, the last few time points differ wildly from the rest of the series. This artefact arises because the process simulations do not place any requirements on the boundary of the realisations, for example that the end of the series be similar to the start. However the LSW modelling approach of Nason et al. (2000) requires that such a condition be made. To overcome this artefact, we will only focus on the first 120 time points of $\widehat{C}_7(z, \tau)$.

Since the process used in the simulation of $\{X_t\}_{t=0,\dots,127}$ is second-order stationary, $C(z, \tau)$, is independent of z . Hence in this case, an estimate of the CLACV may be obtained by averaging $\widehat{C}_\tau(z, \tau)$ over time (see Table 6.1). Note how these estimates concur with the theoretical values in equation (6.22), displaying noticeable bias at lags 2, 3 and 4.

	<i>Estimate</i>
$\widehat{C}_J(z, 0)$	1709
$\widehat{C}_J(z, 1)$	499
$\widehat{C}_J(z, 2)$	-843
$\widehat{C}_J(z, 3)$	-439
$\widehat{C}_J(z, 4)$	-34

Table 6.1: Time-averaged estimates of the mean of 1000 estimates of the CLACV for the MA(3) process defined in equation (6.21).

6.4 Summary

This chapter has considered the use of the curtailed LACV estimator as a measure of the covariance structure within time series. We have demonstrated that the Haar CLACV estimator may be expressed in terms of “windowed” classical autocovariances. This allows us to interpret the estimator as a weighted sum of acvf estimators which estimate the stationary covariance structure on various partitions of the series. Additionally we have shown that, in the case of second order stationary processes, the Haar CLACV estimator displays bias at comparatively small lags. Clearly this is a problem which requires further study.

For stationary time series, the vector of CLACV estimators at a given time point, $\widehat{\mathbf{C}}_J = \{\widehat{C}_J(z, \tau)\}_{\tau=0,\dots,T-1}$ may be expressed as a linear combination of the vector of acvfs $\mathbf{R} = \{R(\tau)\}$: $\widehat{\mathbf{C}}_J = M\mathbf{R}$, the coefficients of M being obtained from equation (6.3). Hence one approach to correcting the bias of the Haar CLACV estimator might be to use a simple linear transformation of the estimates. However, initial investigations indicate that M is not invertible. Thus an alternative approach will be required.

Future avenues of research with this topic may include

1. calculating the mean squared error of the Haar CLACV estimator;
2. evaluating the statistical properties of the CLACV estimator for other wavelet families;
3. investigation of techniques to improve the properties of the estimator;
4. and the application of the LACV measure to other forms of processes, for example locally stationary time series.

Chapter 7

Conclusions and future directions

Wavelet methods have been applied to many branches of statistics, from density estimation to time series analysis. In a departure from these comparatively established areas of research, this thesis has considered the application of wavelets to the modelling of locally stationary random fields which lie on a regular grid. We introduced the LS2W model, which permits a local decomposition of the covariance structure into various scale contributions within certain directions. A wavelet analogue of the Fourier-based spectrum, termed the *local wavelet spectrum*, was introduced to quantify this local structure together with an associated estimation theory.

We then considered the application of the LS2W modelling approach to various texture analysis problems, its potential being contrasted against several recently proposed wavelet-based methods both on a conceptual and applied basis. For many textures, such as the Brodatz and Pill images, the LS2W approach was found to achieve classification rates which were comparable with those of the (uncorrected) NDWT — a consequence of the stationary nature of these textures. However, the true potential of our model becomes clear when we consider its application to non-stationary texture classification. In this case, the results obtained with a NDWT approach were found to be inferior to those of the LS2W model. This disparity is due to the latter's ability to correct for the power leakage which is induced by the redundancy of the NDWT.

By exploiting a well-known recursion which permits an economic construction of discrete autocorrelation wavelets, we were able to derive an efficient scheme for the construction of the inner product matrix of discrete autocorrelation wavelets. This recursive approach permits an $O(\log(N)^3)$ construction which compares favourably

with the brute force $O(N \log N)$ computation which has been used in the past for the estimation of the evolutionary wavelet spectrum.

Finally, we considered the local autocovariance measure proposed by Nason *et al.* (2000). Having introduced the CLACV estimator as a statistic for the estimation of the local covariance structure within time series, we demonstrated that the Haar CLACV estimator may be expressed in terms of classical, windowed, stationary time series measures. This allows us to interpret the estimator as a weighted sum of acvf estimators which estimate the stationary structure on various windows of the series. In addition, we showed that in the case of second order stationary processes, the Haar CLACV estimator displays bias, even at comparatively small lags.

We conclude this thesis by considering various avenues of future research. Naturally, some of these lead on directly from the work contained in this thesis. For example, an important conjecture made during the course of Chapter 3 states that the inner product operator of discrete autocorrelation wavelets, A , has a bounded inverse. This underpins two important results, Theorems 3.2 and 3.3, thus its resolution has considerable motivation. It is thought likely that the proof of Theorem 2, Nason *et al.* (2000), together with the separability of A will prove useful in establishing this result.

The locally stationary two dimensional process model which we have proposed focuses on analysing the covariance structure on regular grids of size $2^m \times 2^n$. Clearly, it is desirable from a practical perspective to extend such an approach to more general structures, including those with missing observations and/or unevenly spaced locations. The NDWT does not readily lend itself to such extensions thus alternative approaches, such as the lifting scheme (see Section 2.5.1), may need to be considered.

Local autocovariance estimation has received little attention to date, be it in the context of modelling time series or regular lattice processes. The work of Chapter 6 highlights several potential directions for future research with this measure, including issues of bias removal in the case of the Haar CLACV estimator and the investigation of the statistical properties of this estimator when founded upon other wavelet families. The extension of this work to two-dimensions may well prove to be a useful addition to the literature, since a map of local variation within images could have applications in medical imaging and geophysics.

Finally we turn to texture analysis. Although statistics has devised numerous

discrimination and classification schemes which are applied in this field, the issue of obtaining suitable measures from textured images has not yet received much attention in the statistics literature. Our exploratory analyses involving the standard Brodatz dataset and various industrial problems indicates that none of the approaches considered to date consistently excels. Thus the problem of measure choice is one which is ripe for future research. One interesting avenue lies in extending recent time series work involving wavelet packets. The approach which Hunt & Nason (2002) and Nason & Sapatinas (2001) have proposed consists of using wavelet packets associated with an explanatory time series $\{X_t\}$ to model a response series $\{Y_t\}$. By extending such an approach into two dimensions one could attempt to discover those packets which enable *discrimination* between two or more texture classes.

Appendix A

Chapter 3 Proofs

Proof of Proposition 3.1

Consider the proof for the horizontal case. Setting $\boldsymbol{\tau} = (\tau_1, \tau_2) \in \mathbb{Z}^2$, it follows from Definitions 3.1 and 3.6 that

$$\begin{aligned}\Psi_j^h(\boldsymbol{\tau}) &= \sum_u \sum_v \psi_{j,(u,v)}^h \psi_{j,(u-\tau_1,v-\tau_2)}^h \\ &= \sum_u \phi_{j,u} \phi_{j,u-\tau_1} \sum_v \psi_{j,v} \psi_{j,v-\tau_2} \\ &= \Phi_j(\tau_1) \Psi_j(\tau_2)\end{aligned}$$

as required. The proofs for $\Psi_j^v(\boldsymbol{\tau})$ and $\Psi_j^d(\boldsymbol{\tau})$ follow similarly. \square

Proof of Proposition 3.2

It is easily shown that $\Psi_j(\boldsymbol{\tau})$ and $\Phi_j(\boldsymbol{\tau})$ are symmetric in $\boldsymbol{\tau}$. Thus the first part follows from Proposition 3.1. To demonstrate that these two-dimensional autocorrelation wavelets are positive semi-definite functions, let $k_1, k_2, \dots, k_n \in \mathbb{R}$. Then for any set of locations $\boldsymbol{\tau}_1, \dots, \boldsymbol{\tau}_n$,

$$\begin{aligned}\sum_{p=1}^n \sum_{q=1}^n \Psi_j^l(\boldsymbol{\tau}_p \Leftrightarrow \boldsymbol{\tau}_q) k_p k_q &= \sum_{p=1}^n \sum_{q=1}^n \left\{ \sum_{\mathbf{u}} \psi_{j,\mathbf{u}}^l \psi_{j,\mathbf{u}-(\boldsymbol{\tau}_p-\boldsymbol{\tau}_q)}^l \right\} k_p k_q \\ &= \sum_{p=1}^n \sum_{q=1}^n \sum_{\mathbf{u}} \psi_{j,\mathbf{u}}^l \psi_{j,\mathbf{u}-(\boldsymbol{\tau}_p-\boldsymbol{\tau}_q)}^l k_p k_q.\end{aligned}$$

Setting $\mathbf{m} = \mathbf{u} \Leftrightarrow \tau_p$, we obtain

$$\begin{aligned} \sum_{p=1}^n \sum_{q=1}^n \Psi_j^l(\tau_p \Leftrightarrow \tau_q) k_p k_q &= \sum_{\mathbf{m}} \left\{ \sum_{p=1}^n \psi_{j, \mathbf{m} + \tau_p}^l k_p \right\} \left\{ \sum_{q=1}^n \psi_{j, \mathbf{m} + \tau_q}^l k_q \right\} \\ &= \sum_{\mathbf{m}} \left\{ \sum_{i=1}^n \psi_{j, \mathbf{m} + \tau_i}^l k_i \right\}^2 \\ &\geq 0. \end{aligned}$$

□

Proof of Lemma 3.1

$$\begin{aligned} \widehat{\psi}_j(\omega) &= \sum_k \psi_{j,k} e^{-i\omega k} \\ &= \sum_k \left\{ \sum_l h_{k-2l} \psi_{j-1,l} \right\} e^{-i\omega k} \\ &= \sum_l \psi_{j-1,l} \sum_k h_{k-2l} \psi_{j-1,l} e^{-i\omega k}. \end{aligned}$$

Upon making the substitution $p = k \Leftrightarrow 2l$, we obtain

$$\begin{aligned} \widehat{\psi}_j(\omega) &= \sum_l \sum_p h_p \psi_{j-1,l} e^{-i\omega(p+2l)} \\ &= \sum_l \psi_{j-1,l} e^{-i\omega 2l} \sum_p h_p e^{-i\omega p}. \end{aligned}$$

However from equation (2.12), $\sum_p h_p e^{-i\omega p} = \sqrt{2}m_0(\omega)$. Hence,

$$\begin{aligned} \widehat{\psi}_j(\omega) &= \sqrt{2}m_0(\omega) \widehat{\psi}_{j-1}(2\omega) \\ &= \sqrt{2}m_0(\omega) \sqrt{2}m_0(2\omega) \widehat{\psi}_{j-2}(4\omega). \end{aligned}$$

Repeating the above argument for scales $(j \Leftrightarrow 2), (j \Leftrightarrow 3), \dots, 1$ and using $\psi_{1,n} = g_n$, together with equation (2.20), we obtain

$$\widehat{\psi}_j(\omega) = 2^{j/2} m_1(2^{j-1}\omega) \prod_{k=0}^{j-2} m_0(2^k \omega).$$

The proof for $\widehat{\phi}_j(\omega)$ follows similarly.

□

Proof of Corollary 3.1

Consider the case in the vertical direction. Setting $\boldsymbol{\omega} = (\omega_1, \omega_2)$,

$$\begin{aligned}\widehat{\psi}_j^v(\boldsymbol{\omega}) &= \sum_x \sum_y \psi_{j,x} \phi_{j,y} e^{-i\omega_1 x} e^{-i\omega_2 y} \\ &= \sum_x \psi_{j,x} e^{-i\omega_1 x} \sum_y \phi_{j,y} e^{-i\omega_2 y} \\ &= \widehat{\psi}_j(\omega_1) \widehat{\phi}_j(\omega_2).\end{aligned}$$

Hence, using the results of Proposition 3.1,

$$\widehat{\psi}_j^v(\boldsymbol{\omega}) = 2^j m_1(2^{j-1}\omega_1) m_0(2^{j-2}\omega_2) \prod_{p=0}^{j-2} m_0(2^p \omega_1) m_0(2^p \omega_2).$$

The results for $\widehat{\psi}_j^h(\boldsymbol{\omega})$ and $\widehat{\psi}_j^d(\boldsymbol{\omega})$ follow similarly. \square

Proof of Theorem 3.1

The structure of this proof is similar to that of the one dimensional case, considered by Nason *et al.* (Theorem 1, 2000), although added care is required when dealing with the zeros of $m_0(\omega)$ and $m_1(\omega)$. This is due to the addition of directionality, $l \in \{h, v, d\}$, as well as scale, j , within the decomposition.

Suppose, by way of contradiction, that there exist two spectral representations of the *same* LS2W process. In other words, assume that there exist $w_{\eta, \mathbf{u}}^{(1)}$ and $w_{\eta, \mathbf{u}}^{(2)}$ such that

$$\left| w_{\eta, \mathbf{u}}^{(i)} \Leftrightarrow W_{\eta}^{(i)} \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right| = O \left(\frac{1}{\max\{R, S\}} \right) \quad \text{for } i = 1, 2$$

which also possess the same covariance structure. In other words

$$C(\mathbf{z}, \boldsymbol{\tau}) = \sum_{\eta} S_{\eta}^{(1)}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau}) = \sum_{\eta} S_{\eta}^{(2)}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau})$$

where C is defined in (3.43), for all $\mathbf{z} \in (0, 1)^2$, $\boldsymbol{\tau} = \mathbb{Z}^2$, where $S_{\eta}^i(\mathbf{z}) = \left| W_{\eta}^{(i)}(\mathbf{z}) \right|^2$ for $i = 1, 2$.

Let $\Delta_{\eta}(\mathbf{z}) \equiv S_{\eta}^{(1)}(\mathbf{z}) \Leftrightarrow S_{\eta}^{(2)}(\mathbf{z})$. To prove this result, we must show that

$$\begin{aligned}0 &= \sum_{\eta} \Delta_{\eta}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau}) \quad \forall \mathbf{z} \in (0, 1)^2, \forall \boldsymbol{\tau} \in \mathbb{Z}^2, \\ \Rightarrow \Delta_{\eta}(\mathbf{z}) &= 0 \quad \forall \eta, \forall \mathbf{z} \in (0, 1)^2,\end{aligned}$$

What we actually show is that

$$0 = \sum_{\eta} \tilde{\Delta}_{\eta}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau}) \quad \forall \mathbf{z} \in (0, 1)^2, \forall \boldsymbol{\tau} \in \mathbb{Z}^2,$$

implies that $\tilde{\Delta}_{\eta}(\mathbf{z}) = 0, \forall \eta \geq 1, \forall \mathbf{z} \in (0, 1)^2$, where

$$\tilde{\Delta}_{\eta}(\mathbf{z}) = 2^{-2j(\eta)} \Delta_{\eta}(\mathbf{z}) \quad (\text{A.1})$$

and $j(\eta) \equiv \eta \Leftrightarrow \lfloor \frac{\eta-1}{J} \rfloor J$ for $\eta = 1, \dots, 3J$. Here $\lfloor \cdot \rfloor$ denotes the floor function. Thus $j(\eta)$ simply refers to scale.

To start, recall that the operator $A = (A_{\eta, \nu})_{\eta, \nu \geq 1}$ is defined by

$$A_{\eta, \nu} = \sum_{\boldsymbol{\tau}} \Psi_{\eta}(\boldsymbol{\tau}) \Psi_{\nu}(\boldsymbol{\tau}).$$

However, by Parseval's relation

$$\begin{aligned} A_{\eta, \nu} &= \sum_{\boldsymbol{\tau}} \Psi_{\eta}(\boldsymbol{\tau}) \Psi_{\nu}(\boldsymbol{\tau}) \\ &= \left(\frac{1}{2\pi} \right)^2 \int \int \hat{\Psi}_{\eta}(\boldsymbol{\omega}) \hat{\Psi}_{\nu}(\boldsymbol{\omega}) d\boldsymbol{\omega}, \end{aligned} \quad (\text{A.2})$$

where $\hat{\Psi}_{\eta}(\boldsymbol{\omega})$ takes one of the following forms:

$$\left. \begin{aligned} |\hat{\Psi}_j^v(\boldsymbol{\omega})|^2 &= 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_0(2^{j-1}\omega_2)|^2 \prod_{p=0}^{j-2} |m_0(2^p w_1) m_0(2^p w_2)|^2 \\ |\hat{\Psi}_j^h(\boldsymbol{\omega})|^2 &= 2^{2j} |m_0(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{p=0}^{j-2} |m_0(2^p w_1) m_0(2^p w_2)|^2 \\ |\hat{\Psi}_j^d(\boldsymbol{\omega})|^2 &= 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{p=0}^{j-2} |m_0(2^p w_1) m_0(2^p w_2)|^2 \end{aligned} \right\} \quad (\text{A.3})$$

The above follows as a consequence of Proposition 3.1 and the result that $\hat{\Psi}_j^l(\boldsymbol{\omega}) = \left| \hat{\psi}_j^l(\boldsymbol{\omega}) \right|^2$. Thus,

$$\begin{aligned} 0 &= \sum_{\eta} \tilde{\Delta}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau}) \\ \Rightarrow 0 &= \sum_{\eta} \tilde{\Delta}_{\eta}(\mathbf{z}) \Psi_{\eta}(\boldsymbol{\tau}) \sum_{\nu} \tilde{\Delta}_{\nu} \Psi_{\nu}(\boldsymbol{\tau}), \quad \forall \mathbf{z} \in (0, 1)^2, \forall \boldsymbol{\tau} \in \mathbb{Z}^2. \end{aligned}$$

Hence,

$$0 = \sum_{\eta} \sum_{\nu} \tilde{\Delta}_{\eta}(\mathbf{z}) \tilde{\Delta}_{\nu}(\mathbf{z}) \sum_{\boldsymbol{\tau}} \Psi_{\eta}(\boldsymbol{\tau}) \Psi_{\nu}(\boldsymbol{\tau}).$$

Applying Parseval's relation, (A.2), we obtain

$$\begin{aligned}
0 &= \sum_{\eta} \sum_{\nu} \tilde{\Delta}_{\eta}(\mathbf{z}) \tilde{\Delta}_{\nu}(\mathbf{z}) \left(\frac{1}{2\pi}\right)^2 \int \int \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) \widehat{\Psi}_{\nu}(\boldsymbol{\omega}) d\boldsymbol{\omega} \\
&= \int \int d\boldsymbol{\omega} \sum_{\eta} \sum_{\nu} \tilde{\Delta}_{\eta}(\mathbf{z}) \tilde{\Delta}_{\nu}(\mathbf{z}) \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) \widehat{\Psi}_{\nu}(\boldsymbol{\omega}) \\
&= \int \int d\boldsymbol{\omega} \left(\sum_{\eta} \tilde{\Delta}_{\eta}(\mathbf{z}) \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) \right)^2. \tag{A.4}
\end{aligned}$$

By Definition 3.5, $S_{\eta}(\mathbf{z})$ is positive, hence $|S_{\eta}(z)| = S_{\eta}(z)$. Furthermore, by Property 3.1, $\sum_{\eta} S_{\eta}(\mathbf{z}) < \infty$, uniformly in \mathbf{z} . Thus, $\sum_{\eta} |\Delta_{\eta}(\mathbf{z})| < \infty$ and hence $\sum_{\eta} 2^{2j(\eta)} |\tilde{\Delta}_{\eta}(\mathbf{z})| < \infty$. Further, we can infer that $\sum_{\eta} \tilde{\Delta}_{\eta}(\mathbf{z}) \widehat{\Psi}_{\eta}(\boldsymbol{\omega})$ is a continuous function for $\boldsymbol{\omega} \in [\Leftrightarrow\pi, \pi]^2$. This is because $2^{-2j(\eta)} \widehat{\Psi}_{\eta}(\boldsymbol{\omega})$ is continuous in this domain (it is simply a trigonometric polynomial in two variables, uniformly bounded above by 1). Hence, (A.4) if and only if

$$0 = \sum_{\eta} \tilde{\Delta}_{\eta}(\mathbf{z}) \widehat{\Psi}_{\eta}(\boldsymbol{\omega}), \quad \forall \boldsymbol{\omega} \in [\Leftrightarrow\pi, \pi]^2, \forall \mathbf{z} \in (0, 1)^2.$$

All that remains now is to demonstrate the pointwise implication of $\tilde{\Delta}_{\eta}(\mathbf{z}) = 0$ $\forall \eta \geq 1, \forall \mathbf{z} \in (0, 1)^2$. To achieve this, we use continuity arguments and the insertion of the zeros of $|m_0(2^l \omega)|^2$ and $|m_1(2^l \omega)|^2$.

We start by fixing $\mathbf{z} \in (0, 1)^2$ and set $\tilde{\Delta}_{\eta} = \tilde{\Delta}_{\eta}(\mathbf{z})$ at this fixed point \mathbf{z} . Then, ,

$$\begin{aligned}
0 &= \sum_{\eta} \tilde{\Delta}_{\eta} \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) \\
&= \sum_{\eta=1}^J \tilde{\Delta}_{\eta} \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) + \sum_{\eta=J+1}^{2J} \tilde{\Delta}_{\eta} \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) + \sum_{\eta=2J+1}^{3J} \tilde{\Delta}_{\eta} \widehat{\Psi}_{\eta}(\boldsymbol{\omega}) \\
&= \sum_{\eta=1}^J \tilde{\Delta}_{\eta} 2^{2j} |m_1(2^{j-1} \omega_1)|^2 |m_0(2^{j-1} \omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \\
&\quad + \sum_{\eta=J+1}^{2J} \tilde{\Delta}_{\eta} 2^{2j} |m_0(2^{j-1} \omega_1)|^2 |m_1(2^{j-1} \omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \tag{A.5} \\
&\quad + \sum_{\eta=2J+1}^{3J} \tilde{\Delta}_{\eta} 2^{2j} |m_1(2^{j-1} \omega_1)|^2 |m_1(2^{j-1} \omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2.
\end{aligned}$$

From Daubechies (Chapter 5, 1992) we know that m_0 is a 2π -periodic function which is such that $|m_0(\xi)|^2 + |m_0(\xi + \pi)|^2 = 1$ and,

$$|m_0(\pi)|^2 = 0. \tag{A.6}$$

Thus, $|m_0(0)|^2 = 1$. Recall also that $|m_1(\omega)|^2 = 1 \Leftrightarrow |m_0(\omega)|^2$ for Daubechies compactly supported wavelets.

To show that $\tilde{\Delta}_1, \tilde{\Delta}_{J+1}$ and $\tilde{\Delta}_{2J+1}$ are all zero, consider the following: Let $\omega_1 = \pi$ and ω_2 vary. Then by the construction of $\widehat{\Psi}_\eta(\omega_1, \omega_2)$ and using (A.6) it follows that $\widehat{\Psi}_\eta(\pi, \omega_2) = 0$ for $\eta = 2, 3, \dots, J, J+1, \dots, 2J, 2J+2, \dots, 3J$. However since $|m_1(\pi)|^2 = 1$ (A.5) simplifies to

$$\begin{aligned} 0 &= \tilde{\Delta}_1 4|m_1(\pi)|^2|m_0(\omega_2)|^2 + \tilde{\Delta}_{2J+1} 4|m_1(\pi)|^2|m_1(\omega_2)|^2 \\ &= \tilde{\Delta}_1|m_0(\omega_2)|^2 + \tilde{\Delta}_{2J+1}|m_1(\omega_2)|^2, \end{aligned}$$

$\forall \omega_2 \in [\Leftrightarrow\pi, \pi]$. Now suppose, without loss of generality, that $\omega_2 = 0$. Then

$$|m_1(0)|^2 = 1 \Leftrightarrow |m_0(0)|^2 = 0.$$

Hence,

$$\begin{aligned} 0 &= \tilde{\Delta}_1|m_0(0)|^2 + \tilde{\Delta}_{2J+1}|m_1(0)|^2 \\ &= \tilde{\Delta}_1|m_0(0)|^2. \end{aligned}$$

In other words,

$$\tilde{\Delta}_1 = 0. \tag{A.7}$$

To show that $\tilde{\Delta}_{2J+1}$ is zero, reconsider (A.7):

$$\begin{aligned} 0 &= \tilde{\Delta}_1|m_1(\pi)|^2|m_0(\omega_2)|^2 + \tilde{\Delta}_{2J+1}|m_1(\pi)|^2|m_1(\omega_2)|^2 \\ &= \tilde{\Delta}_{2J+1}|m_1(\pi)|^2|m_1(\omega_2)|^2, \quad \text{as } \tilde{\Delta}_1 \text{ is zero} \\ &= \tilde{\Delta}_{2J+1}|m_1(\omega_2)|^2 \quad \forall \omega_2 \in [\Leftrightarrow\pi, \pi]. \end{aligned}$$

Setting $\omega_2 = \pi$, we obtain,

$$\begin{aligned} 0 &= \tilde{\Delta}_{2J+1}|m_1(\pi)|^2 \\ \implies \tilde{\Delta}_{2J+1} &= 0. \end{aligned} \tag{A.8}$$

To conclude this part of the proof, it remains to show that $\tilde{\Delta}_{J+1} = 0$. To this end, reconsider (A.5) setting $\omega_2 = \pi$ and letting ω_1 vary. Then, as $|m_0(\pi)|^2 = 0$, it follows

that $\widehat{\Psi}_\eta(\omega_1, \pi) = 0$ for all η except $\eta = J + 1$ and $2J + 1$. However, we have already shown that $\tilde{\Delta}_{2J+1} = 0$. Thus (A.5) simplifies to

$$0 = \tilde{\Delta}_{J+1} |m_0(\omega_1)|^2 \quad \forall \omega_1 \in [\Leftarrow\pi, \pi].$$

Setting $\omega_1 = 0$ ($\Rightarrow |m_0(w_1)|^2 = 1$), we find that

$$\tilde{\Delta}_{J+1} = 0. \quad (\text{A.9})$$

We have therefore shown that $\tilde{\Delta}_1, \tilde{\Delta}_{J+1}$ and $\tilde{\Delta}_{2J+1} = 0$. Thus (A.5) simplifies to

$$\begin{aligned} 0 = & \sum_{\eta=2}^J \tilde{\Delta}_\eta 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_0(2^{j-1}\omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \\ & + \sum_{\eta=J+2}^{2J} \tilde{\Delta}_\eta 2^{2j} |m_0(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \\ & + \sum_{\eta=2J+2}^{3J} \tilde{\Delta}_\eta 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{l=0}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \end{aligned}$$

In other words,

$$\begin{aligned} 0 = & |m_0(w_1) m_0(w_2)|^2 \left\{ \sum_{\eta=2}^J \tilde{\Delta}_\eta 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_0(2^{j-1}\omega_2)|^2 \prod_{l=1}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \right. \\ & + \sum_{\eta=J+2}^{2J} \tilde{\Delta}_\eta 2^{2j} |m_0(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{l=1}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \quad (\text{A.10}) \\ & \left. + \sum_{\eta=2J+2}^{3J} \tilde{\Delta}_\eta 2^{2j} |m_1(2^{j-1}\omega_1)|^2 |m_1(2^{j-1}\omega_2)|^2 \prod_{l=1}^{j-2} |m_0(2^l w_1) m_0(2^l w_2)|^2 \right\}. \end{aligned}$$

As $|m_0(\omega)|^2$ and $|m_1(\omega)|^2$ are analytic and $m_0(\omega), m_1(\omega)$, as trigonometric polynomials, have finitely many zeros, it follows that the (continuous) function in the braces must vanish identically. Setting $\omega_1 = \pi/2$ and letting ω_2 vary, we find that

$$|m_0(2w_1)|^2 = |m_0(\pi)|^2 = 0 \quad \text{and} \quad |m_1(2w_1)|^2 = 1.$$

Hence (A.10) reduces to

$$\begin{aligned} 0 &= \tilde{\Delta}_2 2^4 |m_1(\pi)|^2 |m_0(2\omega_2)|^2 + \tilde{\Delta}_{2J+2} 2^4 |m_1(\pi)|^2 |m_1(2\omega_2)|^2 \\ 0 &= \tilde{\Delta}_2 |m_0(2\omega_2)|^2 + \tilde{\Delta}_{2J+2} |m_1(2\omega_2)|^2 \quad \forall \omega_2 \in [\Leftarrow\pi, \pi]. \end{aligned} \quad (\text{A.11})$$

Without loss of generality, let $\omega_2 = 0$. Then as $|m_1(0)|^2 = 0$, the above simplifies to

$$\tilde{\Delta}_2 = 0.$$

Thus the expression in (A.11), where ω_2 can take *any* value, simplifies to

$$0 = \tilde{\Delta}_{2J+2} |m_1(2\omega_2)|^2.$$

Setting $\omega_2 = \pi/2$, we obtain

$$0 = \tilde{\Delta}_{2J+2} |m_1(\pi)|^2 = \tilde{\Delta}_{2J+2}.$$

Finally to show that $\tilde{\Delta}_{J+2} = 0$, reconsider (A.10), this time allowing ω_1 to vary and setting $\omega_2 = \pi/2$. The expression reduces to

$$\begin{aligned} 0 &= \tilde{\Delta}_{J+2} 2^4 |m_0(2\omega_1)|^2 |m_1(\pi)|^2 + \tilde{\Delta}_{2J+2} 2^4 |m_1(2\omega_1)|^2 |m_1(\pi)|^2 \quad \text{but } \tilde{\Delta}_{2J+2} = 0, \\ &= \tilde{\Delta}_{J+2} |m_1(2\omega_1)|^2 \quad \forall \omega_1 \in [\pi/2, \pi]. \end{aligned}$$

Setting $\omega_1 = 0$ it follows that

$$\tilde{\Delta}_{J+2} = 0.$$

Continuing with this scheme for $j(\eta) = 3, 4, 5, \dots$ leads to the result that

$$\tilde{\Delta}_\eta(\mathbf{z}) = 0 \quad \forall \eta, \forall \mathbf{z} \in (0, 1)^2.$$

Hence the LWS are uniquely defined given the corresponding LS2W process. Furthermore, since we have shown that $0 = \sum_\eta \tilde{\Delta}_\eta(\mathbf{z}) \Psi_\eta(\boldsymbol{\tau})$ if, and only if $\tilde{\Delta}_j(\mathbf{z}) = 0$, we have that $\{\Psi_\eta(\boldsymbol{\tau})\}_{\eta=1}^\infty$ are linearly independent. Moreover, since A is the Inner Product (or Gram) matrix of the Ψ_η , A is clearly symmetric and also positive definite. Consequently the eigenvalues of A are positive. \square

Proof of Theorem 3.2

Let $\mathbf{p} = [z\mathbf{R}]$. By definition,

$$\begin{aligned} \mathbb{E}(I_{j,\mathbf{p}}^l) &= \mathbb{E}[(d_{j,\mathbf{p}}^l)^2] \\ &= \mathbb{E}\left[\left(\sum_{\mathbf{r}} X_{\mathbf{r}} \psi_{j,\mathbf{p}}^l(\mathbf{r})\right)^2\right]. \end{aligned}$$

As $\{X_{\mathbf{r}}\}$ is assumed to be a LS2W process, we obtain

$$\begin{aligned}
\mathbb{E}(I_{j,\mathbf{p}}^l) &= \mathbb{E} \left[\left(\sum_{\mathbf{r}} \left\{ \sum_{l_1, j_1, \mathbf{u}_1} w_{j_1, \mathbf{u}_1}^{l_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}) \xi_{j_1, \mathbf{u}_1}^{l_1} \right\} \psi_{j, \mathbf{p}}^l(\mathbf{r}) \right)^2 \right] \\
&= \mathbb{E} \left[\left(\sum_{\mathbf{r}_1} \sum_{l_1, j_1, \mathbf{u}_1} w_{j_1, \mathbf{u}_1}^{l_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \xi_{j_1, \mathbf{u}_1}^{l_1} \psi_{j, \mathbf{p}}^l(\mathbf{r}_1) \right. \right. \\
&\quad \left. \left. \sum_{\mathbf{r}_2} \sum_{l_2, j_2, \mathbf{u}_2} w_{j_2, \mathbf{u}_2}^{l_2} \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{r}_2) \xi_{j_2, \mathbf{u}_2}^{l_2} \psi_{j, \mathbf{p}}^l(\mathbf{r}_2) \right) \right] \\
&= \sum_{\mathbf{r}_1, \mathbf{r}_2} \sum_{l_1, l_2} \sum_{j_1, j_2} \sum_{\mathbf{u}_1, \mathbf{u}_2} w_{j_1, \mathbf{u}_1}^{l_1} w_{j_2, \mathbf{u}_2}^{l_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{r}_2) \psi_{j, \mathbf{p}}^l(\mathbf{r}_1) \psi_{j, \mathbf{p}}^l(\mathbf{r}_2) \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}).
\end{aligned}$$

By the orthonormality of the increment sequence and Assumption 1 (page 51), it follows that

$$\begin{aligned}
\text{Cov}(\xi_{j_1, \mathbf{u}_1}^{l_1}, \xi_{j_2, \mathbf{u}_2}^{l_2}) &= \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \\
&= \delta_{j_1, j_2} \delta_{l_1, l_2} \delta_{\mathbf{u}_1, \mathbf{u}_2}.
\end{aligned}$$

Hence,

$$\mathbb{E}(I_{j,\mathbf{p}}^l) = \sum_{l_1, j_1, \mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j, \mathbf{p}}^l(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j, \mathbf{p}}^l(\mathbf{r}_2). \quad (\text{A.12})$$

Upon making the substitution $\mathbf{u} = \mathbf{x} + \mathbf{p}$ we obtain:

$$\begin{aligned}
\mathbb{E}(I_{j,\mathbf{p}}^l) &= \sum_{l_1, j_1, \mathbf{x}} (w_{j_1, \mathbf{x}+\mathbf{p}}^{l_1})^2 \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}+\mathbf{p}}^{l_1}(\mathbf{r}) \psi_{j, \mathbf{p}}^l(\mathbf{r}) \right\}^2 \\
&= \sum_{l_1, j_1, \mathbf{x}} (w_{j_1, \mathbf{x}+\mathbf{p}}^{l_1})^2 \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}+\mathbf{p}-\mathbf{r}}^{l_1} \psi_{j, \mathbf{p}-\mathbf{r}}^l \right\}^2. \quad (\text{A.13})
\end{aligned}$$

As the sum over \mathbf{x} ranges over $\{\mathbf{x} = (x_1, x_2) : x_1, x_2 \in \mathbb{Z}\}$, it follows that \mathbf{p} in the final summation of equation (A.13) becomes redundant. Hence,

$$\mathbb{E}(I_{j,\mathbf{p}}^l) = \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} (w_{j_1, \mathbf{x}+\mathbf{p}}^{l_1})^2 \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j, -\mathbf{r}}^l \right\}^2.$$

It is easily shown that

$$\left| |w_{j, [\mathbf{zR}] + \mathbf{x}}^l|^2 \Leftrightarrow S_j^l \left(\mathbf{z} + \frac{\mathbf{x}}{\mathbf{R}} \right) \right| \leq \frac{C_j^l}{\max\{R, S\}}.$$

See the proof of Proposition 3.3 for further details. Hence

$$|w_{j, [\mathbf{zR}] + \mathbf{x}}^l|^2 \Leftrightarrow S_j^l \left(\mathbf{z} + \frac{\mathbf{x}}{\mathbf{R}} \right) \leq \frac{C_j^l}{\max\{R, S\}}.$$

In other words,

$$|w_{j_1, \mathbf{x}+\mathbf{p}}^{l_1}|^2 = S_{j_1}^{l_1} \left(\frac{\mathbf{x} + \mathbf{p}}{\mathbf{R}} \right) + O \left(\frac{C_{j_1}^{l_1}}{\max\{R, S\}} \right).$$

Thus,

$$\begin{aligned} \mathbb{E}(I_{j, \mathbf{p}}^l) &= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} \left(S_{j_1}^{l_1} \left(\frac{\mathbf{x} + \mathbf{p}}{\mathbf{R}} \right) + O \left(\frac{C_{j_1}^{l_1}}{\max\{R, S\}} \right) \right) \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}^2 \\ &= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} S_{j_1}^{l_1} \left(\frac{\mathbf{x} + \mathbf{p}}{\mathbf{R}} \right) \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}^2 + O \left(\frac{1}{\max\{R, S\}} \right). \end{aligned}$$

Aside: The remainder term can be brought out because

1. the number of terms in the wavelet product $\left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}$ is finite and bounded as a function of \mathbf{x} due to j being fixed and the fact that discrete wavelets have compact support.
2. and as $\sum_{l_1} \sum_{j_1} C_{j_1}^{l_1} < \infty$.

Moreover, as we show in the proof of proof of Proposition 3.3, if we set $\mathbf{z} = (z_1, z_2)$ and $\tau = (\tau_1, \tau_2)$, then

$$|S_j^l(z_1 + \tau_1/R, z_2 + \tau_2/S) \Leftrightarrow S_j^l(z_1, z_2)| = O \left(L_j^l \left(\frac{|\tau_1|}{R} + \frac{|\tau_2|}{S} \right) \right).$$

Thus,

$$S_j^l(z_1 + \tau_1/R, z_2 + \tau_2/S) = S_j^l(z_1, z_2) + O \left(L_j^l \frac{|\tau_1| + |\tau_2|}{\min\{R, S\}} \right). \quad (\text{A.14})$$

Incorporating this Lipschitz property of the $\{S_j^l\}$, (A.14), we obtain

$$\begin{aligned} \mathbb{E}(I_{j, \mathbf{p}}^l) &= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} \left(S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) + O \left(\frac{L_{j_1}^l \|\mathbf{x}\|_1}{\min\{R, S\}} \right) \right) \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}^2 \\ &\quad + O \left(\frac{1}{\max\{R, S\}} \right) \\ &\quad \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} \left\{ S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}^2 \right\} + O \left(\frac{1}{\min\{R, S\}} \right), \end{aligned}$$

again due to $\left\{ \sum_{\mathbf{r}} \psi_{j_1, \mathbf{x}-\mathbf{r}}^{l_1} \psi_{j_1, -\mathbf{r}}^l \right\}$ being finite and the summability of the Lipschitz constants L_j^l .

Expanding the squared wavelet product term yields

$$\mathbb{E}(I_{j,\mathbf{p}}^l) = \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \left\{ \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{x}-\mathbf{r}_1}^{l_1} \psi_{j, -\mathbf{r}_1}^l \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{x}-\mathbf{r}_2}^{l_1} \psi_{j, -\mathbf{r}_2}^l \right\} + O \left(\frac{1}{\min\{R, S\}} \right).$$

Upon making the substitution $\mathbf{s} = \mathbf{r}_2 \Leftrightarrow \mathbf{r}_1$ we obtain

$$\begin{aligned} \mathbb{E}(I_{j,\mathbf{p}}^l) &= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{x}} S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \left\{ \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{x}-\mathbf{r}_1}^{l_1} \psi_{j, -\mathbf{r}_1}^l \sum_{\mathbf{s}} \psi_{j_1, \mathbf{x}-\mathbf{s}-\mathbf{r}_1}^{l_1} \psi_{j, -\mathbf{s}-\mathbf{r}_1}^l \right\} \\ &\quad + O \left(\frac{1}{\min\{R, S\}} \right) \\ &= \sum_{l_1} \sum_{j_1} S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \sum_{\mathbf{r}_1} \sum_{\mathbf{s}} \psi_{j, -\mathbf{r}_1}^l \psi_{j, -\mathbf{s}-\mathbf{r}_1}^l \sum_{\mathbf{x}} \psi_{j_1, \mathbf{x}-\mathbf{r}_1}^{l_1} \psi_{j_1, \mathbf{x}-\mathbf{r}_1-\mathbf{s}}^{l_1} \\ &\quad + O \left(\frac{1}{\min\{R, S\}} \right). \end{aligned}$$

By recognition, this last summation is simply the discrete a.c. wavelet, $\Psi_{j_1}^{l_1}(\mathbf{s})$. Thus,

$$\begin{aligned} \mathbb{E}(I_{j,\mathbf{p}}^l) &= \sum_{l_1} \sum_{j_1} S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \sum_{\mathbf{s}} \Psi_{j_1}^{l_1}(\mathbf{s}) \sum_{-\mathbf{r}_1} \psi_{j, \mathbf{r}_1}^l \psi_{j, -\mathbf{s}-\mathbf{r}_1}^l + O \left(\frac{1}{\min\{R, S\}} \right) \\ &= \sum_{l_1} \sum_{j_1} S_{j_1}^{l_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \sum_{\mathbf{s}} \Psi_{j_1}^{l_1}(\mathbf{s}) \Psi_j^l(\mathbf{s}) + O \left(\frac{1}{\min\{R, S\}} \right). \end{aligned} \quad (\text{A.15})$$

Setting $\eta = (j, l)$ and $\eta_1 = (j_1, l_1)$, and recalling that $\sum_{\mathbf{s}} \Psi_{\eta_1}(\mathbf{s}) \Psi_{\eta}(\mathbf{s}) = A_{\eta, \eta_1}$, equation (A.15) reduces to:

$$\mathbb{E}(I_{\eta, \mathbf{p}}) = \sum_{\eta_1} A_{\eta, \eta_1} S_{\eta_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) + O \left(\frac{1}{\min\{R, S\}} \right),$$

as required. □

Proof of Theorem 3.3

Variance: The variance of a wavelet periodogram,

$$\begin{aligned} \text{Var}(I_{j,\mathbf{p}}^l) &= \text{Var}((d_{j,\mathbf{p}}^l)^2) \\ &= \mathbb{E}((d_{j,\mathbf{p}}^l)^4) \Leftrightarrow \mathbb{E}((d_{j,\mathbf{p}}^l)^2)^2. \end{aligned}$$

We already know the asymptotic form of $\mathbb{E}((d_{j,\mathbf{p}}^l)^2)$. We therefore focus on

$$\begin{aligned}
\mathbb{E}((d_{j,\mathbf{p}}^l)^4) &= \mathbb{E}\left(\left(\sum_{\mathbf{r}} X_{\mathbf{r}} \psi_{j,\mathbf{p}}^l(\mathbf{r})\right)^4\right) \\
&= \mathbb{E}\left(\left(\sum_{\mathbf{r}} \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} w_{j_1, \mathbf{u}_1}^{l_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}) \xi_{j_1, \mathbf{u}_1}^{l_1} \psi_{j,\mathbf{p}}^l(\mathbf{r})\right)^4\right) \\
&= \mathbb{E}\left(\prod_{i=1}^4 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \xi_{j_i, \mathbf{u}_i}^{l_i} \psi_{j,\mathbf{p}}^l(\mathbf{r}_i)\right) \\
&= \prod_{i=1}^4 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j,\mathbf{p}}^l(\mathbf{r}_i).
\end{aligned}$$

Consider the term $\mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4})$. Using a result due to Isserlis (1918),

$$\begin{aligned}
\mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) &= \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \mathbb{E}(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&\quad + \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_3, \mathbf{u}_3}^{l_3}) \mathbb{E}(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&\quad + \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_4, \mathbf{u}_4}^{l_4}) \mathbb{E}(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3}) + \kappa_4
\end{aligned}$$

where κ_4 is the fourth order cumulant of the distribution of $\{\xi_{j_1, \mathbf{u}_1}^{l_1}, \xi_{j_2, \mathbf{u}_2}^{l_2}, \xi_{j_3, \mathbf{u}_3}^{l_3}, \xi_{j_4, \mathbf{u}_4}^{l_4}\}$. Moreover when $\{\xi_{j, \mathbf{u}}^l\}$ is Gaussian, as in this case, $\kappa_4 \equiv 0$. (See Priestley (Section 5.3, 1981) for further details.)

Using this quadrivariate decomposition, the expression of $\mathbb{E}((d_{j,\mathbf{p}}^l)^4)$ simplifies to

$$\begin{aligned}
\mathbb{E}((d_{j,\mathbf{p}}^l)^4) &= \prod_{i=1}^4 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j,\mathbf{p}}^l(\mathbf{r}_i) \{ \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \mathbb{E}(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&\quad + \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_3, \mathbf{u}_3}^{l_3}) \mathbb{E}(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_4, \mathbf{u}_4}^{l_4}) + \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_4, \mathbf{u}_4}^{l_4}) \mathbb{E}(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3}) \} \\
&= I_1 + I_2 + I_3,
\end{aligned}$$

where, for example,

$$I_1 = \prod_{i=1}^4 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \mathbb{E}(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j,\mathbf{p}}^l(\mathbf{r}_i). \quad (\text{A.16})$$

By construction

$$\begin{aligned}
\mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) &= \text{Cov}(\xi_{j_1, \mathbf{u}_1}^{l_1}, \xi_{j_2, \mathbf{u}_2}^{l_2}) \\
&= \delta_{j_1, j_2} \delta_{\mathbf{u}_1, \mathbf{u}_2} \delta_{l_1, l_2}.
\end{aligned}$$

Hence (A.16) simplifies as follows:

$$\begin{aligned}
I_1 &= \prod_{i=1}^2 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_i, \mathbf{p}}^l(\mathbf{r}_i) \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \\
&\quad \times \prod_{i=3}^4 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_i, \mathbf{p}}^l(\mathbf{r}_i) \mathbb{E}(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_1, \mathbf{p}}^l(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_1, \mathbf{p}}^l(\mathbf{r}_2) \\
&\quad \sum_{l_3} \sum_{j_3} \sum_{\mathbf{u}_3} (w_{j_3, \mathbf{u}_3}^{l_3})^2 \sum_{\mathbf{r}_3} \psi_{j_3, \mathbf{u}_3}^{l_3}(\mathbf{r}_3) \psi_{j_3, \mathbf{p}}^l(\mathbf{r}_3) \sum_{\mathbf{r}_4} \psi_{j_3, \mathbf{u}_3}^{l_3}(\mathbf{r}_4) \psi_{j_3, \mathbf{p}}^l(\mathbf{r}_4) \\
&= \left[\sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_1, \mathbf{p}}^l(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_1, \mathbf{p}}^l(\mathbf{r}_2) \right]^2 \\
&= \mathbb{E}(I_{j, \mathbf{p}}^l)^2 \quad (\text{by recognition from formula (A.12)}) \\
&= I_2 \quad \text{and} \quad I_3.
\end{aligned}$$

Thus, (changing to $\eta(j, l)$ notation)

$$\begin{aligned}
\text{Var}(I_{\eta, \mathbf{p}}) &= 3\mathbb{E}(I_{\eta, \mathbf{p}})^2 \Leftrightarrow \mathbb{E}(I_{\eta, \mathbf{p}})^2 \\
&= 2\mathbb{E}(I_{\eta, \mathbf{p}})^2.
\end{aligned}$$

However, from Theorem 3.2, we know that

$$\mathbb{E}(I_{\eta, \mathbf{p}}) = \sum_{\eta_1} S_{\eta_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) A_{\eta, \eta_1} + O\left(\frac{1}{\min\{R, S\}} \right).$$

Hence,

$$\begin{aligned}
\text{Var}(I_{\eta, \mathbf{p}}) &= 2\mathbb{E}(I_{\eta, \mathbf{p}})^2 \\
&= 2 \left\{ \sum_{\eta_1} A_{\eta, \eta_1} S_{\eta_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) + O\left(\frac{1}{\min\{R, S\}} \right) \right\}^2.
\end{aligned}$$

From the work of Nason *et al.* (2000) it is known that $\Psi_j^l(\boldsymbol{\tau}) = O(1)$, uniformly in $\boldsymbol{\tau}$.

Hence it follows that

$$A_{\eta(j, l), \eta(j_1, l_1)} = \sum_{\boldsymbol{\tau}} \Psi_j^l(\boldsymbol{\tau}) \Psi_{j_1}^{l_1}(\boldsymbol{\tau}) = O(2^{2j(\eta)}).$$

Thus, as η is fixed

$$\text{Var}(I_{\eta, \mathbf{p}}) = 2 \left\{ \sum_{\eta_1} A_{\eta, \eta_1} S_{\eta_1} \left(\frac{\mathbf{p}}{\mathbf{R}} \right) \right\}^2 + O\left(\frac{2^{2j(\eta)}}{\min\{R, S\}} \right).$$

□

Covariance:

$$\begin{aligned} \text{Cov}(I_{j_a, \mathbf{p}}^{l_a}, I_{j_b, \mathbf{q}}^{l_b}) &= \text{Cov}\left(\left(d_{j_a, \mathbf{p}}^{l_a}\right)^2, \left(d_{j_b, \mathbf{q}}^{l_b}\right)^2\right) \\ &= \mathbb{E}\left(\left(d_{j_a, \mathbf{p}}^{l_a}\right)^2 \left(d_{j_b, \mathbf{q}}^{l_b}\right)^2\right) \Leftrightarrow \mathbb{E}\left(\left(d_{j_a, \mathbf{p}}^{l_a}\right)^2\right) \mathbb{E}\left(\left(d_{j_b, \mathbf{q}}^{l_b}\right)^2\right) \end{aligned}$$

We already know the form of $\mathbb{E}\left(\left(d_{j, \mathbf{p}}^l\right)^2\right)$. Hence we focus on the term

$$\begin{aligned} \mathbb{E}\left(\left(d_{j_a, \mathbf{p}}^{l_a}\right)^2 \left(d_{j_b, \mathbf{q}}^{l_b}\right)^2\right) &= \mathbb{E}\left(\left(\sum_{\mathbf{r}} X_{\mathbf{r}} \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r})\right)^2 \left(\sum_{\mathbf{s}} X_{\mathbf{s}} \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s})\right)^2\right) \\ &= \mathbb{E}\left(\left(\sum_{\mathbf{r}} \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} w_{j_1, \mathbf{u}_1}^{l_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}) \xi_{j_1, \mathbf{u}_1}^{l_1} \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r})\right)^2\right. \\ &\quad \left. \times \left(\sum_{\mathbf{s}} \sum_{l_2} \sum_{j_2} \sum_{\mathbf{u}_2} w_{j_2, \mathbf{u}_2}^{l_2} \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{s}) \xi_{j_2, \mathbf{u}_2}^{l_2} \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s})\right)^2\right) \\ &= \prod_{i=1}^2 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_i) \\ &\quad \times \prod_{i=3}^4 \sum_{\mathbf{s}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{s}_i) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_i) \\ &\quad \times \mathbb{E}\left(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}\right) \end{aligned}$$

Using Isserlis' theorem, together with the fact that the fourth order joint cumulant of Gaussian random variables is zero, we can expand the above expression as follows:

$$\begin{aligned} \mathbb{E}\left(\left(d_{j_a, \mathbf{p}}^{l_a}\right)^2 \left(d_{j_b, \mathbf{q}}^{l_b}\right)^2\right) &= \prod_{i=1}^2 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_i) \\ &\quad \times \prod_{i=3}^4 \sum_{\mathbf{s}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{s}_i) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_i) \\ &\quad \times \left\{ \mathbb{E}\left(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}\right) \mathbb{E}\left(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}\right) + \mathbb{E}\left(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_3, \mathbf{u}_3}^{l_3}\right) \mathbb{E}\left(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_4, \mathbf{u}_4}^{l_4}\right) \right. \\ &\quad \left. + \mathbb{E}\left(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_4, \mathbf{u}_4}^{l_4}\right) \mathbb{E}\left(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_3, \mathbf{u}_3}^{l_3}\right) \right\} \\ &= I_1 + I_2 + I_3. \end{aligned}$$

Now recall that by construction $\mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) = \delta_{j_1, j_2} \delta_{\mathbf{u}_1, \mathbf{u}_2} \delta_{l_1, l_2}$. It therefore follows

that:

$$\begin{aligned}
I_1 &= \prod_{i=1}^2 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_i) \\
&\quad \times \prod_{i=3}^4 \sum_{\mathbf{s}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{s}_i) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_i) \\
&\quad \times \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_2, \mathbf{u}_2}^{l_2}) \mathbb{E}(\xi_{j_3, \mathbf{u}_3}^{l_3} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_2) \\
&\quad \sum_{l_3} \sum_{j_3} \sum_{\mathbf{u}_3} (w_{j_3, \mathbf{u}_3}^{l_3})^2 \sum_{\mathbf{s}_3} \psi_{j_3, \mathbf{u}_3}^{l_3}(\mathbf{s}_3) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_3) \sum_{\mathbf{s}_4} \psi_{j_3, \mathbf{u}_3}^{l_3}(\mathbf{s}_4) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_4).
\end{aligned}$$

However, recall from equation (A.12) that

$$\mathbb{E}(I_{j_a, \mathbf{p}}^{l_a}) = \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_2)$$

Hence, $I_1 = \mathbb{E}(I_{j_a, \mathbf{p}}^{l_a}) \mathbb{E}(I_{j_b, \mathbf{q}}^{l_b})$. Furthermore,

$$\begin{aligned}
I_2 &= \prod_{i=1}^2 \sum_{\mathbf{r}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{r}_i) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_i) \\
&\quad \times \prod_{i=3}^4 \sum_{\mathbf{s}_i} \sum_{l_i} \sum_{j_i} \sum_{\mathbf{u}_i} w_{j_i, \mathbf{u}_i}^{l_i} \psi_{j_i, \mathbf{u}_i}^{l_i}(\mathbf{s}_i) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_i) \\
&\quad \times \mathbb{E}(\xi_{j_1, \mathbf{u}_1}^{l_1} \xi_{j_3, \mathbf{u}_3}^{l_3}) \mathbb{E}(\xi_{j_2, \mathbf{u}_2}^{l_2} \xi_{j_4, \mathbf{u}_4}^{l_4}) \\
&= \sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_1) \sum_{\mathbf{s}_3} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{s}_3) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_3) \\
&\quad \sum_{l_2} \sum_{j_2} \sum_{\mathbf{u}_2} (w_{j_2, \mathbf{u}_2}^{l_2})^2 \sum_{\mathbf{r}_2} \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{r}_2) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_2) \sum_{\mathbf{s}_4} \psi_{j_2, \mathbf{u}_2}^{l_2}(\mathbf{s}_4) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{s}_4) \\
&= \left[\sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{r}_2) \right]^2.
\end{aligned}$$

Finally, it is easily shown that $I_3 = I_2$.

Drawing our expressions for I_1 , I_2 and I_3 together we find that,

$$\begin{aligned}
\text{Cov}(I_{j_a, \mathbf{p}}^{l_a}, I_{j_b, \mathbf{q}}^{l_b}) &= \mathbb{E}(I_{j_a, \mathbf{p}}^{l_a} I_{j_b, \mathbf{q}}^{l_b}) \Leftrightarrow \mathbb{E}(I_{j_a, \mathbf{p}}^{l_a}) \mathbb{E}(I_{j_b, \mathbf{q}}^{l_b}) \\
&= I_1 + I_2 + I_3 \Leftrightarrow \mathbb{E}(I_{j_a, \mathbf{p}}^{l_a}) \mathbb{E}(I_{j_b, \mathbf{q}}^{l_b}) \\
&= 2 \left[\sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \sum_{\mathbf{r}_1} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_1) \psi_{j_a, \mathbf{p}}^{l_a}(\mathbf{r}_1) \sum_{\mathbf{r}_2} \psi_{j_1, \mathbf{u}_1}^{l_1}(\mathbf{r}_2) \psi_{j_b, \mathbf{q}}^{l_b}(\mathbf{r}_2) \right]^2 \\
&= 2 \left[\sum_{l_1} \sum_{j_1} \sum_{\mathbf{u}_1} (w_{j_1, \mathbf{u}_1}^{l_1})^2 \alpha_{j_1, j_a}^{l_1, l_a}(\mathbf{u}_1, \mathbf{p}) \alpha_{j_1, j_b}^{l_1, l_b}(\mathbf{u}_1, \mathbf{q}) \right]^2.
\end{aligned}$$

□

Proof of Proposition 3.3

Using the LS2W process representation in (3.6),

$$\begin{aligned} C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) &= \text{Cov}(X_{[\mathbf{zR}]}, X_{[\mathbf{zR}]+\boldsymbol{\tau}}) \\ &= \mathbb{E} \left((X_{[\mathbf{zR}]} \Leftrightarrow \mu_{[\mathbf{zR}]}) (X_{[\mathbf{zR}]+\boldsymbol{\tau}} \Leftrightarrow \mu_{[\mathbf{zR}]+\boldsymbol{\tau}}) \right). \end{aligned}$$

However, by Assumption 1, $\mathbb{E}(X_{\mathbf{r}}) = 0$ for all \mathbf{r} . Hence,

$$\begin{aligned} C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) &= \mathbb{E} (X_{[\mathbf{zR}]} X_{[\mathbf{zR}]+\boldsymbol{\tau}}) \\ &= \mathbb{E} \left(\sum_l \sum_j \sum_{\mathbf{u}} w_{j,\mathbf{u}}^l \psi_{j,\mathbf{u}}^l([\mathbf{zR}]) \xi_{j,\mathbf{u}}^l \sum_{l_0} \sum_{j_0} \sum_{\mathbf{u}_0} w_{j_0,\mathbf{u}_0}^{l_0} \psi_{j_0,\mathbf{u}_0}^{l_0}([\mathbf{zR}] + \boldsymbol{\tau}) \xi_{j_0,\mathbf{u}_0}^{l_0} \right) \\ &= \mathbb{E} \left(\sum_l \sum_j \sum_{\mathbf{u}} \sum_{l_0} \sum_{j_0} \sum_{\mathbf{u}_0} w_{j,\mathbf{u}}^l w_{j_0,\mathbf{u}_0}^{l_0} \psi_{j,\mathbf{u}}^l([\mathbf{zR}]) \psi_{j_0,\mathbf{u}_0}^{l_0}([\mathbf{zR}] + \boldsymbol{\tau}) \xi_{j,\mathbf{u}}^l \xi_{j_0,\mathbf{u}_0}^{l_0} \right) \\ &= \sum_l \sum_j \sum_{\mathbf{u}} \sum_{l_0} \sum_{j_0} \sum_{\mathbf{u}_0} w_{j,\mathbf{u}}^l w_{j_0,\mathbf{u}_0}^{l_0} \psi_{j,\mathbf{u}}^l([\mathbf{zR}]) \psi_{j_0,\mathbf{u}_0}^{l_0}([\mathbf{zR}] + \boldsymbol{\tau}) \mathbb{E}(\xi_{j,\mathbf{u}}^l \xi_{j_0,\mathbf{u}_0}^{l_0}) \end{aligned}$$

for the $w_{j,\mathbf{u}}^l$ and the $\psi_{j,\mathbf{u}}^l$ are deterministic. Moreover, since

$$\text{Cov}(\xi_{j,\mathbf{u}}^l, \xi_{j_0,\mathbf{u}_0}^{l_0}) = \mathbb{E}(\xi_{j,\mathbf{u}}^l \xi_{j_0,\mathbf{u}_0}^{l_0}) = \delta_{j,j_0} \delta_{l,l_0} \delta_{\mathbf{u},\mathbf{u}_0},$$

it follows that

$$\begin{aligned} C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) &= \sum_l \sum_j \sum_{\mathbf{u}} |w_{j,\mathbf{u}}^l|^2 \psi_{j,\mathbf{u}}^l([\mathbf{zR}]) \psi_{j,\mathbf{u}}^l([\mathbf{zR}] + \boldsymbol{\tau}), \quad \text{next let } \mathbf{u} = \mathbf{p} + [\mathbf{zR}] \\ &= \sum_l \sum_j \sum_{\mathbf{u}} |w_{j,\mathbf{p}+[\mathbf{zR}]}^l|^2 \psi_{j,\mathbf{p}+[\mathbf{zR}]}^l([\mathbf{zR}]) \psi_{j,\mathbf{p}+[\mathbf{zR}]}^l([\mathbf{zR}] + \boldsymbol{\tau}) \\ &= \sum_l \sum_j \sum_{\mathbf{u}} |w_{j,\mathbf{p}+[\mathbf{zR}]}^l|^2 \psi_{j,\mathbf{p}}^l(\mathbf{0}) \psi_{j,\mathbf{p}}^l(\boldsymbol{\tau}). \end{aligned}$$

We now derive two limit results which are required to complete this proof.

Limit result 1 By Definition 3.5, $S_j^l(\mathbf{z}) = |W_j^l(\mathbf{z})|^2$ for all $\mathbf{z} \in (0, 1)^2$. Furthermore, Property 3.1 states that

$$S_j^l(\mathbf{z}) = \lim_{R,S \rightarrow \infty} |w_{j,[\mathbf{zR}]}^l|^2 \quad \text{for } \mathbf{z} \in (0, 1)^2.$$

By Assumption 2,

$$\sup_{\mathbf{u}} \left| w_{j,\mathbf{u}}^l \Leftrightarrow W_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right| \leq \frac{C_j^l}{\max\{R, S\}}.$$

The triangle inequality implies that

$$\begin{aligned}
\sup_{\mathbf{u}} |w_{j,\mathbf{u}}^l| &\Leftrightarrow \left| W_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right| \leq \frac{C_j^l}{\max\{R, S\}} \\
&\Rightarrow |w_{j,\mathbf{u}}^l| = \left| W_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right| + O \left(\frac{C_j^l}{\max\{R, S\}} \right) \\
&\Rightarrow |w_{j,\mathbf{u}}^l|^2 = \left| W_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} \right) \right|^2 + O \left(\frac{C_j^l}{\max\{R, S\}} \right) \quad \text{as } \sum_j |W_j^l(\mathbf{z})|^2 < \infty.
\end{aligned}$$

Hence, setting $\mathbf{z} = \mathbf{u}/\mathbf{R}$, we obtain

$$\left| |w_{j,\mathbf{u}}^l|^2 \Leftrightarrow S_j^l(\mathbf{z}) \right| = O \left(\frac{C_j^l}{\max\{R, S\}} \right). \quad (\text{A.17})$$

Limit result 2 Recall that the $W_j^l(\mathbf{z})$ are assumed to be Lipschitz continuous functions (with respect to the L_1 -norm). Hence,

$$\begin{aligned}
\|W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R}) \Leftrightarrow W_j^l(\mathbf{z})\| &\leq L_j^l \|\mathbf{z} + \boldsymbol{\tau}/\mathbf{R} \Leftrightarrow \mathbf{z}\| \quad \text{where } \boldsymbol{\tau}/\mathbf{R} = (\tau_1/R, \tau_2/S) \\
\Rightarrow |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R}) \Leftrightarrow W_j^l(\mathbf{z})| &\leq L_j^l \|\boldsymbol{\tau}/\mathbf{R}\|_1 \\
\Rightarrow |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})| \Leftrightarrow |W_j^l(\mathbf{z})| &\leq L_j^l \|\boldsymbol{\tau}/\mathbf{R}\|_1 \quad \text{by the triangle inequality} \\
&\Rightarrow |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})| = |W_j^l(\mathbf{z})| + O(L_j^l \|\boldsymbol{\tau}/\mathbf{R}\|_1) \\
&\Rightarrow |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})|^2 = |W_j^l(\mathbf{z})|^2 + O(L_j^l \|\boldsymbol{\tau}/\mathbf{R}\|_1)
\end{aligned}$$

for $\sum_j \sum_l |W_j^l(\mathbf{z})|^2 < \infty$ and the L_j^l are uniformly bounded in (j, l) . Hence

$$\begin{aligned}
\left| |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})|^2 \Leftrightarrow |W_j^l(\mathbf{z})|^2 \right| &= O \left(L_j^l \left(\frac{|\tau_1|}{R} + \frac{|\tau_2|}{S} \right) \right) \\
\Rightarrow \left| |W_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})|^2 \Leftrightarrow |W_j^l(\mathbf{z})|^2 \right| &= O \left(L_j^l \left(\frac{|\tau_1| + |\tau_2|}{\min\{R, S\}} \right) \right).
\end{aligned}$$

Thus,

$$\left| S_j^l(\mathbf{z} + \boldsymbol{\tau}/\mathbf{R})^2 \Leftrightarrow S_j^l(\mathbf{z})^2 \right| = O \left(L_j^l \left(\frac{|\tau_1| + |\tau_2|}{\min\{R, S\}} \right) \right). \quad (\text{A.18})$$

With the above limit results in place, we are now in a position to consider the asymptotic convergence of $C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau})$ to $C(\mathbf{z}, \boldsymbol{\tau})$:

$$\begin{aligned}
|C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| &= \left| \sum_l \sum_j \sum_{\mathbf{u}} |w_{j,\mathbf{u}+\mathbf{z}\mathbf{R}}^l|^2 \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \Leftrightarrow \sum_l \sum_j S_j^l(\mathbf{z}) \Psi_j^l(\boldsymbol{\tau}) \right| \\
&= \left| \sum_l \sum_j \sum_{\mathbf{u}} \left(|w_{j,\mathbf{u}+\mathbf{z}\mathbf{R}}^l|^2 \Leftrightarrow S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \right) \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \right. \\
&\quad \left. + \sum_l \sum_j \sum_{\mathbf{u}} S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \Leftrightarrow \sum_{l,j} S_j^l(\mathbf{z}) \Psi_j^l(\boldsymbol{\tau}) \right| \\
&\leq \left| \sum_l \sum_j \sum_{\mathbf{u}} \left(|w_{j,\mathbf{u}+\mathbf{z}\mathbf{R}}^l|^2 \Leftrightarrow S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \right) \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \right| \\
&\quad + \left| \sum_l \sum_j \sum_{\mathbf{u}} S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \Leftrightarrow \sum_{l,j} S_j^l(\mathbf{z}) \Psi_j^l(\boldsymbol{\tau}) \right|.
\end{aligned}$$

However $\Psi_j^l(\boldsymbol{\tau}) = \sum_{\mathbf{u}} \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})$. Hence, using limit result 1

$$\begin{aligned}
|C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| &\leq \sum_l \sum_j \sum_{\mathbf{u}} \frac{C_j^l}{\max\{R, S\}} |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\
&\quad + \left| \sum_l \sum_j \sum_{\mathbf{u}} S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \right. \\
&\quad \left. \Leftrightarrow \sum_l \sum_j S_j^l(\mathbf{z}) \sum_{\mathbf{u}} \psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau}) \right| \\
&\leq \sum_l \sum_j \sum_{\mathbf{u}} \frac{C_j^l}{\max\{R, S\}} |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\
&\quad + \sum_l \sum_j \sum_{\mathbf{u}} \left| S_j^l \left(\frac{\mathbf{u}}{\mathbf{R}} + \mathbf{z} \right) \Leftrightarrow S_j^l(\mathbf{z}) \right| |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})|.
\end{aligned}$$

Using Limit Result 2, in conjunction with the modelling assumptions made in equations (3.10) and (3.12), we obtain

$$\begin{aligned}
|C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| &= \sum_l \sum_j \sum_{\mathbf{u}} \frac{C_j^l}{\max\{R, S\}} |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\
&\quad + \sum_j \sum_l \sum_{\mathbf{u}} L_j^l \frac{|u_1| + |u_2|}{\min\{R, S\}} |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\
&\leq \sum_l \sum_j \sum_{\mathbf{u}} \left[\frac{C_j^l + L_j^l (|u_1| + |u_2|)}{\min\{R, S\}} \right] |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\
&\quad + O\left(\frac{1}{\min\{R, S\}}\right).
\end{aligned}$$

□

Proof of Lemma 3.2

This proof is identical to that of the one-dimensional case considered by Nason *et al.* (Proposition 2, 2000) and is included for completeness. Consider,

$$\sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\boldsymbol{\tau}} C(\mathbf{z}, \boldsymbol{\tau}) \Psi_{\eta}(\boldsymbol{\tau})$$

By definition, $C(\mathbf{z}, \boldsymbol{\tau}) = \sum_{\nu} S_{\nu} \Psi_{\nu}(\boldsymbol{\tau})$. Hence

$$\begin{aligned} \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\boldsymbol{\tau}} C(\mathbf{z}, \boldsymbol{\tau}) \Psi_{\eta}(\boldsymbol{\tau}) &= \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\boldsymbol{\tau}} \left\{ \sum_{\nu} S_{\nu}(\mathbf{z}) \Psi_{\nu}(\boldsymbol{\tau}) \right\} \Psi_{\eta}(\boldsymbol{\tau}) \\ &= \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\nu} S_{\nu}(\mathbf{z}) \sum_{\boldsymbol{\tau}} \Psi_{\nu}(\boldsymbol{\tau}) \Psi_{\eta_1}(\boldsymbol{\tau}). \end{aligned}$$

The order of the summations may be changed above for $\sum_{\eta} S_{\eta}(\mathbf{z}) < \infty \forall \mathbf{z}$ whilst the sum over $\boldsymbol{\tau}$ is finite. By definition $\sum_{\boldsymbol{\tau}} \Psi_{\nu}(\boldsymbol{\tau}) \Psi_{\eta_1}(\boldsymbol{\tau}) = A_{\nu, \eta_1} = A_{\nu, \eta_1}$. Hence,

$$\begin{aligned} \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\boldsymbol{\tau}} C(\mathbf{z}, \boldsymbol{\tau}) \Psi_{\eta}(\boldsymbol{\tau}) &= \sum_{\eta_1} A_{\eta, \eta_1}^{-1} \sum_{\nu} S_{\nu}(\mathbf{z}) A_{\nu, \eta_1} \\ &= \sum_{\nu} S_{\nu}(\mathbf{z}) \sum_{\eta_1} A_{\eta, \eta_1}^{-1} A_{\eta_1, \nu} \\ &= \sum_{\nu} S_{\nu}(\mathbf{z}) \delta_{\eta, \nu} \\ &= S_{\eta}(\mathbf{z}). \end{aligned}$$

□

Proof of Proposition 3.4

To prove part (a), define $\kappa(\boldsymbol{\tau}) = \{k_{\eta}(\boldsymbol{\tau})\}$ via $\kappa(\boldsymbol{\tau}) = A^{-1} \Psi(\boldsymbol{\tau})$. Assuming Conjecture 3.1, then $\sum_{\eta} |\kappa_{\eta}(\boldsymbol{\tau})| < \infty$ uniformly in $\boldsymbol{\tau}$. Hence, by Lemma 3.2 we can relate the autocovariance to a sequence of LWS $\{S_{\eta}(\mathbf{z})\}$ which satisfy $\sum_j \sum_l S_j^l(\mathbf{z}) < \infty$.

To prove part (b) we consider the following result which demonstrates the absolute summability of $C(z, \boldsymbol{\tau})$ in $\boldsymbol{\tau}$ and, consequently, that the process autocovariance is also absolutely summable.

Corollary A.1

Assume that $\sum_{\eta} 2^{j(\eta)} S_{\eta}(z) < \infty$ uniformly in \mathbf{z} . Then $\sum_{\boldsymbol{\tau}} |C(\mathbf{z}, \boldsymbol{\tau})| < \infty$ uniformly in \mathbf{z} and

$$\sum_{\boldsymbol{\tau}} |C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| = o(1), \quad \text{as } \eta \rightarrow \infty$$

uniformly in \mathbf{z} .

Proof Let $C_{J_0}(\mathbf{z}, \boldsymbol{\tau}) = \sum_{\eta=1}^{3J_0} S_\eta(\mathbf{z}) \Psi_\eta(\boldsymbol{\tau})$. Then, as it follows from the work of Nason *et al.* (2000) that $\Psi_\eta(\boldsymbol{\tau}) = O(1)$ uniformly in $\boldsymbol{\tau}$, we therefore obtain $\sum_{\boldsymbol{\tau}} |\Psi_\eta(\boldsymbol{\tau})| = O(2^{2j(\eta)})$. Thus,

$$\begin{aligned} \sum_{\boldsymbol{\tau}} |C(\mathbf{z}, \boldsymbol{\tau})| &\leq \sum_{\boldsymbol{\tau}} \left| \sum_{\eta} S_\eta(\mathbf{z}) \Psi_\eta(\boldsymbol{\tau}) \right| \\ &\leq \sum_{\boldsymbol{\tau}} \sum_{\eta} S_\eta(\mathbf{z}) |\Psi_\eta(\boldsymbol{\tau})| \\ &\leq \sum_{\eta} S_\eta(\mathbf{z}) \sum_{\boldsymbol{\tau}} |\Psi_\eta(\boldsymbol{\tau})| \\ &\leq c \sum_{\eta} S_\eta(\mathbf{z}) 2^{2j(\eta)} < \infty. \end{aligned}$$

Furthermore,

$$\begin{aligned} \sum_{\boldsymbol{\tau}} |C_{J_0}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau})| &\leq \sum_{\boldsymbol{\tau}} \sum_l \sum_j \sum_{\mathbf{u}} \left[\frac{C_j^l + L_j^L(|u_1| + |u_2|)}{\min\{R, S\}} \right] |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\ &\leq \frac{k}{\min\{R, S\}} \sum_l \sum_j \sum_{\boldsymbol{\tau}} \sum_{\mathbf{u}} |\psi_{j,\mathbf{u}}^l(\mathbf{0}) \psi_{j,\mathbf{u}}^l(\boldsymbol{\tau})| \\ &= O\left(\frac{2^{2J_0}}{\min\{R, S\}}\right). \end{aligned}$$

Additionally,

$$\begin{aligned} \sum_{\boldsymbol{\tau}} |C_{J_0}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| &= \sum_{\boldsymbol{\tau}} \left| \sum_l \sum_{j=J_0+1}^{\infty} S_j^l(\mathbf{z}) \Psi_j^l(\boldsymbol{\tau}) \right| \\ &\leq \sum_{\boldsymbol{\tau}} \sum_l \sum_{j=J_0+1}^{\infty} S_j^L(\mathbf{z}) |\Psi_j^l(\boldsymbol{\tau})| \\ &= \sum_l \sum_{j=J_0+1}^{\infty} S_j^l(\mathbf{z}) \sum_{\boldsymbol{\tau}} |\Psi_j^l(\boldsymbol{\tau})| \\ &\leq k \sum_l \sum_{j=J_0+1}^{\infty} 2^{2j} S_j^l(\mathbf{z}) \\ &= o(1). \end{aligned}$$

Hence $\sum_{\boldsymbol{\tau}} |C_{\mathbf{R}}(\mathbf{z}, \boldsymbol{\tau}) \Leftrightarrow C(\mathbf{z}, \boldsymbol{\tau})| = o(1)$ as $\min\{R, S\} \rightarrow \infty$. \square

Thus, as a consequence of the above and Proposition 3.3 we have the desired result for those processes whose LWS are constant over location. \square

Appendix B

Chapter 5 Proofs

Proof of Proposition 5.2

By Definitions 2.5 and 2.6

$$\Psi_{j+1}(\tau) = \sum_l \psi_{(j+1)l} \psi_{(j+1)(l-\tau)} \quad (\text{B.1})$$

$$= \sum_l \left(\sum_k h_{l-2k} \psi_{jk} \right) \left(\sum_m h_{l-\tau-2m} \psi_{jm} \right), \quad (\text{B.2})$$

Case A: (even argument). Now

$$\begin{aligned} \Psi_{j+1}(2\tau) &= \sum_l \sum_k h_{l-2k} \psi_{jk} \sum_m h_{l-2(\tau+m)} \psi_{jm} \quad (\text{then let } r = m + \tau) \\ &= \sum_l \sum_k h_{l-2k} \psi_{jk} \sum_r h_{l-2r} \psi_{j(r-\tau)} \\ &= \sum_k \psi_{jk} \sum_r \psi_{j(r-\tau)} \sum_m h_m h_{m+2(k-r)} \\ &= \sum_k \psi_{jk} \sum_r \psi_{j(r-\tau)} \delta_{k-r,0} \quad (\text{by equation 5.1.39 of Daubechies (1992)}) \\ &= \Psi_j(\tau). \end{aligned}$$

Case B: (odd argument). Now

$$\begin{aligned} \Psi_{j+1}(2\tau + 1) &= \sum_l \sum_k h_{l-2k} \psi_{jk} \sum_m h_{l-1-2(\tau+m)} \psi_{jm} \quad (\text{again let } r = m + \tau) \\ &= \sum_k \psi_{jk} \sum_r \psi_{j(r-\tau)} \sum_l h_{l-2k} h_{l-1-2r} \\ &= \sum_k \psi_{jk} \sum_r \psi_{j(r-\tau)} \Phi_1 \{2(r \Leftrightarrow k) + 1\} \quad (\text{now let } p = r \Leftrightarrow k) \\ &= \sum_p \Phi_1(2p + 1) \Psi_j(\tau \Leftrightarrow p). \end{aligned}$$

The limits of the above are obtained by considering the supports of $\Phi_1(2p+1)$ and $\Psi_j(\tau \Leftrightarrow p)$.

Proof of Proposition 5.1

First, consider the case of $P_{j,n}$ when n is even. By definition:

$$\begin{aligned} P_{j,2p} &= \sum_k \Psi_j(k) \Psi_j(k \Leftrightarrow 2p) \\ &= \sum_{k \text{ (even)}} \Psi_j(k) \Psi_j(k \Leftrightarrow 2p) + \sum_{k \text{ (odd)}} \Psi_j(k) \Psi_j(k \Leftrightarrow 2p) \\ &= \sum_l \Psi_j(2l) \Psi_j(2(l \Leftrightarrow p)) + \sum_l \Psi_j(2l+1) \Psi_j(2(l \Leftrightarrow p) + 1). \end{aligned}$$

Applying Proposition 5.2, the above expression can be re-written as

$$P_{j,2p} = P_{j-1,p} + \sum_r \sum_s \Phi_1(2r+1) \Phi_1(2s+1) \sum_l \Psi_{j-1}(l \Leftrightarrow r) \Psi_{j-1}(l \Leftrightarrow p \Leftrightarrow s).$$

Setting $q = l \Leftrightarrow r$, we obtain:

$$\begin{aligned} P_{j,2p} &= P_{j-1,p} + \sum_r \sum_s \Phi_1(2r+1) \Phi_1(2s+1) \sum_q \Psi_{j-1}(q) \Psi_{j-1}(q \Leftrightarrow (s+p \Leftrightarrow r)) \\ &= P_{j-1,p} + \sum_r \sum_s \Phi_1(2r+1) \Phi_1(2s+1) P_{j-1,s+p-r}. \end{aligned}$$

Finally, on making the substitution $q = s \Leftrightarrow r$ and applying Definition 5.1, the above can be simplified to

$$\begin{aligned} P_{j,2p} &= P_{j-1,p} + \sum_r \sum_q \Phi_1(2r+2q+1) \Phi_1(2r+1) P_{j-1,p+q} \\ &= P_{j-1,p} + \sum_q P_{j-1,p+q} Q_q, \end{aligned} \tag{B.3}$$

as required.

We now consider the case of $P_{j,n}$ when n is odd. By Definition 5.2,

$$\begin{aligned} P_{j,2p+1} &= \sum_k \Psi_j(k) \Psi_j(k \Leftrightarrow (2p+1)) \\ &= \sum_{k \text{ (even)}} \Psi_j(k) \Psi_j(k \Leftrightarrow 2p \Leftrightarrow 1) + \sum_{k \text{ (odd)}} \Psi_j(k) \Psi_j(k \Leftrightarrow 2p \Leftrightarrow 1) \\ &= \sum_l \Psi_j(2l) \Psi_j(2(l \Leftrightarrow p) \Leftrightarrow 1) + \sum_l \Psi_j(2l+1) \Psi_j(2(l \Leftrightarrow p)). \end{aligned}$$

Again, applying Proposition 5.2, the above can be re-expressed as

$$\begin{aligned} P_{j,2p+1} &= \sum_r \Phi_1(2r+1) \sum_l \Psi_{j-1}(l) \Psi_{j-1}(l \Leftrightarrow p \Leftrightarrow r) \\ &+ \sum_q \Phi_1(2q+1) \sum_l \Psi_{j-1}(l \Leftrightarrow p) \Psi_{j-1}(l \Leftrightarrow q). \end{aligned} \quad (\text{B.4})$$

Finally, on making the substitution $s = l \Leftrightarrow q$ in the second double sum of B.4 and then applying Definition 5.2, we obtain:

$$\begin{aligned} P_{j,2p+1} &= \sum_r \Phi_1(2r+1) \sum_l \Psi_{j-1}(l) \Psi_{j-1}(l \Leftrightarrow p \Leftrightarrow r) \\ &+ \sum_q \Phi_1(2q+1) \sum_s \Psi_{j-1}(s+q \Leftrightarrow p) \Psi_{j-1}(s) \\ &= \sum_r \Phi_1(2r+1) P_{j-1,p+r} + \sum_q \Phi_1(2q+1) P_{j-1,p-q}. \end{aligned}$$

Hence result. \square

Proof of Proposition 5.2

Let $j \in \mathbb{Z}^+$. From Definition 2.7, we know that:

$$\begin{aligned} A_{j+1,j+1} &= \langle \Psi_{j+1}, \Psi_{j+1} \rangle \\ &= \sum_k \Psi_{j+1}(2k) \Psi_{j+1}(2k) + \sum_k \Psi_{j+1}(2k+1) \Psi_{j+1}(2k+1). \end{aligned}$$

Applying Proposition 5.2, we obtain:

$$A_{j+1,j+1} = A_{j,j} + \sum_p \sum_q \Phi_1(2p+1) \Phi_1(2q+1) \sum_k \Psi_j(k \Leftrightarrow p) \Psi_j(k \Leftrightarrow q).$$

Next, make the substitution $r = k \Leftrightarrow p$,

$$\begin{aligned} A_{j+1,j+1} &= A_{j,j} + \sum_{p,q} \Phi_1(2p+1) \Phi_1(2q+1) \sum_r \Psi_j(r) \Psi_j(r+p \Leftrightarrow q) \\ &= A_{j,j} + \sum_p \sum_q \Phi_1(2p+1) \Phi_1(2q+1) P_{j,q-p}. \end{aligned} \quad (\text{B.5})$$

Finally, we make the substitution $r = q \Leftrightarrow p$ in (B.5). This results in

$$\begin{aligned} A_{j+1,j+1} &= A_{j,j} + \sum_r P_{j,r} \sum_p \Phi_1(2p+1) \Phi_1(2(p+r)+1) \\ &= A_{j,j} + \sum_r P_{j,r} Q_r, \quad (\text{by Definition 5.1.}) \end{aligned} \quad (\text{B.6})$$

as required. \square

Proof of Proposition 5.3

This is similar to the proof of Proposition 5.1 in that we consider the cases of $T_j, k, 2p$ and $T_{j,k,2p+1}$ separately. The approach for each case consists of a divide and conquer approach; dividing the summation into odd and even parts, before applying Proposition 5.2 to obtain the desired result. (See Eckley & Nason (2000) for further details). \square

Proof of Proposition 5.4

Proof is similar to that of Proposition 5.2. Recall that $A_{j+1,k+1} = \sum_{\tau} \Psi_{j+1}(\tau)\Psi_{k+1}(\tau)$. By dividing this summation into odd and even parts and then applying Proposition 5.2 we obtain the required result. (See Eckley & Nason (2000) for further details). \square

Proof of Proposition 5.5

As with earlier proofs, this result can be shown by adopting a divide and conquer approach. By Definition 5.4,

$$R_{j,q}^l = \sum_r \Psi_1(2^{j-l+1}r + q + 1)\Psi_j(r) \quad (\text{B.7})$$

where $j, l > 0$. Dividing the summation in (B.7) into odd and even valued arguments, applying Proposition 5.2 and re-arranging the resulting expression, we find that

$$R_{j,q}^l = R_{j-1,q}^{l-2} + \sum_p \Phi_1(2p+1)R_{j-1,2^{j-l+1}+2^{j-l+2}p+q}^{l-2}. \quad (\text{B.8})$$

The limits of the summation in (B.7) can be found by considering the support of the Ψ_1 and Ψ_j term. (See Eckley & Nason (2000) for further details).

Proof of Proposition 5.6

By recalling that

$$A_{1,j+1} = \sum_{\tau} \Psi_1(\tau)\Psi_{j+1}(\tau),$$

dividing the summation into odd and even arguments of τ and using the result that

$$\Psi_1(2\tau) = \sum_k g_k g_{k-2\tau} = \delta_{\tau,0} \quad \forall \tau \in \text{supp}\{\Psi_j\}.$$

it is easily shown that

$$A_{1,j+1} = \Psi_j(0) + \sum_k \Psi_1(2k+1)\Psi_{j+1}(2k+1).$$

However, from the properties of wavelets, we know that

$$\Psi_j(0) = \sum_k \psi_{j,k}^2 = \sum_k g_{j,k}^2 = 1.$$

Hence,

$$A_{1,j+1} = 1 + \sum_k \Psi_1(2k+1)\Psi_{j+1}(2k+1).$$

Using Proposition 5.2, the above expression may be simplified to

$$A_{1,j+1} = 1 + \sum_p \Phi_1(2p+1)R_{j,p}^j \quad \text{as required.} \quad (\text{B.9})$$

□

Appendix C

The LS2W software suite

Motivated by the principle of reproducible research, this appendix contains help pages for various programs which have been written to implement the estimation scheme associated with the LS2W model proposed in Chapter 3. These programs may be downloaded from the following web site,

`http://www.stats.bris.ac.uk/~maiae/LS2W`

where details of other subsidiary routines may be obtained. It should be noted that this suite has been designed to tie in with the freeware package, `WaveThresh`.

D2ACW

Compute 2-D discrete autocorrelation wavelets.

Description

This function computes two-dimensional discrete autocorrelation wavelets. The inner products of these wavelets are required for correction of the (biased) raw wavelet periodograms. See Chapter 3 for further details.

Usage

```
D2ACW(J, filter.number = 1, family = "DaubExPhase", switch =  
"direction", tol = 1e-100, OPLENGTH = 2000)
```

Required arguments

J

Discrete autocorrelation wavelets will be computed for scales 1 to J within each decomposition direction (horizontal, vertical and diagonal). This number should be a positive integer.

Optional arguments

`filter.number`

The index of the wavelet used to compute the discrete autocorrelation wavelets.

`family`

The wavelet family used to compute the discrete autocorrelation wavelets.

`switch`

Allows the user to define how they wish their inner product matrix to be formed. There are two available options:

- `switch = "direction"` - structures the matrix by scale within each decomposition direction. Thus, the ordering goes as follows $(1, V), (2, V), \dots$
- `switch = "level"` - structures the matrix by direction within each scale. Thus the ordering is as follows $(1, V), (1, H), (1, D), (2, V), (2, H), \dots$

`tol`

In the brute force computation for Daubechies' compactly supported wavelets many inner product computations are performed. This tolerance discounts any results which are smaller than `tol` which effectively defines how long the inner product/autocorrelation products are.

`OPLength`

This integer variable defines some workspace of length `OPLength`. The code uses this workspace. If the workspace is not long enough then the routine will stop and tell you what `OPLength` should be set to.

Value

A list containing $3J$ components, numbered from 1 to $3J$. If `switch="direction"`, the first J components contain the vertical autocorrelation wavelet coefficients, the second set of J components contains the horizontal autocorrelation wavelet coefficients (scales $1, \dots, J$) and the last J components constitute the diagonal autocorrelation wavelet coefficients. However, if `switch="level"`, then the first 3 components contain the finest scale autocorrelation wavelet coefficients in the vertical, horizontal and diagonal decomposition directions respectively. The second set of 3 contains the vertical, horizontal and diagonal coefficients at scale 2 etc.

Note that these 2-D autocorrelation wavelets are stored as matrices. The central element of the matrix refers to lag 0.

Side effects

None.

Details

This function computes the 2-D discrete autocorrelation wavelets. It does not have any direct use for space-scale analysis. The construction method is a brute force approach — a more elegant solution would be based on the recursive schemes of Chapter 5. The routine returns only the values of the discrete autocorrelation wavelets, not their spatial positions. Each discrete autocorrelation wavelet is compactly supported. This support is determined from the discrete wavelets upon which these autocorrelations are based.

Examples

```
#
# Let us create the discrete autocorrelation wavelets for the Haar wavelet.
# We shall create up to scale 2.
#
> D2ACW(J=2, filter.number=1, family="DaubExPhase", switch="direction")
[[1]]:
      [,1] [,2] [,3]
[1,] -0.25 -0.5 -0.25
[2,]  0.50  1.0  0.50
[3,] -0.25 -0.5 -0.25

[[2]]:
      [,1] [,2] [,3] [,4] [,5] [,6] [,7]
[1,] -0.0625 -0.125 -0.1875 -0.25 -0.1875 -0.125 -0.0625
[2,] -0.1250 -0.250 -0.3750 -0.50 -0.3750 -0.250 -0.1250
[3,]  0.0625  0.125  0.1875  0.25  0.1875  0.125  0.0625
[4,]  0.2500  0.500  0.7500  1.00  0.7500  0.500  0.2500
[5,]  0.0625  0.125  0.1875  0.25  0.1875  0.125  0.0625
[6,] -0.1250 -0.250 -0.3750 -0.50 -0.3750 -0.250 -0.1250
[7,] -0.0625 -0.125 -0.1875 -0.25 -0.1875 -0.125 -0.0625
#
```

```
#... and the remaining terms follow suit.  
#  
# You can also plot components to get an idea of what the  
# 2-D autocorrelation wavelet looks like. This can be done using the  
# command D2autoplot (see web-site for further details).  
#  
> D2autoplot(J=2, filter.number=1, family="DaubExPhase", direction="3")
```

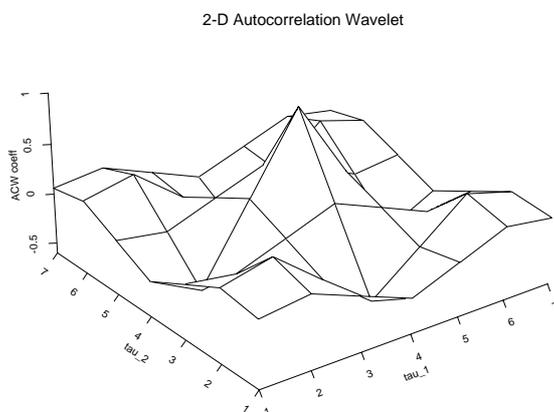


Figure C.1: Scale 2, discrete Haar autocorrelation wavelet in the diagonal decomposition direction.

D2Amat

Creates the A matrix required for analysing LS2W processes.

Description

This function creates the matrix used to correct the raw periodogram of a LS2W process.

Usage

```
D2Amat(J, filter.number=10, family="DaubExPhase", OPLENGTH=2000,  
switch="direction", verbose = F)
```

Required Arguments

J

The level to which the decomposition must extend. This number should be a positive integer.

Optional Arguments

`filter.number`

The index of the wavelet used to compute the correction matrix A.

`family`

The wavelet family used to compute A.

`OPLENGTH`

This integer variable defines some workspace of length `OPLENGTH` which is used by the code. If the workspace is not long enough, then the routine will stop and tell you what `OPLENGTH` should be set to.

switch

Dictates the structure of the matrix [by direction or by scale].

verbose

Allows certain informative messages to be printed on screen.

Value

A matrix of order $3J \times 3J$ containing the elements $A_{j,l}$ defined in Chapter 3. Each element is the sum over all lags of the product of the matrix coefficients of a 2-D DACW matrix at level j_1 in direction l_1 with that of another (not necessarily different) matrix of DACW coefficients at level j_2 in direction l_2 . The structure of this matrix is as follows: the rows and columns of the matrix are labeled $1, \dots, 3J$ in accordance with the notation of Chapter 3. When `switch="direction"` the matrix rows (and columns) are structured as follows:

- Rows (columns) $1, \dots, J$ correspond to the different levels of the decomposition in the vertical direction. 1 = fine and J = coarse scale.
- Rows (columns) $J + 1, \dots, 2J$ correspond to the different levels in the horizontal direction.
- Rows (columns) $2J + 1, \dots, 3J$ correspond to the different directions in the diagonal direction.

A pictorial representation of this structure is displayed in figure 5.1.

When `switch="level"`, the row and column elements cycle as follows:

level 1 vertical, level 1 horizontal, level 1 diagonal, level 2 vertical, etc.

Side effects

None.

Details

See Section 3.5.1 and Chapter 5 for further details about this matrix.

Examples

```
#
# Let's compute the A matrix for the Haar wavelet in 2-D.
#
> D2Amat(J=2, filter.number=1, family="DaubExPhase", switch="direction")
      1      2      3      4      5      6
1 2.2500 1.3125 0.2500 0.3125 0.7500 0.9375
2 1.3125 4.8125 0.3125 0.5625 0.1875 1.3125
3 0.2500 0.3125 2.2500 1.3125 0.7500 0.9375
4 0.3125 0.5625 1.3125 4.8125 0.1875 1.3125
5 0.7500 0.1875 0.7500 0.1875 2.2500 0.5625
6 0.9375 1.3125 0.9375 1.3125 0.5625 3.0625
```

Element (6,6) of the above matrix represents the inner product of Ψ_2^d with itself, whilst the entry in (2,4) represents the inner product of Ψ_2^h with Ψ_2^v . A pictorial representation of the structure of this matrix may be seen on page 133.

```
#
# And now for the same matrix structured by level
#
>D2Amat(J=2, filter.number=1, family="DaubExPhase", switch="level")
      1      2      3      4      5      6
1 2.2500 0.2500 0.7500 1.3125 0.3125 0.9375
2 0.2500 2.2500 0.7500 0.3125 1.3125 0.9375
3 0.7500 0.7500 2.2500 0.1875 0.1875 0.5625
4 1.3125 0.3125 0.1875 4.8125 0.5625 1.3125
5 0.3125 1.3125 0.1875 0.5625 4.8125 1.3125
6 0.9375 0.9375 0.5625 1.3125 1.3125 3.0625
```

cddews

Compute the local wavelet spectrum estimate

Description

This function computes the local wavelet spectrum (LWS) estimate of an image (or non-decimated wavelet transform of a time series). The estimate is computed by taking the non-decimated wavelet transform of the image, squaring the detail coefficients, smoothing using wavelet shrinkage and then correcting the redundancy caused by use of the non-decimated wavelet transform.

Usage

```
cddews(data, filter.number=1, family="DaubExPhase", switch="direction",
correct = T, verbose = F, smooth = T, sm.filter.number = 4., sm.family
= "DaubExPhase", levels = 3.:6., type = "hard", policy = "LSuniversal",
by.level = F, value = 0., dev = var)
```

Required arguments

`data` The image you want to analyse.

Optional arguments

`filter.number`

This selects the index of the wavelet used in the analysis of the time series (i.e. the wavelet basis functions used to model the time series). For Daubechies compactly supported wavelets the filter number is the number of vanishing moments.

`family`

This selects the wavelet family to use in the analysis of the time series (i.e. which wavelet family to use to model the time series). Only use the Daubechies compactly supported wavelets `DaubExPhase` and `DaubLeAsymm`.

switch

This allows one to order the corrected spectrum by scale or decomposition direction. Two options are available

```
switch = "direction"
```

structures the matrix by scale within each decomposition direction. Thus, the ordering goes as follows $(1, V), (2, V), (2, D) \dots$

```
switch = "level"
```

structures the matrix by direction within each scale. Thus the ordering is as follows $(1, V), (1, H), (1, D), (2, V), (2, H), (2, D) \dots$

For further details, see Eckley & Nason (2000).

correct

In Chapter 3 we demonstrated that, as a consequence of the inherent redundancy of the non-decimated wavelet transform, the raw wavelet spectrum is biased. However, an asymptotically unbiased estimator may be obtained by applying the inverse of the inner product matrix of discrete autocorrelation wavelets. This argument permits the user to decide whether or not to correct for this inherent bias.

verbose

Allows certain informative messages to be printed on screen.

smooth

This T/F switch argument allows the user to specify whether or not the resulting local wavelet periodogram should be smoothed to obtain. It is advised that this option be set to T in order that consistent estimates be obtained.

sm.filter.number

Selects the index number of the wavelet that smooths each scale of the wavelet periodogram.

`sm.family`

Selects the wavelet family that smooths each scale of the wavelet periodogram.

`levels`

This specifies the levels which are smoothed when performing the wavelet shrinkage.

`type`

The type of shrinkage: either "hard" or "soft".

`policy`

This dictates the threshold selection method used for smoothing. For LWS estimation `LSuniversal` is recommended for the Chi-squared nature of the periodogram coefficients.

`by.level`

If `T` then the wavelet shrinkage is performed by computing and applying a separate threshold to each level in the transform of each scale. Note that each scale in the LWS is smoothed separately and independently. Each smooth consists of taking the (second-stage) non-decimated wavelet transform and applying a threshold to each level of a wavelet transformed scale.

If `F` then the same threshold is applied to the discrete wavelet transform of a scale. Different thresholds may be computed for different scales but the threshold will be the same for each level arising from the non-decimated transform of a scale.

`value`

This argument supplies the threshold value used when a manual policy is adopted.

`dev`

The method for estimating the variance of the empirical wavelet coefficients for smoothing purposes.

Value

A list with the following components:

S The directionally dependent wavelet spectral estimate of the input data. This is a *large* array, the first dimension refers to a specific scale-direction pair (see Chapter 3 for further details). The next dimension refers to the rows of the spectral image, whilst the third element refers to the columns of the image.

datadim The dimension of the original image.

filter.number This gives the index of the wavelet used in the analysis of the image (i.e. the wavelet basis functions used in the modelling). For Daubechies compactly supported wavelets the filter number is the number of vanishing moments.

family This contains the wavelet family used in the analysis of the image (i.e. the wavelet family used in the modelling).

STRUCTURE

Explains the protocol by which the inner product matrix and **S** are structured. This can only take two values, **direction** and **scale**. In other words, the first dimension of **S** may be structured by scale, for example $(1, H), (1, V), (1, D), (2, H), (2, V) \dots$, or by direction, $(1, V), (2, V), (3, V), \dots, (1, H), (2, H), \dots$

Levels The number of levels in the decomposition.

correct T or F, depending on whether the user corrected for the bias.

smooth T or F, depending on whether the LWP has been smoothed.

date The date at which the analysis was made.

Side effects

None

Details

This function computes an estimate of the directionally dependent wavelet spectrum of an image according to the work of Chapter 3. The function works as follows:

1. The non-decimated wavelet transform of the series is computed.
2. The squared modulus of the non-decimated wavelet transform is computed (this is the raw wavelet periodogram, which is returned as `S` if `smooth="F"`).
3. The squared modulus is smoothed using wavelet shrinkage.
4. The smoothed coefficients are corrected using the inverse of the inner product matrix of the autocorrelation wavelets.

To display the LWS use the `specplot` function on the `S` component (see the examples below).

Examples

```
#
# Apply the cddews estimate function to the lennon image
#
> lennon.cddews <- cddews(lennon, filter.number=1, family="DaubExPhase")
> lennon.cddews
Class 'cddews' : corrected directional dependent wavelet spectrum:
  ~~~~~~ : List with 10 components with names
          S datadim filter.number family STRUCTURE Levels
  invIPmatrix correct smooth date
```

The spectrum of this image was corrected (IP matrix).

`$$` is a large array of data

Created on : Sun Aug 26 18:45:40 BST 2001

`summary(.)`:

Levels: 8

dimension of original image was: 256 x 256 pixels.

Filter was: DaubExPhase N= 1

Structuring is by direction

Date: Sun Aug 26 18:45:40 BST 2001

specplot

Plot the local wavelet periodogram associated with a `cddews` object

Description

This function displays the LWP associated with a `cddews` object, allowing the user to dictate display type.

Usage

```
specplot(cddews, scaling = "by.level", arrangement = c(3., 3.), page =  
T, dataname = "Image", display = "persp", reset = T, title = T)
```

Required Arguments

`cddews`

An object of class `cddews` must be supplied to the function.

Optional Arguments

`scaling`

Two scaling options are available. The default setting is to scale `"by.level"` — an option which is useful if you wish to compare coefficients within a resolution level. The alternative setting is `global`, whereby one scale factor is chosen for all plots. This factor depends on the largest coefficient which is to be included in the suite of plots.

`arrangement`

Allows the user to specify the number of spectral plots which are to appear on any given page.

`page`

An argument which allows the user to request that they be prompted when a new page of plots appears. Two options are available: `T` or `F`.

dataname

A name for the image whose LWP is being displayed. This will appear as part of the title associated with each plot.

display

Two display methods are available. Using the option `display="persp"` displays a 3-dimensional plot of the LWP, using the S-Plus routine `persp`. The option `display="image"` displays the LWP as a collection of images.

reset

If set to T, this restores the plot settings to their default configuration (i.e. `par(mfrow=c(1,1))`). If F, then the current settings will remain in operation.

Title

Setting this option to F allows the suppression of titles in the displayed output.

Value

No value is returned.

Side effects

None.

Haar2MA.diag, Haar2MA.vert, Haar2MA.horiz

Generate 2-D Haar MA process (diagonal/vertical/horizontal direction).

Description

These functions generate an arbitrary number of observations from a Haar MA process of any order with a particular variance. We will focus here on `Haar2MA.diag` — the routine which generates processes having spectral structure solely in the diagonal decomposition direction.

Usage

```
Haar2MA.diag(n, sd = 1, order = 5)
```

Required arguments

`n`

The number of rows *and* columns in realisation that you want to create. Note that `n` does NOT have to be a power of two.

Optional arguments

`sd`

The standard deviation of the innovations.

`order`

The order of the MA process.

Value

A matrix containing a realisation of the specified dimension, order and standard deviation.

Side effects

None.

Details

A two-dimensional Haar MA process is a special kind of moving-average (MA) field. A *diagonal* Haar MA process of order k is a MA field of order $2^k \Leftrightarrow 1$, the coefficients of the process being given by the filter coefficients of the two-dimensional, discrete Haar wavelet at various scales within the diagonal direction. For example: the diagonal Haar MA field of order 1 is an MA process of order 1. The coefficients of this process are given by

$$\begin{bmatrix} 1/2 & \Leftrightarrow 1/2 \\ \Leftrightarrow 1/2 & 1/2 \end{bmatrix}.$$

The Haar MA process of order 2 is an MA process of order 3, its model coefficients being given by

$$\begin{bmatrix} 1/4 & 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 \\ 1/4 & 1/4 & \Leftrightarrow 1/4 & \Leftrightarrow 1/4 \\ \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & 1/4 & 1/4 \\ \Leftrightarrow 1/4 & \Leftrightarrow 1/4 & 1/4 & 1/4 \end{bmatrix}.$$

It is possible to define such processes for other wavelets as well.

Examples

```
#
# Generate a realisation of a diagonal component 2-D MA field
# of order 4.
#
image1 <- HaarMA.diag(n=128, sd=3, order=4)
```

HaarMontage

Generate a 2-D Haar MA process.

Description

This function generates a particular set of four 2-D Haar MA processes. These are subsequently collated to form a montage.

Usage

```
HaarMontage(direction="diagonal")
```

Required arguments

None.

Optional arguments

`direction`

Three directions can be specified: `horizontal`, `vertical` and `diagonal`. The direction chosen dictates the decomposition direction in which the wavelet spectral structure exists.

Value

A vector containing 512*512 observations from four collated 2-D Haar MA processes.

Side effects

None.

Details

This function generates a realisation of a particular kind of non-stationary lattice model, an example of which is displayed in figure 3.3. The returned lattice is the

result of combining four HaarMA processes. One process is of order 1, whilst another is of order 2. The two remaining processes are of order 3 and 4 respectively. Each individual lattice has dimension 128×128 . The standard deviation of the innovations is 1.

Examples

```
#  
# Generate a realisation of the non-stationary combined Haar MA  
# process with structure in the vertical direction.  
#  
> MyHaar2 <- HaarMontage(direction="vertical")  
#  
# Plot it.  
#  
image(MyHaar2)
```

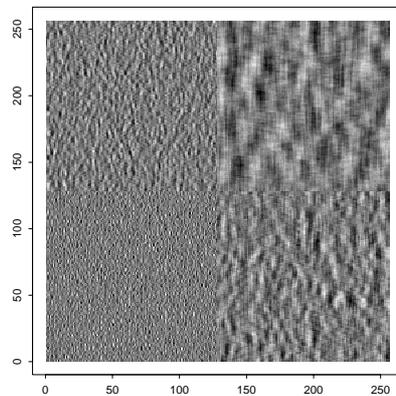


Figure C.2: A realisation of a montage process whose wavelet spectral contributions lie in the vertical direction.

Bibliography

- Abramovich, F., & Benjamini, Y. 1996. Adaptive thresholding of wavelet coefficients. *Computational Statistics and Data Analysis*, **22**, 351–361.
- Abramovich, F., Sapatinas, T., & Silverman, B. W. 1998. Wavelet thresholding via a Bayesian approach. *Journal of the Royal Statistical Society (Series B)*, **60**, 725–749.
- Abramovich, F., Bailey, T. C., & Sapatinas, T. 2000. Wavelet analysis and its statistical application. *Journal of the Royal Statistical Society (Series D)*, **49**, 1–29.
- Antoniadis, A. 1997. Wavelet in statistics: a review (with discussion). *Journal of the Italian Statistical Society*, **2**, 97–138.
- Antoniadis, A., Grégoire, G., & Nason, G. P. 1999. Density and hazard rate estimation for right-censored data using wavelet methods. *Journal of the Royal Statistical Society (Series B)*, **61**, 63–84.
- Bovic, A. C., Clark, M., & Geisler, W. S. 1990. Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 55–73.
- Brockwell, P. J., & Davis, R. A. 1991. *Time Series*. Springer-Verlag, Berlin.
- Brodatz, P. 1966. *Textures: A photographic album for artists and designers*. Dover Publications, New York.
- Candès, E. J. 1999. Harmonic analysis of neural networks. *Applied and Computational Harmonic Analysis*, **6**, 197–218.

- Candès, E. J., & Donoho, D. L. 1999a. Curvelets - a surprisingly effective nonadaptive representation for objects with edges. *Pages 125–150 of: Schumaker, L. L. et al. (ed), Curves and Surfaces.* Vanderbilt University Press, Nashville, TN.
- Candès, E. J., & Donoho, D. L. 1999b. Ridgelets: they key to higher-dimensional intermittency? *Philosophical Transactions of the Royal Society of London (Series A)*, **357**, 2495–2509.
- Champeney, D. C. 1973. *Fourier transforms and their physical applications.* Academic Press, London.
- Chang, T., & Kuo, C.-C. J. 1993. Texture analysis and classification with tree-structured wavelet transform. *IEEE Transactions on Image Processing*, **2**, 429–441.
- Chiann, C., & Morettin, P. A. 1999. A wavelet analysis for time series. *Journal of Nonparametric Statistics*, **10**, 1–46.
- Choi, H., Romberg, J. K., Baraniuk, R. G., & Kingsbury, N. G. 2000. Hidden Markov tree modeling of complex wavelet transforms. *Pages 133–136 of: Proceedings of 2000 IEEE International Conference on Acoustics, Speech and Signal Processing Processing, Istanbul, Turkey*, vol. 1. IEEE, Piscataway, NJ.
- Chui, C. K. 1992. *An introduction to wavelets.* Academic Press, London.
- Coggins, J. M. 1982. *A framework for texture analysis based on spatial filtering.* Ph.D. thesis, Michigan State University.
- Coifman, R. R., & Donoho, D. L. 1995. Translation-invariant de-noising. *Pages 125–150 of: Antoniadis, A., & Oppenheim, G. (eds), Wavelets and Statistics.* Lecture Notes in Statistics, no. 103. Springer-Verlag, New York.
- Coifman, R. R., & Wickerhauser, M. V. 1992. Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, **38**, 713–719.
- Coifman, R. R., Meyer, Y., & Wickerhauser, M. V. 1992. Wavelet analysis and signal processing. *Pages 153–178 of: Ruskai, B. et al. (ed), Wavelets and their applications.* Jones and Bartlett, Boston.

- Connors, R. W., & Harlow, C. A. 1980. A theoretical comparison of texture algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 204–222.
- Cressie, N. A. C. 1991. *Statistics for spatial data*. Wiley, New York.
- Cross, G. C., & Jain, A. K. 1983. Markov random field texture models. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **6**, 661–674.
- Crouse, M. S., Nowak, R. D., & Baraniuk, R. G. 1998. Wavelet-based statistical signal processing using hidden Markov models. *IEEE Transactions on Signal Processing*, **46**, 886–902.
- Dahlhaus, R. 1997. Fitting time series models to nonstationary processes. *The Annals of Statistics*, **25**, 1–37.
- Daubechies, I. 1988. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, **XLI**, 909–966.
- Daubechies, I. 1992. *Ten Lectures on Wavelets*. SIAM, Philadelphia.
- Daubechies, I., Guskov, I., Schröder, P., & Sweldens, W. 1999. Wavelets on irregular point sets. *Philosophical Transactions of the Royal Society of London (Series A)*, **357**, 2397–2413.
- Daugman, J. G. 1990. An information-theoretic view of analog representation in striate cortex. *Pages 403–423 of: Schwartz, E. L. (ed), Computational Neuroscience*. MIT Press, Cambridge, Mass.
- Davies, L. S., & Mitchie, A. 1980. Edge detection in textures. *Computer graphics and image processing*, **12**, 25–39.
- de Rivaz, P. F. C., & Kingsbury, N. G. 1999. Complex wavelet features for fast texture image retrieval. *Pages 109–113 of: Proceedings of the IEEE International Conference on Image Processing, Kobe Japan*, vol. 1. IEEE, Piscataway, NJ.
- Deguchi, K., & Morishita, I. 1978. Texture characterization and texture-based partitioning using two-dimensional linear estimation. *IEEE Transactions on Computers*, **27**, 739–745.

- Do, M. N., & Vetterli, M. 2000. Orthonormal finite ridgelet transform for image compression. *Pages 367–370 of: Proceedings of 7th IEEE International Conference on Image Processing, Vancouver, BC, Canada*, vol. 2. IEEE, Piscataway, NJ.
- Do, M. N., & Vetterli, M. 2001. Discrete ridgelet transforms for image representation. *Submitted for publication*.
- Donoho, D. L., & Johnstone, I. M. 1994. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**, 425–455.
- Donoho, D. L., & Johnstone, I. M. 1995. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, **20**, 1200–1224.
- Donoho, D. L., Johnstone, I. M., Kerkyacharian, G., & Picard, D. 1995. Wavelet shrinkage: asymptopia? (with discussion). *Journal of the Royal Statistical Society (Series B)*, **57**, 301–369.
- Donoho, D. L., Johnstone, I. M., Kerkyacharian, G., & Picard, D. 1996. Density estimation by wavelet thresholding. *The Annals of Statistics*, **24**, 508–539.
- Dunn, D., & Higgins, W. E. 1995. Optimal Gabor filters for texture segmentation. *IEEE Transactions on Image Processing*, **4**, 947–964.
- Eckley, I. A., & Nason, G. P. 2000. *Efficient computation of the inner-product matrix of discrete autocorrelation wavelets*. Technical Report 00:12. School of Mathematics, University of Bristol.
- Fan, J. 1996. Test of significance based on wavelet thresholding and Neyman's truncations. *Journal of the American Statistical Association*, **91**, 674–688.
- Field, D. J. 1999. Wavelets, vision and the statistics of natural scenes. *Philosophical Transactions of the Royal Society of London (Series A)*, **357**, 2527–2542.
- Galloway, M. M. 1975. Texture classification using gray level run length. *Computer Graphics and Image Processing*, **4**, 172–179.
- Gao, H.-Y. 1997. Choice of thresholds for wavelet shrinkage estimation of the spectrum. *Journal of Time Series Analysis*, **18**, 231–251.

- Goel, P., & Vidakovic, B. 1995. *Wavelet transformations as diversity enhancers*. Discussion paper 95-04, ISDS. Duke University.
- Graps, A. 1995. An introduction to wavelets. *IEEE Computational Sciences and Engineering*, **2**, 50–61.
- Haar, A. 1910. Zur Theorie der orthogonalen Funktionen-Systeme. *Mathematische Annalen*, **69**, 331–371.
- Haas, T. 1990. Lognormal and moving window methods of estimating acid decomposition. *Journal of the American Statistical Association*, **85**, 950–963.
- Haining, R. P. 1978. The moving average model for spatial interaction. *Transactions of the Institute of British Geographers*, **3**, 202–225.
- Hall, P., & Patil, P. 1995. Formulae for mean integrated squared error of nonlinear wavelet-based density estimation. *The Annals of Statistics*, **23**, 905–928.
- Haralick, R. M. 1979. Statistical and structural approaches to texture. *Proceedings of the IEEE*, **67**, 786–804.
- Haralick, R. M. 1986. Statistical image texture analysis. *Pages 247–279 of: Young, T., & Fu, K. S. (eds), The handbook of pattern recognition and image processing*. Academic Press, Orlando, Fl.
- Haralick, R. M., Shanmugam, K., & Dinstein, I. 1973. Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-3**, 610–621.
- Herrick, D. R. M. 2000. *Wavelet methods for curve and surface estimation*. Ph.D. thesis, University of Bristol.
- Herrick, D. R. M., Nason, G. P., & Silverman, B. W. 2002. Some new methods for wavelet density estimation. *Sankhya (to appear)*.
- Higdon, D., Swall, J., & Kern, J. 1999. Non-stationary spatial modeling. *Pages 761–768 of: et al. J. M. Bernardo (ed), Bayesian Statistics 6*. Oxford University Press, Oxford.

- Hubbard, B. Burke. 1998. *The world according to wavelets*. First edn. A K Peters, Wellesley, Massachusetts.
- Hunt, K., & Nason, G. P. 2002. Wind speed modelling and short-term prediction using wavelets. *Wind Engineering (to appear)*.
- Isserlis, L. 1918. On a formula for the product moment coefficient of any order of a normal frequency distribution in any number of variables. *Biometrika*, **12**, 134–139.
- Iversen, H., & Lønnestad, T. 1992. An evaluation of stochastic models for analysis and synthesis of gray-scale texture. *Pattern Recognition Letters*, **15**, 575–585.
- Jain, A. K., & Farrokhnia, F. 1991. Unsupervised texture segmentation using Gabor filters. *Pattern Recognition*, **24**, 1167–1186.
- Jain, A. K., Farrokhnia, F., & Alman, D. H. 1990. Texture analysis of automotive finishes. *Pages 1–16 of: Proceedings of SME Machine Vision Applications Conference, Detroit*.
- Jawerth, B., & Sweldens, W. 1994. An overview of wavelet based multiresolution analysis. *SIAM Review*, **36**, 377–412.
- Johnstone, I. M., & Silverman, B. W. 1997. Wavelet threshold estimators for data with correlated noise. *Journal of the Royal Statistical Society (Series B)*, **59**, 319–351.
- Julesz, B. 1981. Textons, the elements of texture perception and their interactions. *Nature*, **290**, 91–97.
- Kaizer, H. 1955. *A quantification of textures on aerial photographs*. Technical Note 121. Boston University Research Laboratory, Boston, MA.
- Kingsbury, N. G. 1999. Image processing with complex wavelets. *Philosophical Transactions of the Royal Society of London (Series A)*, **357**, 2543–2560.
- Kovac, A. 1998. *Wavelet thresholding for unequally spaced data*. Ph.D. thesis, University of Bristol.
- Laine, A., & Fan, J. 1993. Texture classification by wavelet packet signatures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**, 1186–1191.

- Laine, A., & Fan, J. 1996. Frame representations for texture segmentation. *IEEE Transactions on Image Processing*, **5**, 771–780.
- Laws, K. I. 1979. Texture energy measures. *Pages 47–51 of: DARPA Image Understanding Workshop, Los Angeles, CA.*
- Le, N. D., & Zidek, J. V. 1992. Interpolation with uncertain spatial covariance. *Journal of Multivariate Analysis*, **43**, 351–374.
- Le, N. D., Sun, W., & Zidek, J. V. 1997. Bayesian multivariate spatial interpolation with data missing by design. *Journal of the Royal Statistical Society (Series B)*, **59**, 501–510.
- Lee, J. H., & Philpot, W. D. 1991. Spectral texture pattern matching: a classifier for digital imagery. *IEEE Transactions on Geoscience and Remote Sensing*, **29**, 545–554.
- Loader, C., & Switzer, P. 1992. Spatial covariance estimation for monitoring data. *Pages 52–70 of: Walden, A. T., & Guttorp, P. (eds), Statistics in the Environmental and Earth Sciences.* Arnold, London.
- Lundervold, A. 1992. *Ultrasonic tissue characterization — a pattern recognition approach.* Technical Report. Norwegian Computing Center, OSLO.
- Mallat, S. G. 1989a. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *Transactions of the American Mathematical Society*, **315**, 69–87.
- Mallat, S. G. 1989b. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, 674–693.
- Mallat, S. G. 1999. *A wavelet tour of signal processing.* Second edn. Academic Press, London.
- Mardia, K. V., Kent, J. T., & Bibby, J. M. 1979. *Multivariate analysis.* Academic Press, London.
- Meyer, Y. 1986. Principe d’incertitude, bases Hilbertiennes et algèbres d’opérateurs. *Séminaire Bourbaki*, **662**.

- Meyer, Y. 1992. *Wavelets and Operators*. Cambridge University Press, Cambridge.
- Moore, M. 1988. Spatial linear processes. *Communications in Statistics: Stochastic Models*, **4**, 45–75.
- Morettin, P. A. 1997. Wavelets in Statistics. *Resenhas*, **3**, 211–272.
- Nason, G. P. 1996. Wavelet shrinkage using cross-validation. *Journal of the Royal Statistical Society (Series B)*, **58**, 463–479.
- Nason, G. P. 1998. *WaveThresh3 Software*. Department of Mathematics, University of Bristol, UK.
- Nason, G. P., & Sapatinas, F. 2001. Wavelet packet transfer function modelling of nonstationary time series. *Statistics and Computing*, (to appear).
- Nason, G. P., & Silverman, B. W. 1994. The discrete wavelet transform in S. *Journal of Computational and Graphical Statistics*, **3**, 163–191.
- Nason, G. P., & Silverman, B. W. 1995. The Stationary Wavelet Transform and some Statistical Applications. *Pages 281–300 of: Antoniadis, A., & Oppenheim, G. (eds), Wavelets and Statistics*. Lecture Notes in Statistics, no. 103. Springer-Verlag, New York.
- Nason, G. P., & von Sachs, R. 1999. Wavelets in time series analysis. *Philosophical Transactions of the Royal Society of London (Series A)*, **357**, 2511–2526.
- Nason, G. P., von Sachs, R., & Kroisandt, G. 2000. Wavelet processes and adaptive estimation of the evolutionary wavelet spectrum. *Journal of the Royal Statistical Society (Series B)*, **62**, 271–292.
- Nason, G. P., Sapatinas, F., & Sawczenko, A. 2001. Wavelet packet modelling of infant sleep state using heart rate data. *Sankhya*, (to appear).
- Neumann, M. H., & von Sachs, R. 1995. Wavelet thresholding: beyond the Gaussian i.i.d. *Pages 301–329 of: Antoniadis, A., & Oppenheim, G. (eds), Wavelets and Statistics*. Lecture Notes in Statistics, no. 103. Springer-Verlag, New York.

- Neumann, M. H., & von Sachs, R. 1997. Wavelet thresholding in anisotropic function classes and application to adaptive estimation of evolutionary spectra. *The Annals of Statistics*, **25**, 38–76.
- Ogden, R. T., & Parzen, E. 1996a. Change-point approach to data analytic wavelet thresholding. *Statistics and Computing*, **6**, 93–99.
- Ogden, R. T., & Parzen, E. 1996b. Data dependent wavelet thresholding in nonparametric regression with change-point applications. *Computational Statistics and Data Analysis*, **22**, 53–70.
- Parzen, E. 1961. An approach to time series analysis. *Annals of Mathematical Statistics*, **32**, 951–989.
- Penev, S., & Dechevsky, L. 1997. On non-negative wavelet based density estimators. *Journal of Nonparametric Statistics*, **7**, 365–394.
- Pentland, A. P. 1984. Fractal-based description of natural scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 661–674.
- Percival, D. B., & Walden, A. T. 2000. *Wavelet methods for time series analysis*. Cambridge University Press, Cambridge.
- Portilla, J., & Simoncelli, E. P. 2000. A parametric texture model based on joint statistics of complex wavelet coefficients. *International Journal of Computer Vision*, **40**, 49–70.
- Priestley, M. B. 1965. Evolutionary spectra and non-stationary processes. *Journal of the Royal Statistical Society (Series B)*, **27**, 204–237.
- Priestley, M. B. 1981. *Spectral Analysis and Time Series*. Academic Press, London.
- Qian, W., & Titterton, D. M. 1991. Multidimensional Markov chain models for image textures. *Journal of the Royal Statistical Society (Series B)*, **53**, 661–674.
- Randen, T., & Husøy, J. H. 1999. Filtering for texture classification: a comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**, 291–310.

- Reed, T. R., & du Buf, J. M. H. 1992. A review of recent texture segmentation and feature extraction techniques. *CVGIP: Image Understanding*, **57**, 359–372.
- Reed, T. R., & Wechsler, H. 1990. Segmentation of textured images and Gestalt organization using spatial/spatial-frequency representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 1–12.
- Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge.
- Romberg, J. K., Choi, H., & Baraniuk, R. G. 2000. Bayesian tree structured image modeling using wavelet-domain hidden Markov models. *Submitted for publication*.
- Saito, N., & Beylkin, G. 1993. Multiresolution representation using the autocorrelation functions of compactly supported wavelets. *IEEE Transactions on Signal Processing*, **41**, 3584–3590.
- Saito, N., & Coifman, R. R. 1995. Local discriminant bases and their applications. *Journal of Mathematical Imaging and Vision*, **5**, 337–358.
- Sampson, P. D., & Guttorp, P. 1992. Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, **87**, 108–119.
- Scheunders, P., Livens, S., Van de Wouwer, G., Vautrot, P., & Van Dyck, D. 1998. Wavelet-based texture analysis. *International Journal on Computer Science and Information Management*, **1**, 22–34.
- Serra, J., & Verchery, G. 1973. Mathematical morphology applied to fibre composite materials. *Fibre Science and Technology*, **6**, 141–158.
- Simaan, M. 1990. Knowledge-guided segmentation of texture images. *Pages 539–542 of: Proceedings of the IEEE International Conference on Systems Engineering, Pittsburgh, PA*. IEEE, New York.
- Simoncelli, E. P., & Freeman, W. T. 1995. The steerable pyramid: a flexible architecture for multi-scale derivative computation. *Pages 444–447 of: Proceedings*

- of the IEEE International Conference on Image Processing, Washington, DC*, vol. 3. IEEE Computer Society, Los Alamitos, CA.
- Sonka, M., Hlavac, V., & Boyle, R. 1999. *Image processing, analysis and machine vision (Second Edition)*. PWS Publishing, New York.
- Starck, J. L., Candès, E. J., & Donoho, D. L. 2000. The curvelet transform for image denoising. *Submitted for publication*.
- Stein, C. 1981. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, **9**, 1135–1151.
- Stuart, A., Ord, J. K., & Arnold, S. 1999. *Kendall's advanced theory of statistics (vol. 2A)*. Sixth edn. Arnold, London.
- Sutton, R., & Hall, E. 1972. Texture measures for automatic classification of pulmonary diseases. *IEEE Transactions on Computers*, **C-21**, 667–678.
- Sweldens, W. 1996. The lifting scheme: a custom-design construction of biorthogonal wavelets. *Applied and Computational Harmonic Analysis*, **3**, 186–200.
- Sweldens, W. 1997. The lifting scheme: a construction of second generation wavelets. *SIAM Journal of Mathematical Analysis*, **29**, 511–546.
- Tomita, F., & Tsuji, S. 1990. *Computer analysis of visual textures*. Kluwer Academic Publishers, Dodrecht.
- Tuceryan, M., & Jain, A. K. 1999. Texture Analysis. *Pages 207–248 of: Chen, C. H., Pau, L. F., & P., Wang P. S. (eds), Handbook of pattern recognition and computer vision (Second Edition)*. World Scientific Publishing, Singapore.
- Turner, M. R. 1986. Texture discrimination by Gabor functions. *Biological Cybernetics*, **55**, 71–82.
- Unser, M. 1986. Local linear transforms for texture measurements. *Signal Processing*, **11**, 61–79.
- Unser, M. 1995. Texture classification and segmentation using wavelet frames. *IEEE Transactions on Image Processing*, **4**, 1549–1560.

- Unser, M., & Eden, M. 1989. Multiresolution feature extraction and selection for texture segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, 717–728.
- Van de Wouwer, G. 1998. *Wavelets for multiscale texture analysis*. Ph.D. thesis, University of Antwerp.
- Van de Wouwer, G., Scheunders, P., & Van Dyck, P. 1999a. Statistical texture characterization from discrete wavelet representation. *IEEE Transactions on Image Processing*, **8**, 592–598.
- Van de Wouwer, G., Scheunders, P., Livens, P., & Van Dyck, P. 1999b. Wavelet correlation signatures for color texture characterization. *Pattern Recognition*, **32**, 443–451.
- Vannucci, M., & Vidakovic, B. 1997. Preventing the Dirac disaster: wavelet based density estimation. *Journal of the Italian Statistical Society*, **6**, 145–159.
- Venkatachalam, V., Choi, H., & Baraniuk, R. G. 2000. Multiscale SAR image segmentation using wavelet-domain hidden Markov tree models. *Pages 110–120 of: Proceedings of the SPIE 14th International Symposium on Aerospace/Defense Sensing, Simulation, and Controls, Algorithms for Synthetic Aperture Radar Imagery VII, Orlando, FL.*, vol. 4053. SPIE, Bellingham, WA.
- Vidakovic, B. 1999. *Statistical modelling by wavelets*. Wiley, New York.
- von Sachs, R., & MacGibbon, B. 2000. Non-parametric curve estimation by wavelet thresholding with locally stationary errors. *Scandinavian Journal of Statistics*, **27**, 475–499.
- von Sachs, R., & Schneider, K. 1996. Wavelet smoothing of evolutionary spectra by nonlinear thresholding. *Applied and Computational Harmonic Analysis*, **3**, 268–282.
- von Sachs, R., Nason, G. P., & Kroisandt, G. 1997. *Adaptive estimation of the evolutionary wavelet spectrum*. Technical Report 516. Department of Statistics, Stanford University.

-
- Walden, A. T., Percival, D. B., & McCoy, E. J. 1998. Spectrum estimation by wavelet thresholding of multitaper estimators. *IEEE Transactions on Signal Processing*, **46**, 3153–3165.
- Wechsler, H. 1980. Texture analysis — a survey. *Signal Processing*, **2**, 271–282.
- Weszka, J. S., Dyer, C., & Rosenfeld, A. 1976. A comparative study of texture measures for terrain classification. *IEEE Transactions on Systems, Man and Cybernetics*, **6**, 269–285.