Image and Video Coding Using a Wavelet Decomposition

H. J. Barnard

Image and Video Coding Using a Wavelet Decomposition

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus Prof. ir. K. F. Wakker,
in het openbaar te verdedigen ten overstaan van een commissie,
door het College van Dekanen aangewezen,
op dinsdag 10 mei 1994 te 16.00 uur

door

Herman Johannes BARNARD

wiskundig ingenieur,
geboren te Almkerk.

Dit proefschrift is goedgekeurd door de promotor: Prof. dr. ir. J. Biemond
en de toegevoegd promotor: Dr. ir. J. H. Weber.

Samenstelling promotiecommissie:

Rector Magnificus	(voorzitter)
Prof. dr. ir. J. Biemond	(TU Delft, Fac. Elektrotechniek)
Dr. ir. J. H. Weber	(TU Delft, Fac. Elektrotechniek)
Prof. dr. ir. E. Backer	(TU Delft, Fac. Elektrotechniek)
Prof. dr. ir. F. C. A. Groen	(Univ. van Amsterdam, Fac. Wiskunde en Informatica)
Prof. dr. ir. A. W. Heemink	(TU Delft, Fac. Technische Wiskunde en Informatica)
Prof. dr. B. Sankur	(Bogaziçi Universitesi, Istanbul, Turkije)
Prof. dr. I. T. Young	(TU Delft, Fac. Technische Natuurkunde)

CIP - GEGEVENS KONINKLIJKE BIBLIOTHEEK, DEN HAAG

Barnard, H.J.

Image and video coding using a wavelet decomposition /
H.J. Barnard. - Delft : Technische Universiteit Delft,
Faculteit der Elektrotechniek, Vakgroep Informatietheorie. - Ill.
Thesis Technische Universiteit Delft. - With ref. - With summary in Dutch
ISBN 90-5326-016-1
Subject headings: image coding / video coding / wavelets.

Copyright © 1994 by H. J. Barnard

All rights reserved. No part of this thesis may be reproduced or transmitted in any
form or by any means without written permission from the author.

Table of Contents

Summary	ix
Notations and Abbreviations	xiii
1 Introduction	1
1.1 Image and video coding techniques	2
1.2 Scope and outline of this thesis	4
2 Wavelet Theory	7
2.1 The continuous-time wavelet transform and series	8
2.2 Multiresolution representation of functions	11
2.3 The discrete-time wavelet series	19
2.4 The discrete wavelet transform	22
2.5 Regularity	28
2.6 Short history of the wavelet theory	31
2.7 Conclusions	32
3 Wavelet Decomposition of Arbitrary Length Signals	33
3.1 Introduction	34
3.2 Proper design and implementation of filters for perfect reconstruction	36
3.3 Proper signal extensions for perfect reconstruction	38
3.3.1 Periodic extension	38
3.3.2 Symmetric extension	40
3.4 Efficient signal extension for decomposition of arbitrary length signals	42
3.5 Efficient signal extension for more dimensional filters and multi-channel analysis/synthesis schemes	44

3.6	Symmetric-periodic signal extension	45
3.7	Conclusions	46
4	Image and Video Coding Schemes	47
4.1	The image coding scheme	47
4.2	Some video coding schemes	53
4.3	Conclusions	57
5	Comparison of Wavelet Filters and Subband Coding Filters	59
5.1	Criteria for filter design	60
5.2	Subband coding filter design: Johnston filters	63
5.3	Wavelet filter design: Daubechies filters and biorthogonal filter pairs	70
5.3.1	Design of the Daubechies filters	70
5.3.2	Design of a biorthogonal filter pair	71
5.4	Comparison of SBT and WT filter characteristics	76
5.5	Comparison of various filters for image and video coding	77
5.5.1	The subband coding gain for still images	79
5.5.2	The objective and subjective coding performance for still images	82
5.5.3	The objective and subjective coding performance for video sequences	92
5.6	Conclusions	96
6	The Two-Dimensional Discrete Wavelet Transform	97
6.1	The dyadic two-dimensional discrete wavelet transform	98
6.2	The general two-dimensional discrete wavelet transform	99
6.3	The quincunx two-dimensional discrete wavelet transform	102
6.4	Comparison of the dyadic and quincunx wavelet transform	110
6.5	Conclusions	112
7	A Region-Based Discrete Wavelet Transform	115
7.1	Contour-texture coding	116
7.2	Principle and implementation of the RBDWT	117
7.2.1	Decomposition of the segmentation mask	119
7.2.2	The filtering and downsampling of the regions	121
7.3	The coding scheme	126
7.4	Properties, advantages, and disadvantages of the RBDWT	128
7.5	Experimental comparison of the RBDWT and the standard DWT	130
7.5.1	Full resolution coding of images	131
7.5.2	Multiresolution coding of images	139
7.6	Application to video coding	141
7.7	Conclusions and further research	143

Bibliography	145
Appendix A Overview of Wavelet Transforms and Series	155
A.1 Overview of the formulas for wavelet and Fourier transforms and series	155
A.2 The two-dimensional continuous-time wavelet transform	159
Appendix B Linear Phase and Biorthogonality	161
B.1 The importance of linear phase for image coding	161
B.2 Biorthogonality	163
Appendix C Derivations of Formulas	167
C.1 Some wavelet theory equations (Section 2.2)	167
C.2 The input-output relation of a two-band splitting scheme (Section 3.2)	169
C.3 The optimal bit rate (Section 5.5.1)	172
C.4 The input-output relation of a quincunx two-band splitting scheme (Section 6.3)	173
Appendix D On the Entropy of a 4-Connected Chain Code	175
Appendix E Filters	179
E.1 The Johnston filters	180
E.2 The Daubechies filters	185
E.3 The biorthogonal wavelet filters of Cohen, Daubechies and Feauveau	188
E.4 The filter pair with lengths 9 and 7 of Kronander	192
E.5 The Smith and Barnwell filter of length 16	194
Appendix F Test Images and Video Sequences	197
Samenvatting (Summary in Dutch)	203
Acknowledgements	207
Curriculum Vitae	209

Summary

In the last decade there has been an enormous increase in the exchange and storage of information. Together with this growth, there has also been a change in the type of information: visual information in the form of digital images and video sequences has become increasingly important. This type of information especially gives rise to high transmission and storage costs. Therefore, the development of reliable and fast compression techniques for several quality levels has become an important research topic. Many algorithms have been proposed in literature and some of them have already been standardized. However, standardization is a time-consuming process and development goes on. There is a need for additional standards and for special-purpose algorithms. This thesis discusses one of the relatively recently developed compression techniques and its application to digital image and video compression. This compression technique uses multiple resolutions of the signal and is based on the wavelet theory.

The wavelet theory was developed because of the inability of the Fourier theory to represent non-stationary signals. Despite the fact that the wavelet theory is in one sense a follow-up of the Fourier theory, this fact has not been reflected in the notation and naming of transforms in many of the texts on wavelets that have appeared. One of the contributions of this thesis is that it aims to present a consistent notation and naming in complete analogy to the Fourier theory. We have to make a choice between the mathematical and the signal-processing notation conventions of Fourier theory. Since the application area we have in mind is the coding of digital images and video

sequences, the latter notation convention is the most logical choice. We base our notation and naming on those on Fourier theory appearing in [Oppe83].

The wavelet theory is explained in many texts by a very mathematical approach. However, for the application to image and video coding large theoretical parts are not really necessary for proper understanding and correct implementation. Therefore, we derive in this thesis the wavelet theory for signal processing applications by using as little mathematics as possible. Nevertheless, in some cases a rather mathematical description is inevitable.

We concentrate on that part of the wavelet theory which is used in digital image processing, that is, the discrete wavelet transform (DWT). The implementation of the DWT appears to have a close resemblance to the two-channel subband coding scheme, which has been used for signal processing since the seventies. In fact, the main differences in the implementation are in the characteristics of the filters. The filters in subband coding schemes are designed to have good frequency responses, like stopband attenuation and transition bandwidth. Examples of these types of filters are the widely used Johnston filters, for some of which we propose a modification that improves the performance. In other words, subband coding filter design concentrates on approximating the ideal block shape of the filter frequency response. The filters in wavelet transform schemes, however, are mainly designed to have a certain degree of regularity. The Daubechies filters are the most famous examples of this filter type.

The concept of regularity was not considered in the subband coding theory. It can be roughly understood as the smoothness of the limit functions of the iterated filters in a splitting tree. This is closely related to flatness of the filter frequency response at half the sampling frequency. Another contribution of wavelet theory to signal processing is that these limit functions form a link between digital filtering and the theory of continuous function spaces.

One of the first research groups to use wavelet filters in application to image compression stated that their regularity is important in giving smooth quantization error responses in the reconstructed image [Anto92]. This is not affirmed by our experiments, because no relation between regularity and performance has shown up. However, another assertion is supported by our experiments: the most regular filter of a biorthogonal filter pair should be in the synthesis filter bank. Nevertheless, as ascertained experimentally, other filter characteristics are determinant in deciding about the best location of the filters when the regularity of both filters is above ca. 2. In fact, a regularity higher than 2 seems not to be needed in image processing applications.

In the overall comparison of various filters, designed in accordance with either the subband coding theory or the wavelet theory, in general no large differences in coding performance appear. The criteria we use in the comparison of filters are the energy compaction capability (or, subband coding gain), the objective coding performance (by the signal-to-noise ratio), and the subjective coding performance. Six test images and two test video sequences were used in the experiments. The biorthogonal (linear phase) wavelet filters perform best, followed by almost all linear phase subband coding filters, and the non-linear phase wavelet filters.

The theory of wavelets can be extended to the two-dimensional (2-D) case. For application to discrete signals we derive a general 2-D DWT, which has the usual separable implementation as a special, degenerated case. This has been used in the experiments mentioned above and is here referred to as the dyadic 2-D DWT. Another variant of the general 2-D DWT is the quincunx 2-D DWT. Just like the one-dimensional DWT, it divides a signal into two bands, one with the low frequencies and the other with the high frequencies. Since the quincunx 2-D DWT is not supporting any direction, it is expected to perform better than the dyadic 2-D DWT for some classes of images, like aerial photographs and images with many diagonal structures.

To compare the coding performances of the quincunx and dyadic 2-D DWT we use quincunx filters that are related to the filters of the dyadic transform by the McClellan transform. In the experiments, the quincunx 2-D DWT showed a worse performance than the dyadic 2-D DWT. If the quincunx 2-D DWT is investigated further, the main topic should be the filter design.

In addition to transform coding methods like the DWT, several region-based coding techniques are described in literature. The idea of region-based coding, or contour-texture coding, is to describe the image by means of contours and textures, which are coded separately. Ideally, the regions should correspond to objects in the scene. This type of coding method has been extensively discussed in [Kunt85]. In region-based coding, the edges in an image are preserved better than they are in transform coding and thus the objects can be recognized more easily.

The most common method for coding the texture is to fit a polynomial to the grey-values in a region. Up to now, the DWT as described in literature could not be used to code the texture. The reason for this is that a d -level decomposition requires the regions to consist of square blocks sized $2^d \times 2^d$. In general, such regions will not correspond to the objects in a scene. However, a new, efficient signal extension

method for filtering is proposed in this thesis and it enables a wavelet decomposition of arbitrary length signals without introducing additional transform coefficients. The new extension is visually demonstrated with diagrams, which show that the perfect reconstruction capability is preserved. When applied to two-dimensional signals, this new method enables us to perform a DWT on arbitrarily shaped regions. We call this the region-based discrete wavelet transform (RBDWT). From another point of view, the RBDWT can be considered a special kind of DWT, in which the subbands are divided into regions.

Thus, a new contour-texture coding scheme can be designed in which the texture part is coded by the RBDWT. The scheme has the advantage that it is able to distribute the available bit rate over subbands as well as regions. The segmentation of the image should consist of relatively large regions in order to make use of the division into frequency bands. In contrast, texture coding with polynomials requires small regions.

As the main interest is in the implementation aspects of the RBDWT, we do not optimize the segmentation process, but this is a recommended subject for further research. With an optimized segmentation, a comparison of the RBDWT to the polynomial coding of texture should be performed.

Like most region-based coding methods, the RBDWT coding scheme is not very suitable for hierarchical coding. The reason is that the downsampling of the segmentation information introduces a blocking effect. For the application of the RBDWT to video coding, the most promising option seems to be the coding of those areas for which motion compensation has failed.

In the experimental evaluation of the new image coding scheme, we use a fixed segmentation for the two test images. If the subjective and objective coding performance of the region-based discrete wavelet transform and the standard discrete wavelet transform are compared, the standard version has a higher signal-to-noise ratio than the new method. However, the subjective coding performance of the region-based discrete wavelet transform has proved to be quite competitive. In particular, the edges are sharper and the ringing artefacts are smaller. Therefore, we conclude that the RBDWT is a promising method for coding texture in a region-based coding scheme.

Notations and Abbreviations

Notations

In this thesis the set of real numbers is denoted by R , and the set of integers by Z . The imaginary unit is denoted by i . The frequency domain variable is represented by ω .

Vectors and matrices are printed in bold face using lower case and upper case, respectively. The matrix I is the identity matrix, which size will be clear from the context. The absolute value of the determinant of a matrix M is denoted by $|M|$. A bold face number is a vector with all elements equal to that number.

A sequence s with argument n is denoted by s_n , where the argument n is assumed to range over Z or an interval of Z . Analogously, a function f with argument x is denoted by $f(x)$ and the argument x is assumed to range over R or an interval of R . The functions $\phi(x)$ and $\psi(x)$ represent a scaling function and a wavelet, respectively.

When the ranges of an integral or a summation are not given these are assumed to be R and Z , respectively.

$L^2(R)$ is the space of square integrable functions with arguments from R . The inner product is often written with the notation $\langle \cdot, \cdot \rangle$.

The notation $\delta_{m,n}$ is the Dirac delta function, which takes the value 1 if m equals n , otherwise it equals 0.

The Fourier Transform of a function or sequence is denoted by the same letter capitalized. In this thesis we use the following definition of the Fourier transform and its inverse for functions:

$$F(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx, \quad f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{i\omega x} d\omega.$$

For sequences, we define the Fourier transform and its inverse by

$$H(\omega) = \sum_{n=-\infty}^{\infty} h_n e^{-i\omega n}, \quad h_n = \frac{1}{2\pi} \int_0^{2\pi} H(\omega)e^{i\omega n} d\omega.$$

Abbreviations

The following abbreviations are used in this thesis:

CTFS	Continuous Time Fourier Series
CTFT	Continuous Time Fourier Transform
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DPCM	Differential Pulse Code Modulation
DTFT	Discrete Time Fourier Transform
DWT	Discrete Wavelet Transform
FFT	Fast Fourier Transform
HDTV	High Definition Television
HVS	Human Visual System
MPEG	hybrid video coding scheme of the Moving Pictures Experts Group,
MSE	Mean Square Error
PCM	Pulse Code Modulation
PR	Perfect Reconstruction
PSNR	Peak Signal-to-Noise Ratio
QMF	Quadrature Mirror Filter
RBDWT	Region-Based Discrete Wavelet Transform
SBC	Subband Coding
SBD	Subband Decomposition
SBT	Subband Theory
STFT	Short-Time Fourier Transform
WT	Wavelet Theory
bpp	bits per pixel
pdf	probability density function

Chapter 1

Introduction

In the last decade there has been an enormous increase in all kinds of communications. Computers all over the world are connected via modems and networks, creating an explosion of information exchange and information storage. Electronic mail, facsimile and video conferencing have appeared and recently the first generation of videotelephones has entered the consumer market. In the near future, new applications like tele-shopping and tele-education will be introduced. Data-bases of images have appeared in several application areas, for instance, for medical and architecture purposes.

In the future, the variety of communication systems will be merged into one system that handles all types of information (multimedia). There will no longer be a clear distinction between PC, telephone, and television. For this system, storage and transmission of the information will be standardized and designed to be, as much as possible, independent of the application area.

Together with the increase in information exchange and storage, there has also been a change in the type of information: visual information in the form of digital images and video sequences has become increasingly important. This type of information especially gives rise to high transmission and storage costs. Therefore, the development of reliable and fast data compression techniques for several quality levels has become an important research topic.

The research into compression techniques for digital images and video sequences concentrates on two main points. First, there is the need for international standards

for compression. In particular, the network operators, the telecommunication companies, and the microelectronics industry are working on this subject. Further, much research is devoted to the development of new compression techniques for future standards and special-purpose applications. This thesis focuses on this last-mentioned area of research and, using the wavelet theory, it discusses the coding of digital images and video sequences.

Among the various techniques of image and video coding, the discrete wavelet transform (DWT) is a relatively recently developed method. It was introduced in [Mall89a] into the image coding world and invoked a growing interest in the wavelet theory in general. In its application to image and video coding, the DWT resembles the older subband coding (SBC) technique. The purpose of this thesis is to compare the more recent DWT with the older SBC. Further, we propose a new image coding technique combining the DWT with a contour-texture coding method.

In Section 1.1 we start with a discussion of existing coding techniques and standards for the storage and transmission of digital images and video sequences. In Section 1.2 we present the scope and an overview of the contents of this thesis.

1.1 Image and video coding techniques

A standard has been established for the compression of digital images. It was developed by the Joint Photographic Experts Group and the coding standard is accordingly called JPEG [Penn93]. JPEG is based on a division of the images into blocks of 8x8 pixels, after which each block is transformed with the discrete cosine transform (DCT), a transform developed in the seventies. The transform coefficients are quantized, where each frequency is weighted according to its importance to the human visual system (HVS). However, the division into blocks causes a special type of error, which is known as the blocking artefact. Its effect is that images coded at low bit rates seem to be placed behind a rectangular raster. The blocking artefact consists of horizontal and vertical artificial edges in the image, which are perceptibly very annoying, because the HVS has a high degree of sensitivity to horizontal and vertical directions [Marr82]. In Figure 1.1a/b we show an example of an image and its compressed version, generated by JPEG.

Other image coding techniques have been developed, but none of them has been standardized. The most promising technique without the blocking artefact is the subband coding (SBC) technique, which was developed in the seventies for speech coding [Croi76], and applied to image coding in the eighties [Wood86]. The closely

(a)

(b)

(c)

Figure 1.1 Two different compression methods, for a compression factor 27. (a) Original image; (b) The image compressed with JPEG; (c) The image compressed with the DWT.

related discrete wavelet transform (DWT) arose in the middle of the eighties [Meye85][Mall89a]. In Figure 1.1c we show an example of an image coded by the DWT with the coding scheme described in Chapter 4. This coding method has no blocking artefacts and the coding errors are more pleasing to the eye.

Also for video, some coding schemes have been investigated and standardized. For low bit rate applications like videotelephony the Recommendation H.261 [CCIT90] of the International Consultative Committee for Telephone and Telegraph

(CCITT) was internationally adopted in 1991. The Moving Pictures Expert Group of the International Organization for Standardization (ISO/IEC) has been developing a similar, but extended and improved scheme, which is currently at the 'Committee Draft stage' [MPEG-2]. It is a follow-up of MPEG-1, which is intended for the storage of video and associated audio at low to medium bit rates and was standardized in 1992 [MPEG-1]. MPEG-2 is intended for the transmission of video and associated audio at high bit rates, such that it can also be used for digital high definition television (HDTV). In particular, it will be used for the broadcasting of standard digital television. Both Recommendation H.261 and the MPEG coders are hybrid coding schemes, which make use of predictive as well as transform coding techniques. The spatial coding resembles that of JPEG as it is also based on the DCT.

The SBC-related coding techniques are promising, especially for video transmission purposes, because they have an inherent multiresolution structure. This means that several video formats can be extracted from one information stream without any loss in quality for the highest resolution, and without any additional cost. However, the SBC-related techniques do not have as long a history of standardization as the DCT-related techniques. It is therefore to be expected that for medium to high bit rates the DCT will remain the standard. For special-purpose applications and for special circumstances like very low bit rate coding, the SBC-related techniques have a good chance of being adopted as (part of) the standard. For very low bit rates, object-based and knowledge-based coding techniques are also under investigation [ISO93][COST93]. These are commonly referred to as MPEG-4 coding methods.

1.2 Scope and outline of this thesis

This thesis discusses the recently developed wavelet theory with respect to its application to image and video coding. Our purpose is to evaluate what this theory has contributed to the already existing subband coding method. Further, a new image coding technique is presented that can be viewed as a combination of the wavelet transform and a contour-texture coding technique. The coding performance of this new method is compared to existing methods.

Chapter 2 contains an explanation of the wavelet theory. Although there are a number of books on wavelet theory, most books have a more mathematical approach than is necessary for the application of the wavelet theory to the processing of digital signals. This chapter presents the theory from the viewpoint of signal processing,

which is also reflected in the choice of the notation. Further, most abstract mathematical details and the proofs are left out or, in a few cases, deferred to an appendix.

In Chapter 3 we define the basic two-channel analysis/synthesis scheme, which is the basic block of the image and video coding techniques discussed throughout the thesis. It concentrates on the importance of proper signal extension at boundaries and presents a clear illustration of how the signal extension should be implemented in order to guarantee the capability of perfect reconstruction. However, the main theme of the chapter is the introduction of a new signal extension which enables the wavelet decomposition of discrete signals of arbitrary length. The application of this extension is discussed in Chapter 7.

The image and video coding schemes as used in our experiments are described in Chapter 4. As various video coding schemes are possible, we explain our choice for the experiments.

Chapter 5 starts with showing the differences between the filter designs according to subband theory and wavelet theory. For both classes, a typical filter design method is demonstrated and a modification to some widely used subband theory filters is proposed. The chapter continues with an extensive comparison of the coding performances of various filters in image and video coding.

In Chapter 6 the extension of the wavelet transform to two dimensions is discussed. It is followed by a comparison of two different types of two-dimensional wavelet transforms: the dyadic 2-D DWT and the quincunx 2-D DWT.

Finally, Chapter 7 contains a detailed description of a new coding technique, a region-based discrete wavelet transform. It is shown how it can be considered as a combination of two existing coding techniques: the DWT and a contour-texture coding method. First, the implementation of the new method is described in detail and then the chapter is continued by a comparison of the new method with the DWT.

The main body of the thesis is followed by six appendices. Appendix A gives an extensive overview of the transforms and series in the wavelet theory in comparison to the Fourier theory. The formulas are classified and named analogous to those of the Fourier theory. To reveal differences and similarities, corresponding formulas of Fourier theory and wavelet theory are presented in juxtaposition. In Appendix B the extension of the wavelet theory to the biorthogonal case is described, together with a short description of the importance of linear phase for image coding. Some mathematical proofs and derivations of the main text are worked out in Appendix C, and Appendix D considers the estimation of the bit rate needed to code a contour image.

In Appendix E the filter coefficients, filter characteristics, limit functions, and magnitude and phase responses are shown for all filters that appear in this thesis. Finally, Appendix F shows the images and the first frame of the video sequences used in the experiments.

Chapter 2

Wavelet Theory

The purpose of this chapter is to introduce the wavelet theory. We start with a rather mathematical description although we always have the application to signal processing in mind, in particular image and video coding. The reader is assumed to be familiar with the Fourier theory. For the definitions of Fourier formulas that are not shown explicitly in this chapter we refer to the definitions on page xiv.

The theory in this chapter is aimed at being a consistent summary of the various parts of and viewpoints on wavelet theory in several papers and textbooks. The uniform naming and the definitions of all variants of the wavelet transforms and series are new in this chapter. These are based on the corresponding names and definitions of the Fourier transforms and series as given in [Oppe83].

We start with the continuous-time Fourier transform and series in Section 2.1, from which we give reasons for the use of the continuous-time wavelet transform and series. After explaining the important concept of multiresolution analysis in Section 2.2, we define the discrete-time wavelet series and the discrete wavelet transform in Section 2.3 and Section 2.4, respectively. The important concept of regularity is addressed in Section 2.5. Then, a short historical overview of the wavelet theory is presented in Section 2.6. The chapter is closed by Section 2.7, which contains the conclusions.

2.1 The continuous-time wavelet transform and series

In the analysis of stationary continuous signals, the continuous-time Fourier transform (CTFT) is an important tool. The transform describes the signal with a basis consisting of sines and cosines of different frequencies. The weights of these infinitely many basis functions are calculated by taking the inner product of the signal with complex exponential functions, i.e.,

$$F(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx. \quad (2.1)$$

Here $f(x)$ denotes the signal and $F(\omega)$ its Fourier transform. The CTFT has several disadvantages when applied in practice. First, one needs all time information to obtain the transform at a single frequency. This means that the future behaviour of the signal must be known. Further, in practice not all signals are stationary. If a signal has a sharp discontinuity, its Fourier representation needs a lot of sines and cosines to cancel out the waveforms that appear because of the discontinuity. In other words, the discontinuity information is spread out over the whole frequency axis. To overcome these drawbacks of the CTFT, the transform has been slightly modified yielding the short-time Fourier transform (STFT) [Gabo46],

$$STFT_f(\omega, b) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x} w(x-b) dx, \quad (2.2)$$

which windows the continuous signal $f(x)$ before Fourier transforming. The window $w(x-b)$ is a localized function which is shifted over the time axis to compute the transform at several positions b . The transform has become time dependent and thus forms a time-frequency description of the signal. One can also view the STFT as a transform with basis functions that are windowed complex exponentials. The window is often chosen to be a Gaussian, and in this case the inverse transform also has a Gaussian window function.

A drawback of the STFT is that the window is fixed, which means that the window cannot adapt to the characteristics of the signal at a certain point. Therefore, another type of transform has been designed which circumvents this problem: the continuous-time wavelet transform (CTWT), which is defined as the inner product of $f(x)$ with the basis functions

$$\psi_{a,b}(x) = a^{-1/2} \psi\left(\frac{x-b}{a}\right), \quad a \in R^+, b \in R, \quad (2.3)$$

thus

$$CTWT_f(a, b) = a^{-1/2} \int_{-\infty}^{\infty} \psi\left(\frac{x-b}{a}\right) f(x) dx. \quad (2.4)$$

The basis functions $\psi_{a,b} \in L^2(R)$ are real and oscillating. They will either fade away towards infinity and minus infinity, or be zero outside a support interval. The basis functions are called *wavelets* (i.e. short waves) and can be viewed as contracted and shifted versions of the prototype function $\psi(x)$. The argument b is the (time) location variable and a denotes the scale parameter. If the scale parameter is large, then the basis function becomes a stretched version of the prototype, corresponding to a low frequency function. Similarly, small values of the scale parameter a correspond to high frequencies. The windowing is contained in the basis functions and enables the narrowing and widening of the time-frequency window. Thus it is possible to adjust to the characteristics of the signal. In Figure 2.1 the tiling of the time-frequency plane is shown for the STFT and the CTWT. According to the uncertainty principle, a wid-

Figure 2.1 Tiling of the time-frequency plane. (a) Short-time Fourier transform; (b) Continuous-time wavelet transform.

ening in the time domain causes a narrowing in the frequency domain and vice versa. Therefore, the areas of the tiles is constant. We see that the CTWT is able to adapt to the characteristics of the signal.

The function $\psi(x)$ has to satisfy the following admissibility condition,

$$C_\psi = \int_0^\infty \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty, \quad (2.5)$$

in order to enable the reconstruction of $f(x)$ from its CTWT. Here, $\Psi(\omega)$ denotes the Fourier transform of $\psi(x)$. If $\psi(x)$ has some decay at infinity, then (2.5) implies that the function $\psi(x)$ has zero mean,

$$\int_{-\infty}^{\infty} \psi(x) dx = 0. \quad (2.6)$$

The reconstruction formula is then

$$f(x) = \frac{1}{C_\psi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} CTWT_f(a, b) a^{-1/2} \psi\left(\frac{x-b}{a}\right) \frac{dad b}{a^2}. \quad (2.7)$$

The factor $1/a^2$ in the integration is the Jacobian. As we see from (2.7), $f(x)$ can be described as a summation of the basis functions $\psi_{a,b}(x)$ with weights $CTWT_f(a, b)$.

The parameters a and b vary continuously and therefore the set of basis functions is highly redundant. It is possible to discretize the values for a and b , while still being able to reconstruct the signal from its transform. The discretization is

$$a = a_0^m; \quad b = nb_0 a_0^m, \quad m, n \in \mathbb{Z}, \quad a_0 > 1, \quad b_0 \neq 0. \quad (2.8)$$

Here the choice of b_0 is arbitrary since $\psi(x)$ can be adjusted to fix any value of b_0 . For convenience, we choose $b_0 = 1$. We see from this choice that the location parameter depends on the scale parameter. If the scale is large then also the size of the steps in the location increases. This is intuitively correct since a large scale implies that details are not so important.

The corresponding wavelets for the discretized a and b are

$$\psi_{m,n}(x) = a_0^{-m/2} \psi(a_0^{-m} x - n). \quad (2.9)$$

Although the discretized transform is called wavelet transform in some of the literature, it is more analogous with Fourier terminology to call it the continuous-time

wavelet series as it is a discrete representation of a continuous signal. This terminology does not mean that the functions to be represented are periodic. The CTWS is thus defined as a sampling from the CTWT:

$$(CTWS_f)_{m,n} = d_{m,n} = \int_{-\infty}^{\infty} a_0^{-m/2} \psi(a_0^{-m}x - n) f(x) dx. \quad (2.10)$$

The recovering of $f(x)$ from the series representation is possible if there exist numbers $A > 0$ and $B < \infty$ such that

$$A \|f(x)\|^2 \leq \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} |d_{m,n}|^2 \leq B \|f(x)\|^2, \quad (2.11)$$

for all $f(x)$ in $L^2(\mathbb{R})$ [Daub90]. This means that although the reconstruction of $f(x)$ from its wavelet coefficients may not be exact, it is in the neighbourhood of the original function in a mean square sense. If $A = B = 1$ and $a_0 = 2$, then perfect reconstruction is enabled and the set of basis functions $\psi_{a,b}(x)$ forms an orthogonal basis. Then the inverse of the CTWS is

$$f(x) = C_\psi \sum_{m \in \mathbb{Z}} \sum_{n \in \mathbb{Z}} d_{m,n} a_0^{-m/2} \psi(a_0^{-m}x - n). \quad (2.12)$$

If the functions $\psi_{m,n}$ are normalized then $C_\psi = 1$. Now we have defined the continuous-time wavelet transform and series in a form comparable to the CTFT and CTFS (see also Appendix A, Table A.1). Before starting with the discrete versions we introduce the concept of multiresolution, which is another way to introduce the theory of wavelet transforms and series.

2.2 Multiresolution representation of functions

In the analysis of signals, it is often useful to observe a signal in successive approximations. For instance, in pattern recognition applications the vision system attempts to classify an object from a coarse approximation. If the classification does not succeed, additional details are added such that a more accurate view of the object is obtained. This process can be continued until the object has been recognized. The use of multiple resolutions in the recognition process is computationally and conceptually interesting. As another example in signal processing where multiple resolutions are beneficial we mention video coding. Imagine a television screen that is split up into several blocks, each displaying a different TV channel. The viewer may wish to

see all the small blocks simultaneously or may decide to watch one TV channel full size. In these kinds of applications, it is economical to have the information for the small display be part of the full screen information, otherwise the distributor would have to transmit the signals on several scales. Then the receiver uses a small part of the full screen information to build up a small screen display. From these applications of multiple resolutions in pattern recognition and video coding we see that a multiresolution theory is very useful.

The multiresolution theory can best be described by the theory of function spaces. Therefore, we use the mathematical vocabulary in this subsection.

A multiresolution analysis is a description of $L^2(R)$ as a hierarchy of embedded subspaces V_m which have intersection $\{0\}$ and for which the limit of their union is $L^2(R)$, i.e.,

$$\begin{aligned} \dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots, \\ \bigcap_{m \in \mathbb{Z}} V_m = \{0\}, \quad \overline{\bigcup_{m \in \mathbb{Z}} V_m} = L^2(R). \end{aligned} \quad (2.13)$$

Further, these V_m have the property that for each function $f(x) \in V_m$ a contracted version is contained in the subspace V_{m-1} ,

$$f(x) \in V_m \Leftrightarrow f(2x) \in V_{m-1}. \quad (2.14)$$

Finally, in a multiresolution analysis there is a function $\phi(x) \in V_0$ such that its shifted versions $\phi_{0,n}(x) = \phi(x-n)$, $n \in \mathbb{Z}$ constitute an orthonormal basis of V_0 . In Figure 2.2 the embedded subspaces are shown in a schematic way.

Figure 2.2 Multiresolution representation of $L^2(R)$.

As the functions $\phi_{0,n}(x)$ form an orthonormal basis for V_0 , it follows that the functions

$$\phi_{m,n}(x) = 2^{-m/2} \phi(2^{-m}x - n) \quad (2.15)$$

constitute an orthonormal basis for V_m . These basis functions are called *scaling functions* since they build up scaled versions of functions in $L^2(R)$. The multiresolution analysis defined above has the nice implication that a function $f(x)$ in $L^2(R)$ can be seen as a successive approximation by functions $f_m(x)$ in V_m . In other words, the function $f(x)$ is the limit of the approximations $f_m(x) \in V_m$ for m to minus infinity,

$$f(x) = \lim_{m \rightarrow -\infty} f_m(x). \quad (2.16)$$

This creates the possibility to examine the function or signal at several resolutions or scales. The variable m indicates the scale and is called the *scale factor* or *level*. If the scale factor m is high, that is, the scale is large, then the function in V_m is a coarse approximation of $f(x)$, so details are neglected. If the scale factor m is low, a detailed approximation of $f(x)$ is achieved. All functions in V_m can be represented using linear combinations of the scaling functions. In fact, $f_m(x)$ is an orthogonal projection of $f(x)$ onto V_m ,

$$f_m(x) = \sum_n \langle \phi_{m,n}(x), f(x) \rangle \phi_{m,n}(x) = \sum_n c_{m,n} \phi_{m,n}(x). \quad (2.17)$$

Since $\phi(x) = \phi_{0,0}(x) \in V_0 \subset V_{-1}$, we can write

$$\phi_{0,0}(x) = 2^{1/2} \sum_n h_n \phi_{-1,n}(x) = 2 \sum_n h_n \phi(2x - n), \quad (2.18)$$

for a certain sequence h_n , so $\phi_{0,0}(x)$ is a solution of a two-scale difference equation. The constant $2^{1/2}$ in the second term of (2.18) is introduced for later convenience. The sequence is a result of the change in resolution from one subspace to the subspace of one level higher.

The function $\phi(x)$ and the sequence h_n are closely related. Next to (2.18), two other relations can be derived. The first one we derive by generalizing (2.18) to

$$\phi_{m+1,k}(x) = 2^{1/2} \sum_p h_p \phi_{m,p-2k}(x) = 2^{(-m+1)/2} \sum_p h_p \phi(2^{-m}x - (p-2k)), \quad (2.19)$$

and taking the inner product with $\phi_{m,n-2k}(x)$ on both sides,

$$\begin{aligned}
\langle \phi_{m+1,k}(x), \phi_{m,n-2k}(x) \rangle &= \langle 2^{1/2} \sum_p h_p \phi_{m,p-2k}(x), \phi_{m,n-2k}(x) \rangle \\
&= 2^{1/2} \sum_p h_p \langle \phi_{m,p-2k}(x), \phi_{m,n-2k}(x) \rangle = 2^{1/2} h_n.
\end{aligned} \tag{2.20}$$

Note that this relation holds for every m . The second relation starts from the equivalent of (2.18) in Fourier domain,

$$\Phi(\omega) = 2 \sum_n h_n \left(\frac{1}{2} e^{-i(\frac{\omega}{2})n} \Phi\left(\frac{\omega}{2}\right) \right) = \sum_n \left(h_n e^{-i(\frac{\omega}{2})n} \right) \Phi\left(\frac{\omega}{2}\right) = H\left(\frac{\omega}{2}\right) \Phi\left(\frac{\omega}{2}\right), \tag{2.21}$$

where both the CTFT and the DTFT (Discrete-Time Fourier Transform) are used, respectively. Repeating this substitution on the right-hand side of (2.21) yields

$$\Phi(\omega) = \prod_{m=1}^{\infty} H\left(\frac{\omega}{2^m}\right). \tag{2.22}$$

The definition of the multiresolution analysis not only implies strict relations between $\phi(x)$ and h_n , but also creates restrictions on the sequence h_n . In the following, we derive most of these restrictions.

Firstly, the integration of (2.18) on both sides over the whole x -axis produces

$$\sum_n h_n = 1, \tag{2.23}$$

because the function $\phi(x)$ must have non-zero mean in order to build up a multiresolution analysis. Secondly, because of the orthonormality of the basis functions the h_n must also fulfil

$$\begin{aligned}
\delta_{0,k} &= \langle \phi_{0,0}(x), \phi_{0,k}(x) \rangle \\
&= \langle 2^{1/2} \sum_n h_n \phi_{-1,n}(x), 2^{1/2} \sum_p h_p \phi_{-1,p-2k}(x) \rangle = 2 \sum_n h_n h_{n+2k}.
\end{aligned} \tag{2.24}$$

The third property of the sequence h_n has its most concise form in Fourier domain. With the Fourier domain equivalent of the orthonormality of the functions $\phi_{m,n}(x)$,

$$\sum_k |\Phi(\omega + 2k\pi)|^2 = 1, \tag{2.25}$$

it can be derived that

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 1, \tag{2.26}$$

(see Appendix C). Equation (2.23) is equivalent with $H(0) = 1$ in frequency domain and so $H(\pi) = 0$ with (2.26).

These properties of the h_n are used later on. Now we leave for a moment the theory and show a simple example of a set of scaling functions that builds up $L^2(R)$.

Example of a scaling function and corresponding sequence

The simplest example of a set of scaling functions that build up the space $L^2(R)$ is the set of translations and dilations of the unit function on the unit interval.

$$\phi(x) = \begin{cases} 1, & 0 \leq x < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (2.27)$$

For instance, the basis functions of belonging to scaling factor -1 become

$$\phi_{-1,n}(x) = \begin{cases} \sqrt{2}, & n/2 \leq x < (n+1)/2, \\ 0, & \text{otherwise.} \end{cases}$$

The sequence h_n corresponding to this choice of $\phi(x)$ can easily be derived by taking the inner products of two basis functions, one from V_{-1} and the other from V_0 using (2.20). It turns out that the sequence becomes

$$h_n = \begin{cases} 1/2, & n = 0, 1, \\ 0, & \text{otherwise.} \end{cases} \quad (2.28)$$

The basis function and corresponding sequence of this example are plotted in Figure 2.3. In Figure 2.4 we show how a continuous signal from $L^2(R)$ is approximated by functions in the subspaces V_m , when the scaling function is defined by (2.27).

The wavelet representation of functions in $L^2(R)$

When observing the graph of the multiresolution representation (Figure 2.2), we see that the space $L^2(R)$ is built up of the set of 'rings' that are the differences between two consecutive spaces. These difference spaces are denoted by W_m and are defined as the orthogonal complement of the spaces V_m with respect to V_{m-1} ,

$$\begin{aligned} V_{m-1} &= V_m \oplus W_m, & V_m &\perp W_m, \\ \bigcap_{m \in \mathbb{Z}} W_m &= \emptyset, & \overline{\bigcup_{m \in \mathbb{Z}} W_m} &= L^2(R). \end{aligned} \quad (2.29)$$

Let $\psi(x) = \psi_{0,0}(x)$ be a basis function of W_0 . Since $\psi_{0,0}(x) \in W_0 \subset V_{-1}$ we can write

Figure 2.3 Example of a scaling function. (a) $\phi(x)$; (b) $\phi_{0,1}(x)$; (c) $\phi_{1,0}(x)$; (d) The corresponding sequence h_n ; (e) $|H(\omega)|$; (f) $\text{Arg}(H(\omega))$.

$$\psi_{0,0}(x) = 2^{1/2} \sum_n g_n \phi_{-1,n}(x), \quad (2.30)$$

for a certain sequence g_n . The functions $\phi_{m,n}(x)$ are shifted and dilated versions of each other. Therefore, we can also define functions $\psi_{m,n}(x)$ that are shifted and dilated versions of one prototype function $\psi(x)$:

$$\psi_{m,n}(x) = 2^{-m/2} \psi(2^{-m}x - n). \quad (2.31)$$

The functions $\psi_{m,n}(x)$ are identical to the wavelets in Section 2.1 after the discretization of (2.8). The parameter a_0 in (2.9) has been fixed to 2. They form an orthonormal basis for $L^2(R)$.

There are strong relations between $\psi(x)$, $\phi(x)$, g_n , and h_n . The first we derive here is the analog of (2.22). Rewriting (2.30) in frequency domain,

$$\Psi(\omega) = G\left(\frac{\omega}{2}\right) \Phi\left(\frac{\omega}{2}\right), \quad (2.32)$$

and replacing $\Phi(\omega)$ using the infinite product of (2.22) yields

(a)

(b)

(c)

(d)

Figure 2.4 Approximation of a signal in $L^2(R)$ in three successive subspaces V_{m-1} , V_m , and V_{m+1} . (a) The original signal $f(x)$; (b) - (d) The approximations $f_{m-1}(x)$, $f_m(x)$, and $f_{m+1}(x)$ with the scaling function of Figure 2.3.

$$\Psi(\omega) = G\left(\frac{\omega}{2}\right) \prod_{m=2}^{\infty} H\left(\frac{\omega}{2^m}\right). \quad (2.33)$$

Note that $\Psi(\omega)$ is not an infinite product of $G(2^{-m}\omega)$ but of $H(2^{-m}\omega)$ instead, as in (2.30) the wavelet $\psi(x)$ is written as a linear combination of the scaling functions.

Next, we derive a relation between the two sequences g_n and h_n . Since W_m is the orthogonal complement of V_m , the functions $\psi_{0,0}(x)$ and $\phi_{0,0}(x)$ must be orthogonal,

$$0 = \langle \phi_{0,0}, \psi_{0,0} \rangle = 2 \sum_n \sum_p h_n g_p \langle \phi_{-1,n}, \phi_{-1,p} \rangle = 2 \sum_n h_n g_n, \quad (2.34)$$

using (2.18) and (2.30). We see that the choice

$$g_n = (-1)^n h_{-n+2t+1} \quad (2.35)$$

gives a correct solution for all $t \in \mathbb{Z}$. The equivalent of (2.35) in frequency domain is

$$G(\omega) = -H(-\omega + \pi) e^{-i\omega(2t+1)}, \quad (2.36)$$

and with this we can rewrite (2.32) as

$$\Psi(\omega) = -e^{-i\omega/2} H\left(-\frac{\omega}{2} + \pi\right) \Phi\left(\frac{\omega}{2}\right), \quad (2.37)$$

where we took $t = 0$ for convenience.

Finally, note that both the function $\psi(x)$ and the sequence g_n have zero mean. This can easily be verified by the substitution of $\omega = 0$ in (2.37) and (2.36), respectively, and using the fact that $H(\pi) = 0$:

$$\Psi(0) = \int_{-\infty}^{\infty} \psi(x) dx = 0, \quad (2.38)$$

and

$$G(0) = \sum_n g_n = 0. \quad (2.39)$$

The introduction of the wavelet functions enables us to write any function $f(x)$ in $L^2(\mathbb{R})$ as a sum of projections on W_j , $j \in \mathbb{Z}$,

$$\tilde{f}(x) = \sum_{j=-\infty}^{\infty} e_j(x), \quad (2.40)$$

where

$$e_j(x) = \sum_k \langle \psi_{j,k}(x), f(x) \rangle \psi_{j,k}(x). \quad (2.41)$$

If we stop at a certain scale m , then the function $f(x)$ can be written as the sum of a low resolution part $f_m(x) \in V_m$ and a number of detail parts $e_j(x) \in W_j$,

$$\begin{aligned} f(x) &= f_m(x) + \sum_{j=-\infty}^m e_j(x) \\ &= \sum_n \langle \phi_{m,n}(x), f(x) \rangle \phi_{m,n}(x) + \sum_{j=-\infty}^m \sum_k \langle \psi_{j,k}(x), f(x) \rangle \psi_{j,k}(x) \quad (2.42) \\ &= \sum_n c_{m,n} \phi_{m,n}(x) + \sum_{j=-\infty}^m \sum_k d_{j,k} \psi_{j,k}(x). \end{aligned}$$

Example of a wavelet function and corresponding sequence

As an example of a set of wavelet functions that forms a complete orthonormal set building up $L^2(R)$, we show in Figure 2.5 the wavelet that is related to the scaling function of the example given in Figure 2.3. This set of wavelets was first published by Haar [Haar10] and the function $\psi(x)$ is therefore called the Haar wavelet.

Biorthogonality

The multiresolution analysis can be extended to biorthogonal basis functions in such a manner that the description above is contained in it. Biorthogonal basis functions permit the sequences h_n and g_n to be symmetric whereas their length exceeds 2. This is important in image coding applications. Appendix B describes the extension of the multiresolution analysis to biorthogonal basis functions. It also shows why symmetric sequences are important in image coding, and the reason why symmetry cannot be combined with orthogonality.

2.3 The discrete-time wavelet series

In most of our applications the signal that we wish to transform is not continuous but discrete. Therefore, we are interested in discrete versions of the CTWT and CTWS

Figure 2.5 Example of a wavelet. (a) $\psi(x)$; (b) $\psi_{0,1}(x)$; (c) $\psi_{1,0}(x)$; (d) The corresponding sequence g_n ; (e) $|G(\omega)|$; (f) $\text{Arg}(G(\omega))$.

that convert the discrete signal into a continuous and a discrete signal, respectively. The formulas for the discrete-time wavelet transform and series (DTWT and DTWS) are not just discretized versions of the formulas for the CTWT and CTWS. Neither is it possible to define a multiresolution analysis for discrete signals. There is no discrete basis function of which the scaled and shifted versions are basis functions of subspaces of $l^2(\mathbb{R})$, the space of square summable infinite length sequences.

We will derive in this section the formulas for the DTWS from the multiresolution analysis of Section 2.2. In Appendix A the formulas for the wavelet transforms and series are tabulated in a clear form. They are compared to the analogous transforms and series in Fourier theory.

We start our derivation of the DTWS by the observation that every discrete signal c_n can be viewed as the sequence of weights of a set of scaling functions $\phi_{0,n}(x)$ representing a continuous function $f_0(x) \in V_0$. With the definition $c_{0,n} = c_n$ this is

$$f_0(x) = \sum_n c_{0,n} \phi_{0,n}(x). \quad (2.43)$$

In other words, we interpret the sequence c_n as if it were the representation $c_{0,n}$ of a certain continuous function $f_0(x) \in V_0$ for a certain multiresolution analysis of $L^2(\mathbb{R})$. From this function $f_0(x)$ in V_0 we compute smoother versions lying in the spaces $V_1, V_2, \dots$. The spaces $V_{-1}, V_{-2}, \dots$ have no meaning in this interpretation.

Following the multiresolution idea the function $f_0(x)$ is decomposed into two functions $f_1(x) \in V_1$ and $e_1(x) \in W_1$ containing the overall characteristics and the details of $f_0(x)$, respectively:

$$f_0(x) = f_1(x) + e_1(x) = \sum_k c_{1,k} \phi_{1,k}(x) + \sum_k d_{1,k} \psi_{1,k}(x).$$

Now two new sequences $c_{1,n}$ and $d_{1,n}$ have been generated. This process can be continued on $f_1(x)$ and so we can represent $f_0(x)$ (and also the sequence c_n) with the set of coefficients $d_{m,n}$, $m \in \mathbb{Z}^+$, $n \in \mathbb{Z}$.

Now we have defined the concept of the DTWS. However, the calculation seems to depend on the continuous functions $\phi(x)$ and $\psi(x)$. Therefore, we show how the computation of the DTWS can be performed using only operations on discrete signals.

Using the property of $\phi_{1,n}(x)$ constituting an orthonormal basis for V_1 we get

$$\begin{aligned} c_{1,k} &= \langle \phi_{1,k}(x), f_1(x) \rangle = \langle \phi_{1,k}, f_0(x) - e_1(x) \rangle = \langle \phi_{1,k}(x), \sum_n c_{0,n} \phi_{0,n}(x) \rangle \\ &= \sum_{n \in \mathbb{Z}} c_{0,n} \langle \phi_{1,k}(x), \phi_{0,n}(x) \rangle = 2^{1/2} \sum_{n \in \mathbb{Z}} c_{0,n} h_{n+2k}, \end{aligned} \quad (2.44)$$

using (2.20). Thus, it is possible to calculate iteratively coefficients $c_{j,k}$ and $d_{j,k}$ from the previous scale $j-1$ without the explicit use of the functions $\phi(x)$ and $\psi(x)$. Analogous to (2.44) we derive for arbitrary j ,

$$c_{j,k} = 2^{1/2} \sum_n c_{j-1,n} h_{n+2k}, \quad (2.45)$$

$$d_{j,k} = 2^{1/2} \sum_n c_{j-1,n} g_{n+2k}, \quad (2.46)$$

with which the description of the decomposition process is now completely discrete. The sequences h_n and g_n are called *filters*. Note that $c_{j,k}$ and $d_{j,k}$ have each 'half' the length of $c_{j-1,k}$ (both are of infinite length in this stadium). Therefore, no redundancy is introduced.

The inverse process is the reconstruction of c_{j-1} out of c_j and d_j :

$$\begin{aligned}
c_{j-1,n} &= \langle \phi_{j-1,n}, f_{j-1}(x) \rangle = \langle \phi_{j-1,n}(x), f_j(x) + e_j(x) \rangle \\
&= \langle \phi_{j-1,n}(x), \sum_k c_{j,k} \phi_{j,k}(x) \rangle + \langle \phi_{j-1,n}(x), \sum_k d_{j,k} \psi_{j,k}(x) \rangle \\
&= \sum_k c_{j,k} \langle \phi_{j-1,n}(x), \phi_{j,k}(x) \rangle + \sum_k d_{j,k} \langle \phi_{j-1,n}(x), \psi_{j,k}(x) \rangle \\
&= 2^{1/2} \sum_k c_{j,k} h_{n+2k} + 2^{1/2} \sum_k d_{j,k} g_{n+2k}.
\end{aligned} \tag{2.47}$$

Note that here the summation is over the other variable as compared to the summation in (2.45) and (2.46). The length of the sequence c_{j-1} is twice the length of c_j and d_j .

By substituting (2.45) and (2.46) in (2.47) and vice versa we get the following restrictions on the filters h_n and g_n :

$$2 \sum_k (h_{n+2k} h_{p+2k} + g_{n+2k} g_{p+2k}) = \delta_{n,p}, \tag{2.48}$$

$$2 \sum_n h_{n+2k} h_{n+2p} = 2 \sum_n g_{n+2k} g_{n+2p} = \delta_{k,p}, \tag{2.49}$$

$$2 \sum_n h_{n+2k} g_{n+2p} = 0. \tag{2.50}$$

Equation (2.48) is the time domain equivalent of (2.26) combined with (2.36) (see Appendix C for equivalent representations in both domains). Equations (2.49) and (2.50) appeared before in a less general form in (2.24) and (2.34), respectively.

2.4 The discrete wavelet transform

In practical applications the DTWS has to be applied to discrete signals which have finite length. As the DFT works on finite length signals and transforms them to signals of equal length, we also want to modify the DTWS to get a discrete wavelet transform (DWT) which transforms signals into other signals of the same length.

First, we describe the DWT as a series of matrix operations, and after that we describe it as a filter bank implementation. The filter bank description is most often used in signal processing.

For both types of description we assume basis functions $\phi(x)$ and $\psi(x)$ that have finite support, which automatically guarantees that the sequences h_n and g_n are of finite length. In [Daub88] and [Cohe92] several basis functions of compact support are presented.

Further, we assume that the signal to be transformed has a length $N = 2^d$, $d \in \mathbb{Z}^+$, which is also assumed by the DFT.

Matrix description

We have the finite sequence $c_{j,n}$ for a certain value of j and denote it by the vector $\mathbf{v}^j$. This vector is transformed into another vector $\mathbf{v}^{j+1}$ which contains the sequences $c_{j+1,n}$ and $d_{j+1,n}$ in sequel, thus together being the same length as the original. The transform is a matrix multiplication $\mathbf{v}^{j+1} = \mathbf{M}_j \mathbf{v}^j$ [Pres92], where the matrix $\mathbf{M}_j$ is square and consists of elements which are 0 or elements of h_n multiplied by $\sqrt{2}$. Because of the properties of h_n derived in the previous sections, the matrix $\mathbf{M}_j$ is orthonormal and its inverse equals its transpose. As an example, we take a filter length $L = 4$, a sequence length of $N = 8$, and define the starting value $j = 0$. The relation (2.35) between h_n and g_n is applied with $t = L/(2-1) = 4$. The matrix-vector multiplication becomes

$$\begin{bmatrix} c_{1,0} \\ c_{1,1} \\ c_{1,2} \\ c_{1,3} \\ d_{1,0} \\ d_{1,1} \\ d_{1,2} \\ d_{1,3} \end{bmatrix} = \sqrt{2} \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & & & & \\ & h_0 & h_1 & h_2 & h_3 & & & \\ & & h_0 & h_1 & h_2 & h_3 & & \\ h_2 & h_3 & & & & & h_0 & h_1 \\ h_3 & -h_2 & h_1 & -h_0 & & & & \\ & h_3 & -h_2 & h_1 & -h_0 & & & \\ & & h_3 & -h_2 & h_1 & -h_0 & & \\ h_1 & -h_0 & & & & & h_3 & -h_2 \end{bmatrix} \begin{bmatrix} c_{0,0} \\ c_{0,1} \\ c_{0,2} \\ c_{0,3} \\ c_{0,4} \\ c_{0,5} \\ c_{0,6} \\ c_{0,7} \end{bmatrix}. \quad (2.51)$$

The inverse operation is the multiplication of $\mathbf{v}^{j+1}$ with the inverse matrix $\mathbf{M}_j^T$,

$$\begin{bmatrix} c_{0,0} \\ c_{0,1} \\ c_{0,2} \\ c_{0,3} \\ c_{0,4} \\ c_{0,5} \\ c_{0,6} \\ c_{0,7} \end{bmatrix} = \sqrt{2} \begin{bmatrix} h_0 & & h_2 & h_3 & & & h_1 & \\ h_1 & & h_3 & -h_2 & & & -h_0 & \\ h_2 & h_0 & & h_1 & h_3 & & & \\ h_3 & h_1 & & -h_0 & -h_2 & & & \\ & h_2 & h_0 & & h_1 & h_3 & & \\ & h_3 & h_1 & & -h_0 & -h_2 & & \\ & & h_2 & h_0 & & h_1 & h_3 & \\ & & h_3 & h_1 & & -h_0 & -h_2 & \end{bmatrix} \begin{bmatrix} c_{1,0} \\ c_{1,1} \\ c_{1,2} \\ c_{1,3} \\ d_{1,0} \\ d_{1,1} \\ d_{1,2} \\ d_{1,3} \end{bmatrix}. \quad (2.52)$$

Up till now only one step of the DWT has been shown. The complete DWT consists of iteratively applying a matrix multiplication to the upper half of $\mathbf{v}^{j+1}$ with the

square matrix M_{j+1} that has size 2^{d-j} . This procedure can be repeated d steps until the vector is length 1.

When looking at the matrix, we see that at the fourth and eighth row of M_0 in (2.51) the sequence h_n has shifted circularly: the coefficients that do not fit on the right side of the matrix are put on the left side in the same row. In practice, this means that the DWT is exactly one period of length N of the DTWS of the signal $\tilde{c}_{0,n}$, which is the infinite periodic extension of $c_{0,n}$. Thus, the DWT as defined in this way makes essentially use of periodicity as is also the case with the DFT.

The matrix notation gives a concise description of the DWT, but in signal processing the DWT is usually described by a block diagram which is similar to a subband splitting system.

Filter bank description

In subband coding terminology the equations (2.45) and (2.46) are interpreted as a filtering followed by a downsampling of two. As there are two filters h_n and g_n , the filter bank is size 2 and the scheme of this two-band filter bank becomes as shown in Figure 2.6.

Figure 2.6 Two-band filter bank scheme.

As is usual in signal processing, the filtering blocks are indicated by the DTFTs of the filters. The filters F and E denote the filtering by h_{-n} and g_{-n} , respectively. In the lower branch the signal is filtered with $2^{1/2}h_n$, which amounts to a low-pass filtering. The signal yielded shows the original signal on a coarser scale, that is, with fewer details. It is called the low-pass (sub)band. The upper branch of the scheme creates the high-pass band. We remark that in signal processing the constants $2^{1/2}$ are always removed from the filter blocks and shifted towards the right side as a multiplication by 2 (see Figure 3.2).

The scheme only splits up the signal at level $j = 0$ into two signals at level $j = 1$. If the low-pass band is iteratively split up into two branches, we get on the analysis side the scheme shown in Figure 2.7.

Figure 2.7 Subband splitting tree where only the low-pass subbands are split up to a depth of 4.

We recognize in this procedure the matrix implementation of the DWT. However, the implementation differs slightly. In the filtering of a finite length signal one has to face the problem of signal extension at the boundary. The matrix implementation of the DWT is equivalent to a periodic signal extension, and this type of extension is obligatory for orthogonal wavelet filters. When using biorthogonal wavelet filters, one has the possibility to apply other extensions when the filters are symmetric. The subject of signal extension is discussed in detail in Chapter 3.

The scheme shown in Figure 2.8 is equivalent to the scheme in Figure 2.7, where the repeated filtering and downsampling is replaced by one filtering and downsampling per band. The definition of the iterated filters h_n^j and g_n^j is most easily described in the frequency domain:

$$H^j(\omega) = H(\omega)H(2\omega)H(4\omega)\dots H(2^{j-1}\omega) = \prod_{m=1}^j H(2^{m-1}\omega), \quad (2.53)$$

$$G^j(\omega) = G(\omega)H(2\omega)H(4\omega)\dots H(2^{j-1}\omega) = G(\omega) \prod_{m=2}^j H(2^{m-1}\omega). \quad (2.54)$$

In the limit, the iterated filter $H^j(2^{-j}\omega)$ converges to $\Phi(\omega)$ according to the definition (2.22),

$$\lim_{j \rightarrow \infty} H^j(2^{-j}\omega) = \lim_{j \rightarrow \infty} \prod_{m=1}^j H(2^{m-1-j}\omega) = \lim_{j \rightarrow \infty} \prod_{p=1}^j H(2^{-p}\omega) = \Phi(\omega). \quad (2.55)$$

The time domain equivalent implies that the sequence $2^j h_n^j$ plotted against 2^{-jn} converges to $\phi(x)$ when j approaches infinity. This is shown in Figure 2.9 for Daubechies' filter of length 4 [Daub88].

Figure 2.8 The scheme of Figure 2.7 rewritten in an equivalent scheme.

The definition of the DWT can be rewritten in a form that resembles the definition of the DFT. We assume that the signal to be transformed, c_n , has length $N = 2^d$ and is made periodic with period N , which is denoted by $\tilde{c}_n$. Then we get

$$\begin{aligned} (DWT_c)_{j,k} &= 2^{j/2} \sum_{n=-\infty}^{\infty} g_{n-2^j k}^j \tilde{c}_n, & \begin{cases} 1 \leq j < d, \\ 0 \leq k < N2^{-j}, \end{cases} \\ (DWT_c)_{j,k} &= 2^{j/2} \sum_{n=-\infty}^{\infty} h_{n-2^j k}^j \tilde{c}_n, & \begin{cases} j = d, \\ k = 0. \end{cases} \end{aligned} \quad (2.56)$$

Note that the summation is finite, in fact, because the iterated filters have finite length. The DTWS can be defined similarly as

$$(DTWS_c)_{j,k} = 2^{j/2} \sum_{n=-\infty}^{\infty} g_{n-2^j k}^j c_n, \quad j \geq 0, k \in \mathbb{Z}, \quad (2.57)$$

Figure 2.9 The convergence of the scaled iterated filters to the scaling function for Daubechies filter of length 4. (a) - (e) The scaled iterated filters $2^j h_n^j$ for $j = 1, \dots, 5$ plotted against $2^{-j}n$; (f) The limit function $\phi(x)$.

(see also Appendix A, Table A.2). Note that (2.57) is not a discretized version of the CTWS defined in (2.10), because $\psi(x)$ has been replaced by the sequence $2^{1/2}g_n$. However, because of (2.33) and (2.54) the discrete formula converges in the limit to the continuous version.

2.5 Regularity

In this section we discuss the property of regularity for the basis functions $\phi(x)$ and $\psi(x)$. First, an intuitive definition of regularity is given followed by a reason why regularity should be considered in the application of the DWT. The rest of the section explains how the regularity of a basis function and its corresponding filter can be determined.

Intuitively, regularity can be understood to be smoothness. The example given in Figure 2.9 is continuous, but not very smooth. The filter in Figure 2.10 is smoother and its basis function indeed has a higher degree of regularity. If we demand a

Figure 2.10 The scaled iterated filters $2^j h_n^j$ and the limit function for a biorthogonal filter of length 7. (a) - (c) The first three iterations; (d) The limit function.

smooth basis function $\psi(x)$ with which we analyse our signal, we are ensured that no artificial discontinuity will appear in the transform coefficients. In other words, every discontinuity in the transform coefficients is caused by the signal itself. Therefore, smooth (regular) basis functions lead to a better representation of the signal. Another argument for smoothness of the basis functions is that any quantization error in a

transform coefficient of a certain level produces in the reconstruction an error signal that is proportional to the basis functions of that level. In image coding applications, a smooth error signal is less annoying to the eye than a discontinuous error signal. This holds even if their mathematical distortion is the same [Riou92a][Anto92].

Therefore, the concept of regularity seems a good criterion on which to base the design of filters for image coding applications. In Chapter 5 we give a performance comparison between filters which have and have not been designed taking regularity into account. The application area is image coding. In the same chapter we also consider how the regularity condition influences the filter design process.

Several people have designed methods for determining the regularity of a basis function. Here, we consider the Hölder regularity of [Riou92a] since it has the clearest derivation. Rioul's method consists of translating all regularity properties of $\phi(x)$ into equivalent properties of the iterated filters h_n^j . In his paper he explains only the method, leaving the strict conditions, details and proofs to another work on regularity [Riou92b]. We refer also to this work for that kind of information.

The regularity of the scaling function $\phi(x)$ is equal to the regularity of the corresponding wavelet $\psi(x)$, because both can be written as an infinite number of convolutions with the same filter (2.22), (2.33). Therefore, we consider only $\phi(x)$ in this section.

The regularity order of the function $\phi(x)$ is defined to be N if it is N times differentiable and its N th derivative $\frac{d}{dx}\phi(x)$ is continuous. Note that the regularity order is an integer. It is the largest integer less than the Hölder regularity, which is a real number.

A regularity order of N requires at least N zeros of the corresponding filter $H(\omega)$ at $\omega = \pi$. The relation between the number of zeros at half the sampling frequency and the regularity order is not fixed, because other zeros in $H(\omega)$ might disturb the regularity. The problem in the determination of the regularity order is to ratify this effect.

If $2^{-(N+1)}F(\omega)$ represents the remaining factor of $H(\omega)$ when all N zeros at $\omega = \pi$ have been removed, then we can write

$$H(\omega) = 2^{-(N+1)} (1 + e^{-i\omega})^N F(\omega). \quad (2.58)$$

As stated above, some regularity is lost due to the zeros of $F(\omega)$. If $F(\omega)$ has no zeros then the regularity order is N . Otherwise, a lower bound for the regularity R is given by $R = N - \beta$, where the loss in regularity is

$$\beta = \lim_{j \rightarrow \infty} \beta_j = \lim_{j \rightarrow \infty} \frac{1}{j} {}^2\log \left(\max_{0 \leq n \leq 2^j - 1} \left[\sum_k |f_{n-2^j k}^j| \right] \right). \quad (2.59)$$

As in (2.53) f_n^j is the iterated filter belonging to $F^j(\omega)$. In general, the value of β_j is close to β for 20 iterations ($j = 20$).

Because the computation increases exponentially in complexity and memory requirements, often a linear approximation is used,

$$\beta = \lim_{j \rightarrow \infty} \beta_j \approx \lim_{j \rightarrow \infty} \frac{1}{j} {}^2\log \left(\max \left[\sum_k |f_{2^j k}^j|, \sum_k |f_{2^j-1-2^j k}^j| \right] \right), \quad (2.60)$$

which gives a good underestimation for β , and thus an upper bound for the regularity. This formula is equivalent to a much simpler formula which is used to calculate the regularities of the filters in this thesis. It makes use of the matrix F defined by

$$F_{m,n} = f_{2m-n+1}, \quad 0 \leq m, n \leq L_f - 2, \quad (2.61)$$

where L_f is the length of the sequence f_n . The spectral radius ρ of $F_{m,n}$, that is, the largest modulus of the eigenvalues of $F_{m,n}$, determines the upper bound for the regularity,

$$R = N - {}^2\log(\max(|f_0|, |f_{L_f-1}|, \rho)). \quad (2.62)$$

Signals that are built up from polynomials can be decomposed efficiently by filters that have multiple zeros at half the sampling frequency. This can be concluded by observing the equivalent of the derivatives of $\Psi(\omega)$ at $\omega = \pi$ in the time domain. Recall that the relation between $\Psi(\omega)$ and $\Phi(\omega)$ is

$$\Psi(\omega) = -e^{-i\omega/2} H\left(-\frac{\omega}{2} + \pi\right) \Phi\left(\frac{\omega}{2}\right). \quad (2.63)$$

When we take the derivatives of $\Psi(\omega)$ with respect to ω and substitute $\omega = 0$ we get

$$\left. \frac{d^p}{d\omega^p} \Psi(\omega) \right|_{\omega=0} = \left. \frac{d^p}{d\omega^p} H\left(-\frac{\omega}{2} + \pi\right) \right|_{\omega=0} = 0, \quad p = 0, \dots, N-1, \quad (2.64)$$

because $\Phi(0) = 1$. From (2.64) we derive

$$\left. \frac{d^p}{d\omega^p} \Psi(\omega) \right|_{\omega=0} = \left. \frac{d^p}{d\omega^p} \int_{-\infty}^{\infty} \psi(x) e^{-i\omega x} dx \right|_{\omega=0} = (-i)^p \int_{-\infty}^{\infty} x^p \psi(x) dx = 0, \quad (2.65)$$

which implies that the function $\psi(x)$ has N zero moments if $H(\omega)$ has N zeros at $\omega = \pi$. The analog for the filter g_n follows in a direct way,

$$\sum_n n^p g_n = 0, \quad p = 0, \dots, N-1. \quad (2.66)$$

This implies that if the input signal is a polynomial of order less than N , all wavelet coefficients will become zero. Therefore, regular basis functions should be able to represent polynomial signals in an efficient way. In [Zett90] is stated that natural images are of piecewise polynomial nature, where the order of the polynomials is generally 2. Therefore, they assert that a regularity order 2 should be sufficient for image coding purposes. This is discussed in Chapter 5.

2.6 Short history of the wavelet theory

The wavelet theory appeared for the first time in signal processing literature for the analysis of seismic data in a paper by Grossmann and Morlet [Gros84]. Because their interest was in analysis, their set of basis functions was redundant. The mathematician Meyer worked further on the ideas presented in this paper. He showed that there are wavelets that form an orthonormal basis of $L^2(R)$ [Mey85]. The discretization of the wavelet transform was described by Daubechies [Daub90], who, with her paper, made a bridge between the mathematical and signal processing world. She designed a family of wavelet filters which had maximal regularity with respect to the finite filter length. The popularity of wavelets increased after Mallat introduced the concept of multiresolution analysis [Mall89a], which he applied to image coding.

Both Daubechies and Mallat showed that the practical implementation of the discrete wavelet transform amounts to a two-band splitting scheme as is also applied in subband coding. In fact, the subband coding theory, as introduced in the seventies by [Croi76], can be described in wavelet theory terminology and vice versa. The main difference between the two coding methods lies in the design of the filters and their characteristics (Chapter 5).

Some of the ideas of wavelet theory had been partly developed earlier. For instance, Haar published in 1910 [Haar10] a complete orthonormal system of compactly supported basis functions. It equals the simplest wavelet which is now called

the Haar wavelet. In [Asla68], the basis functions of the wavelet transform appeared under the name of coherent states. Recently, the wavelet theory has been extended to cover also the transformation of functions defined on a finite interval [Coh93b].

Today, the wavelet transform has been explained in many signal processing books (e.g. [Akan92]) and there are several books dedicated only to wavelets. A very mathematical description can be found in Meyer's book [Mey85]. A good introduction to wavelets, though still more from a mathematical viewpoint than from a signal processing viewpoint, is Chui's book [Chui92a]. The issue of filter design using wavelets is addressed among other subjects in Vaidyanathan's book [Vaid93]. A book on the application of wavelets in several research areas is [Mey93], also written by Meyer.

2.7 Conclusions

In this chapter we have presented the wavelet theory from the viewpoint of signal processing. We have derived the definition of the CTWT as a logical extension to the CTFT and STFT. After having defined the concept of multiresolution analysis we have concentrated on the processing of discrete signals. The DWT has been described with a matrix and a filter bank derivation and the regularity of filters has been discussed.

Chapter 3

Wavelet Decomposition of Arbitrary Length Signals

Compression of digital signals is often performed with a two-band subband/wavelet decomposition scheme. Conventional tree-structured schemes with depth d that are based on this two-band scheme require an input signal of which the length is a multiple of 2^d . Normally, if the input signal does not meet this condition, samples are added until the requirement is met. However, these additional samples lead to an increase in data. In this chapter, a new technique is presented that makes use of an efficient means of signal extension in the filtering stage. With this method, signals of arbitrary length N can be decomposed into subbands up to arbitrary level without an increase in data. The new image coding method described in Chapter 7 is based on this efficient signal extension.

In Section 3.1, we introduce the problem of the signal length. In the next two sections we examine in detail the implementation of the two-band splitting scheme. We discuss the two conditions for perfect reconstruction: filter design (Section 3.2), and signal extension at the boundaries (Section 3.3). Our new efficient signal extension is described in Section 3.4, after which Section 3.5 discusses the new extension method for more-dimensional signals and multi-channel analysis/synthesis systems. In Section 3.6, a new symmetric-periodic extension method is presented for filtering even length signals with symmetric odd length filters. This method can be used in applications where the wavelet transform is calculated in the frequency domain. All

signal extensions in this chapter are illustrated visually with diagrams to clearly demonstrate the perfect reconstruction conditions. Finally, conclusions are drawn in Section 3.7.

3.1 Introduction

In the area of image and video coding, the subband decomposition (SBD) and the discrete wavelet transform (DWT) are two popular almost similar techniques. Most applications are based on the two-channel scheme in which the original signal is split up into two subbands half the size of the original. This process can be applied repeatedly to both subbands thus producing a tree-structured splitting of a certain number of levels. The subband coefficients are quantized and transmitted, and at the receiver the original signal is reconstructed by filtering and adding the subbands in reverse order.

Perfect reconstruction of the original signal at the receiver is only possible when quantization is left out. We assume this in the remainder of this chapter. However, also with quantization, the scheme should be implemented with (nearly) perfect reconstruction filters to prevent unnecessary distortions. Perfect reconstruction (PR) depends on two conditions:

- proper design and implementation of the filters in the analysis and synthesis stage;
- proper extension of the finite length signals at the boundary.

In most of the literature, this last aspect is not considered. However, the subject deserves attention because a proper extension method can improve coding efficiency as we show in Section 3.4 with our new signal extension method.

The introduction of the new signal extension method is motivated by the following observation. Whereas block transforms like the discrete cosine transform (DCT) depend on a fixed block size, each level of a conventionally implemented tree-structured DWT depends on the signal being of even length. After downsampling of an odd length signal, we would either lose some information, or the total number of samples would increase by 1. Thus, for a tree-structured decomposition of depth d , the original signal must have a length which is a multiple of 2^d . If the signal does not have such a length, then it has to be padded out with extra samples until the length is a multiple of 2^d . In Figure 3.1a, we show how a signal of length 67 must be extended to a length of 72 to enable a splitting tree of depth 3. Clearly, the decomposition shown in Figure 3.1b would be better for coding purposes, since there is no increase

in data. It can be proven that this decomposition is possible, while preserving the perfect reconstruction property. We show this in Section 3.4 with the introduction of our efficient signal extension method.

The efficient signal extension method has wide applicability. For instance, in image archiving systems using a compression method based on DWT, the images are generally decomposed into subbands up to 3 or 5 levels. If a 226x243 sized image needs to be split up into 3 levels for proper storage, then usually the image is padded out with extra samples to make the size 232x248, which are the nearest (greater) mul-

Figure 3.1 Tree-structured subband decomposition up to level 3 of a one-dimensional signal of length 67. The lengths of the subbands are indicated by the numbers in the circles. (a) With ordinary signal extension methods, 5 extra samples are needed to reach level 3; (b) with the new efficient signal extension, the total number of samples equals the signal length at each level.

tibles of 2^3 . This leads to an increase of 4.8% in the number of subband coefficients that have to be quantized. Because of the new method this increase in data is not necessary at all.

In other words, we show how we can apply a one-level DWT decomposition to a signal of odd length N producing two subbands which have N coefficients in total. No additional subband samples are created that would decrease the coding performance. For tree-structured DWT implementations, this means that signals of *any* length N can be split up to *any* level and, in principle, up to N subbands of length 1, without losing the PR property.

3.2 Proper design and implementation of filters for perfect reconstruction

The two-channel analysis/synthesis scheme is shown in Figure 3.2. All filters are defined to be causal and have a separate operation for shifting the filters denoted by the multiplication with a complex exponential [Mart91]. The input signal S is split into two parts by filtering with the low-pass filter H which is shifted over p samples and by filtering with the high-pass filter G which is shifted over q samples. Both generated signals are downsampled by a factor 2, that is, samples 0, 2, 4, ... are kept and samples 1, 3, 5, ... are thrown away. These signals are assumed to be coded in a loss-

Figure 3.2 Two-channel analysis/synthesis scheme. The filters H and F are low-pass filters and the filters G and E are high-pass filters.

less manner (not shown in the diagram) and transmitted error-free over the channel to the receiver side. Here, the signals are decoded, upsampled by a factor 2, and filtered with shifted synthesis filters F and E . After multiplication by 2 to restore the amplitude, both signals are added to obtain back the original signal in the case of perfect reconstruction filters.

The overall Fourier domain description of the input-output system is

$$O(\omega) = [F(\omega)H(\omega)e^{j\omega(p+r)} + E(\omega)G(\omega)e^{j\omega(q+s)}]S(\omega) + [F(\omega)H(\omega+\pi)e^{j\omega(p+r)}e^{jp\pi} + E(\omega)G(\omega+\pi)e^{j\omega(q+s)}e^{jq\pi}]S(\omega+\pi), \quad (3.1)$$

where O is the output signal (see Appendix C). The second part of (3.1) represents the aliasing distortion and has to be removed for perfect reconstruction. This can be established by choosing $p+r = q+s$ and either

$$F(\omega) = G(\omega+\pi), \quad E(\omega) = -H(\omega+\pi), \quad |p-q| \bmod 2 = 0 \quad (3.2)$$

or

$$F(\omega) = G(\omega+\pi), \quad E(\omega) = H(\omega+\pi), \quad |p-q| \bmod 2 = 1 \quad (3.3)$$

as also demonstrated in [Mart91]. In the sequel we take (3.2), for which the transfer function becomes

$$T(\omega) = [H(\omega)G(\omega+\pi) - G(\omega)H(\omega+\pi)]e^{j\omega(p+r)}. \quad (3.4)$$

For PR, proper design of the filters amounts to making $H(\omega)G(\omega+\pi) - G(\omega)H(\omega+\pi)$ a pure delay. Then proper implementation requires a choice of p , q , r , and s to make the total delay of the system zero. In the sequel we always take

$$p = (L_H - 2)/2, \quad q = (L_G - 2)/2 \quad (3.5)$$

for even length filters, and

$$p = (L_H - 1)/2, \quad q = (L_G - 3)/2 \quad (3.6)$$

for odd length filters, where L_H and L_G denote the lengths of the filters H and G . With this choice and for symmetric filters, the delay of the system becomes zero by taking $r = q+1$ and $s = p+1$, and the location of the information in the low-pass subband has the best correspondence with the original signal.

To make $H(\omega)G(\omega+\pi) - G(\omega)H(\omega+\pi)$ a pure delay, we can take, for example, a filter G that is the reverse of H with alternating coefficients multiplied by -1 , in the frequency domain denoted by

$$G(\omega) = H(-\omega+\pi)e^{j(-\omega+\pi)(L_H-1)}. \quad (3.7)$$

Note that G is causal, as required in our definition. This leads to

$$T(\omega) = [H(\omega)H(-\omega) - H(\omega + \pi)H(-\omega + \pi)e^{i\pi(L_H - 1)}] e^{i\omega(p+r+1-L_H)}. \quad (3.8)$$

A pure delay is achieved when $H(\omega)$ is designed such that

$$H(\omega)H(-\omega) + (-1)^{L_H}H(\omega + \pi)H(-\omega + \pi) = 1, \quad (3.9)$$

and we have obtained an orthogonal PR system. Note that equations (3.7) and (3.9) appeared in (2.36) and (2.26) in a slightly different form. Another possibility is to make a biorthogonal PR system, which is described in Appendix B.

3.3 Proper signal extensions for perfect reconstruction

In this section, we review the conventional methods of signal extension used in the filtering of finite length signals. These methods depend on the signal being of even length N . Filtering of finite length signals causes problems at the boundaries of the signal because the filter crosses the signal border. Four main methods of signal extension are used:

- zero padding;
- repeating of the boundary value;
- making the signal periodic (circular convolution);
- mirroring the values of the signal with respect to the boundary.

The criterion of whether a boundary extension allows PR can be stated as follows. *Suppose, the signal of length N is extended to infinity at both boundaries, filtered, and downsampled. If all the coefficients of both infinite length subbands can be determined from a subset of $N/2$ samples in each band then the extension enables PR.*

Somewhat more restrictively, we can say that only $N/2$ different samples may appear in each infinite length subband. This is impossible for the first two extension methods. The third, called periodic extension, and the fourth, symmetric extension, do allow PR. The periodic and symmetric extension methods are handled in more detail in the following subsections and are illustrated with diagrams which clearly demonstrate what perfect reconstruction relies on.

3.3.1 Periodic extension

The periodic extension method is most often used, because the method always allows PR for any type of filter. When filtering is implemented in the frequency domain with the FFT (Fast Fourier Transform), the periodic extension method is used implicitly.

Figure 3.3 Periodic extension (independent of filter properties).

In Figure 3.3 we demonstrate in a schematic diagram how periodic extension takes place. It shows the two-channel analysis/synthesis scheme given in Figure 3.2 after rotation over -90 degrees. On the left-hand side the low-pass channel is displayed. The transmission channel comprises also the coding and decoding. The four large blocks contain a pictorial description of the filtering. To demonstrate the periodic extension we use an orthogonal non-linear phase filter of length 8 fulfilling (3.2), (3.7), and (3.5), e.g., Daubechies' filter [Daub88].

In both analysis blocks, the top row shows the original signal of length $N = 8$ in the dark rectangle. Each little square denotes a sample the value of which is represented by a lower case letter. Outside the dark rectangle, the signal is extended periodically, which is visualized by using the same lower case letters when the same signal value appears. The $N/2 = 4$ rows in the middle represent the four filter positions. The filter coefficients are written in upper case letters. For each filter position, the inner product is calculated with the filter and the part of the extended signal in the top row directly above the filter. The values from the $N/2$ inner products are put into the bottom row in the dark rectangle. For example, the 'k' is the result of the inner

product of the filter H in its first filter position with the part of the signal denoted by the symbols 'f', 'g', ..., 'e'. The downsampling is illustrated by the empty squares. In the bottom row, it is also shown which values would have appeared outside the dark rectangle if the convolution would have been over more than $N/2$ filter positions. We observe that no new values appear, which means that perfect reconstruction is possible: all information in the subbands is contained in the samples in the dark rectangles. One can easily verify that this also holds for arbitrary other filters.

In the synthesis blocks, the subband signal is extended and upsampled as is demonstrated in the top row. All N filter positions are needed for the reconstruction (only 3 are displayed). For our choice of filters (3.7) holds and the delay of the total system is $p + r + 1 - L_H$. Thus, with (3.5) the choice $r = s = 4$ effects to a total delay of 0.

Although periodic extension is general applicable, it has a drawback for some applications, e.g. image coding. In general, there is a dissimilarity between the right-hand and left-hand side of the signal. This causes a sharp transition in the extended signal and generates large subband coefficients. These large values will obstruct the coding of the subbands as they require a lot of bits in order to be represented with enough accuracy. The symmetric extension method solves this problem. However, it can only be applied for symmetric filters.

3.3.2 Symmetric extension

Symmetric extension is only possible for symmetric filters and depends on the filter length being even or odd. First, we show the even length case in the diagram given in Figure 3.4a. For instance, Johnston's nearly perfect reconstruction filters [John80] belong to this class of even length symmetric filters. The symmetry of the filters is symbolized in the way the letters are repeated. The input signal is mirrored on all sides and this gives a smooth extension so no sharp transitions are introduced. The symmetry axis must lie halfway between two samples (even symmetry). Only for the synthesis high-pass filtering is antisymmetric extension required. The reason for this exception can easily be understood from the diagram. From the bottom rows of the synthesis filtering blocks in the diagram it is clear that another choice for r and s would cause a delay of the total system.

For odd length filters, the symmetric extension must be performed in a different way to obtain $N/2$ different samples. This is shown in Figure 3.4b for a biorthogonal wavelet filter pair with lengths 9 and 7 like the one appearing in [Anto91]. In the analysis, the symmetry axis is positioned on a sample point (odd symmetry). In the

Figure 3.4 Symmetric extension; (a) symmetric even length filters;
(b) symmetric odd length filters.

synthesis, the extension depends on the channel. For one channel, the left-hand side is extended with odd symmetry and the right-hand side with even symmetry. For the other channel the reverse holds. Thus, the synthesis extension is different from the analysis extension [Mart91]. It can easily be verified why even symmetric extension as used for even length filters fails here: the sample values of the subbands outside the dark rectangle would have no relationship with the ones inside. Therefore, they contain information about the signal and this information is lost since these samples are not transmitted.

3.4 Efficient signal extension for decomposition of arbitrary length signals

Whenever a signal of arbitrary length has to be decomposed into subbands up to a level d , a number of samples are added to the signal in order to make its length a multiple of 2^d . These extra samples cause the signal to be represented by a number of subband coefficients that is larger than the number of samples in the original signal.

We now show that such an increase in data is not necessary at all. All we have to do is to show how a signal of odd length N can be represented by a number of subband samples equal to N . We do this by implementing an efficient manner of signal extension which has three variants: a periodic extension for any type of filter, and two variants of symmetric extension for symmetric filters.

We use Figure 3.3 and Figure 3.4 given in the previous section. Now, however, our signal has length $N = 7$ and the sample indicated by 'h' does not belong to our signal. Thus, imagine the dark rectangle enclosing the 7 samples denoted by 'a' through 'g'. The new extension method consists of adding one specific sample 'h' and then the signal is extended as before, either periodic (Figure 3.3) or symmetric (Figure 3.4). The value of the sample 'h' in the extension is chosen such that one of the samples in the subbands becomes a fixed value. This fixed value is signal-independent and therefore contains no information about the signal. As both the transmitter and the receiver know this fixed value beforehand, it does not need to be transmitted and so the number of subband samples equals the length of the original signal. In other words, we have now enabled the tree-structured splitting shown in Figure 3.1b.

In our examples, the fixed sample is the rightmost sample of the high-pass subband, represented by a 'w' in the diagrams. The value of 'w' is the result of the filtering with the high-pass filter in the last of the four filter positions and forms a linear

equation with one unknown variable 'h'. We fix 'w' to 0 and 'h' is uniquely determined.

On the receiver side, no additional information is needed to reconstruct. As soon as the reconstruction procedure encounters two subbands with the same 'parent' of which the high-pass subband has shorter length, it knows that a 0 has to be appended on the right-hand side of it, before applying the signal extension in the normal way.

The exact value to which we force 'w' to be equal is not very important for PR. However, the choice will have an effect on the samples in the low-pass band. In general, the statistics of the low-pass band should not change because of our choice for 'w'. Thus, a constant signal should stay constant after adding the specific sample 'h'. It can easily be verified that then the value for 'w' has to be fixed to 0 because of the zero mean of the high-pass filter.

Table 3.1 Formula for the specific sample in the efficient signal extension method for odd length signals

Extension type / Filter length	Formula
Periodic / Even	$-\left(\sum_{k=0}^q s_{N-q-1+k} g_k + \sum_{k=0}^{q-1} s_k g_{q+2+k}\right) / g_{q+1}$
Periodic / Odd	$-\left(\sum_{k=0}^q s_{N-q-1+k} g_k + \sum_{k=0}^q s_k g_{q+2+k}\right) / g_{q+1}$
Symmetric / Even	$-\left(\sum_{k=0}^q s_{N-q-1-k} g_k + \sum_{k=0}^{q-2} s_k g_{q+3+k}\right) / (g_q + g_{q-1})$
Symmetric / Odd	$-2\left(\sum_{k=0}^q s_{N-q-1+k} g_k\right) / g_{q+1}$

In summary, we present in Table 3.1 the value of the specific sample in the extension for all possibilities of filter lengths and PR extensions. The signal of length N is denoted by s_n , and the high-pass filter by g_n . The value of q is chosen according to (3.5) and (3.6). Again, the sample to be fixed is chosen to be the last of the high-pass subband.

The implementation of the efficient signal extension method does not increase the computational complexity. Instead of the last inner product in the filtering (not

needed because it leads to the preset value), we have to calculate the specific sample. This requires the same number of multiplications and additions as for the calculation of an inner product.

3.5 Efficient signal extension for more dimensional filters and multi-channel analysis/synthesis schemes

Thus far, we have only discussed the two-channel splitting scheme for one-dimensional signals. Now, we discuss the implementation of the new signal extension method in the more dimensional and the multi-channel case.

For multidimensional signals, we distinguish between separable and non-separable filters. In the first case, the filtering is built up of the one-dimensional filtering of one-dimensional signals, and all extension methods given in the previous sections will work. In the case of non-separable filters, the normal periodic and symmetric extension work properly for even-sized signals. The efficient extension method for odd-sized signals becomes much more complicated. The calculation of the specific samples requires a lot of computational effort because they depend on each other. In two dimensions with a signal of size $N \times N$ (N odd), for example, the calculation requires the solving of a linear system

$$\mathbf{V}\mathbf{z} = \mathbf{W}\mathbf{x}, \quad (3.10)$$

where $\mathbf{V}$ and $\mathbf{W}$ are square matrices and $\mathbf{z}$ and $\mathbf{x}$ are vectors of size $2N + 1$. The vector $\mathbf{z}$ consists of the specific samples to be calculated. Both $\mathbf{V}$ and $\mathbf{W}$ are almost zero outside a diagonal with width equal to the size of the synthesis filter. In most applications, the increase in data will be preferred above the extra computational complexity of the new extension method.

For M -channel ($M > 2$) analysis/synthesis schemes we have the same requirements for PR. When the length of a signal is a multiple of M the normal periodic and symmetric extension will do. However, if we want to split up a signal of length $n_1 M + n_2$, ($0 < n_2 < M$) without increase in data, we can do the efficient extension analogous to the two-channel case. The n_2 extra samples are found by solving a set of n_2 linear equations. The extra computational complexity will, in general, be less important than the gain in efficiency achieved by the new extension method.

3.6 Symmetric-periodic signal extension

The efficient signal extension for the filtering of odd length signals is also beneficial in one case of filtering even length signals. In some applications, it might be a drawback that the normal symmetric extension for odd length filters (Figure 3.4b) has a total periodicity of $2N - 2$ instead of $2N$, where N is the signal length. For even length filters the periodicity is always $2N$. This periodicity is a nice feature when applying filtering in the transform domain with the FFT.

Using the same method as described in the previous section, we can design an alternative to the symmetric extension for odd length filters shown in Figure 3.4b. This extension, which we call symmetric-periodic extension, is shown in Figure 3.5 again with the 9-7 length filter pair. The extra sample is denoted by 'z' in the dia-

Figure 3.5 Symmetric-periodic extension for symmetric odd length filters.

gram. It makes the periodicity of the extended signal equal to twice the signal length. With a proper choice of the value of 'z', analogous to the previous section, the sam-

ple 'x' on the left-hand side of the extension of the high-pass subband becomes zero, thus reducing the number of different signal-dependent samples in the high-pass subband from $N/2 + 1$ to $N/2$.

The use of an extra sample to make an extension both symmetric and periodic with period $2N$ was suggested earlier in [Blan88] for Kalman filtering. However, the value of the extra sample 'z' was determined with a linear or quadratic interpolation. When applying this in an analysis/synthesis system like our two-channel splitting scheme, the PR property is lost.

3.7 Conclusions

In this chapter, we have illustrated the way signal extension works for subband and wavelet decomposition in a two channel analysis/synthesis scheme with perfect reconstruction. The visualization clearly demonstrates what perfect reconstruction is based on, concerning signal extension. A new efficient signal extension method has been presented for creating a subband/wavelet decomposition up to arbitrary level for arbitrary length signals without an increase in the total number of samples. In applications like image archiving, a substantial data reduction in comparison to existing subband/wavelet decompositions is not uncommon.

The method is also applicable to the multi-channel analysis/synthesis systems and for more dimensional signals, although in the latter case, the implementation is impractical for non-separable filters.

Related to the efficient signal extension is the symmetric-periodic extension for filtering even length signals with symmetric odd length filters. It is an alternative to the normal symmetric signal extension and has the additional feature of a periodicity that is twice the signal length. It is useful in applications where the filtering is performed by means of FFT.

Chapter 4

Image and Video Coding Schemes

The introduction of wavelets and the implementation of the wavelet filters discussed in Chapter 2 and Chapter 3 are followed in this chapter by the description of complete coding schemes. These schemes are used in the experimental part of the thesis. Many coding schemes are possible and a lot of them are treated in the literature. We choose an often used configuration for both our image and our video coding scheme. It is not our primary intention to design the best scheme, we mainly wish to concentrate on the effect of the filtering part in the schemes.

Section 4.1 describes the scheme used for image coding, and Section 4.2 handles the schemes used for video coding. We end with conclusions in Section 4.3.

4.1 The image coding scheme

In this section we describe the scheme used for image coding in this thesis. As explained in Chapter 2, for the coding schemes, there is no difference in subband and wavelet coding, except for the characteristics of the filters. Each time we mention the discrete wavelet transform, a subband decomposition can equally be understood and vice versa. The difference between the filter types is discussed in Chapter 5.

In the preceding chapters all theory and examples considered one-dimensional signals only. Images, however, are two-dimensional. The implementation of the DWT in the experiments makes use of a separate row and column processing. This means that first the rows and then the columns are filtered (or vice versa, which is

equivalent), yielding 4 subbands after a one-level splitting. One of the subbands in the decomposition contains the low-pass information in both the horizontal and the vertical direction. In this thesis we refer to this subband as the low-pass subband. All other subbands are called high-pass subbands. Another type of two-dimensional wavelet transform is described in Chapter 6.

The image coding scheme is shown in Figure 4.1. It assumes that the grey-value image to be coded is already present in a digital format (normally 8 bit per pixel). The image is decomposed in the DWT block into several subbands by iteratively splitting the low-pass subband, as in Figure 2.7. Note that in subband coding theory (e.g. [West89]) and in the theory of wave packets ([Wick92]) also the high-pass subbands are allowed to be split. In this thesis, however, we restrict ourselves to the pure wavelet decomposition.

In general, the depth of the splitting tree ranges from 3 to 5. The reason is that if a low-pass subband contains frequency components that are all equally important in subjective visual quality, then further decomposition is not efficient. In the past, there was another reason for stopping the decomposition at a certain level. In such cases the width and height of the images were not a power of 2, thus a complete decomposition was impossible. However, with the efficient signal extension method given in Chapter 3, arbitrary depths can be achieved. When the decomposition is not complete, the transform is in fact a truncated DWT, but we will still call it a DWT.

Figure 4.1 Image coding scheme as used in the experiments. The transmitted data consists of the variances and the quantizer output.

The division into subbands is assumed to correspond to important and less important information that is used by the human visual system (HVS). This means that some of the subbands have to be transmitted more accurately than others and a bit allocation algorithm is used to distribute the available bit rate to adapt to the HVS.

The subbands have to be transmitted over the channel with the bit rate specified by the bit allocation. In order to achieve this rate, the subbands are quantized, which causes a loss in information. The transmitted data consists of the quantized subbands as well as the variances of the original subbands, which in our implementation are used to guide the bit allocation. The quantized subband coefficients are either entropy coded or arithmetic coded. Arithmetic coding is able to reach the entropy of the source if it is made adaptive.

At the decoder, the bit allocation is repeated with the variance information and therefore the subband coefficients can be reconstructed. Of course, the reconstruction is not perfect as the quantization causes a data reduction. The reconstructed subbands are transformed by the inverse of the DWT into a distorted version of the original signal.

In the remainder of this section we describe in detail the bit allocation algorithm and the quantization as used in our experiments.

Bit allocation

In our experiments, we used the bit allocation that originates from [West89]. The bit allocation problem consists of finding the optimal distribution of the available bit rate among the subbands. If a signal of size N is decomposed into M subbands, where subband k has size N_k , then the total bit rate R (in bit per pixel) is

$$R = \sum_{k=0}^{M-1} \frac{N_k}{N} R_k, \quad (4.1)$$

where the allocated bit rates for the subbands are denoted by R_k . Each of these allocated bit rates invokes a quantization, which causes a distortion D_k per subband. We assume that the total distortion of the signal is the sum of the distortions of the subbands,

$$D = \sum_{k=0}^{M-1} D_k. \quad (4.2)$$

For each subband k a set of Q_k quantizers is available, each with its own rate and distortion. These quantizers are located both at the encoder and the decoder side. The bit allocation consists of assigning to each subband one of its quantizers, in a way such that the total distortion is minimized. Or, if a certain distortion is required, then the bit allocation should find the minimal rate to transmit the signal at the given distortion. The latter case is not examined in this thesis, as in our applications the rate is specified. The number of possible bit allocations is

$$\prod_{k=0}^{M-1} Q_k, \quad (4.3)$$

which can be rather large. All these bit allocations can be represented by dots in the rate-distortion plane.

The algorithm of [West89] consists of finding the optimal distortion, given a target rate, by walking along the lower convex hull of the points until the target rate is achieved (see Figure 4.2). The algorithm is computationally attractive, since only a

Figure 4.2 Rate-distortion plane with dots that represent a possible rate-distortion combination. The line is the lower convex hull connecting points with optimal rate-distortion combinations.

few dots have to be evaluated. For details of the algorithm we refer to [West89]. Another advantage of this bit allocation algorithm is that the quantization is left completely free. The only requirement is that the rate and distortion of each quantizer are known. Further, the algorithm can easily be adjusted to give a better adaptation to the HVS. If one subband has more importance than in our definition, then only its distortion measure has to be weighted.

The assumption that the total distortion equals the sum of the distortions of the subbands can be realized by taking the mean square error (MSE) distortion measure and an orthogonal filter pair. In the case of non-orthogonal filters, (4.2) is closely approximated if the energy of the filters is close to 0.5 (within ca. 1%), that is,

$$\sum_{n=0}^{L-1} h_n^2 \approx 0.5, \quad (4.4)$$

for a filter h_n with length L . If the energy of the filter deviates strongly from 0.5, extra weights have to be introduced, as explained in [Wood92].

The MSE distortion measure does not completely correspond to the working of the HVS, but in the absence of a better distortion measure it is a good choice and it keeps the calculations simple. With this distortion measure, the bit allocation is guided by the variances of the subbands [West89].

Quantization

The bit allocation requires for every subband k a set of Q_k quantizers, each with its own rate and distortion. In our experiments we used scalar quantization with uniform threshold quantizers. The high-pass subbands were PCM (Pulse Code Modulation) coded [Jaya84]. The low-pass subband is a low-pass version of the original signal. It has non-zero mean and the correlation between two neighbouring pixels is high. Therefore, the coding method was DPCM (Differential PCM). The image was scanned line by line from top to bottom and each pixel was predicted using three neighbouring pixels that had already been quantized (Figure 4.3). We used three fixed prediction coefficients which were determined by solving the Yule-Walker equations [Jaya84].

Figure 4.3 DPCM prediction. The coefficient x is predicted by its quantized neighbours a , b , and c .

The scalar quantizers are designed to match the statistical properties of the sub-band coefficients. Quite some research has been dedicated to the determination of the probability density function (pdf) that fits the transform coefficients [Rein83][Egge86][West89][Barn92]. The generalized Gaussian distribution

$$p(x) = \frac{\alpha K(\alpha)}{\sigma \Gamma(\frac{1}{\alpha})} e^{-\left[K(\alpha) \left| \frac{x-\mu}{\sigma} \right| \right]^\alpha}, \quad (4.5)$$

with shape parameter α , scale parameter σ , and location parameter μ has turned out to be the best pdf model. The function $\Gamma(\cdot)$ is the gamma function and the function $K(\alpha)$ is defined as

$$K(\alpha) = \sqrt[\alpha]{\Gamma(\frac{3}{\alpha})/\Gamma(\frac{1}{\alpha})}. \quad (4.6)$$

If α tends to 0, the shape of the pdf becomes a delta function. For $\alpha = 1$ the Laplacian distribution appears, and for $\alpha = 2$ the Gaussian distribution. If α approaches infinity then the pdf becomes a uniform distribution (see Figure 4.4). To find the pdf

Figure 4.4 The generalized Gaussian distribution. The pdf is plotted for α between 0.5 and 6.0 whereas the scale parameter σ and the location parameter μ are set to 1 and 0, respectively.

of a certain subband, the quantization procedure starts with the rescaling of the subband coefficients to unit variance. Then the shape parameter is estimated with $\sigma = 1$ and $\mu = 0$ fixed. The choice of zero mean is justified by the zero mean of the high-pass filter.

4.2 Some video coding schemes

Image compression schemes make use of the spatial correlation between the pixels. Compression techniques for video can also use the correlation in the temporal direction. In some of the literature, the third dimension is handled in the same way as the two spatial dimensions. This means that a group of consecutive frames in a video sequence is coded using a 3-D subband decomposition [Karl87].

This has three disadvantages. The first is that the coding of frames causes a delay, which might be a problem in real-time applications like videotelephony. Another disadvantage is that the filtering in the temporal direction causes features of one frame to appear in another frame, because of the motion in the scene. This produces a degradation of the subjective visual quality of the video. This is the reason that in [Ohm92] the subband coding in the temporal domain is performed after the motion compensation. The third disadvantage is that when filters of a length greater than 2 are used, the extension at the temporal boundaries is never satisfactory. Periodic extension often produces sharp artificial (temporal) edges to be coded and symmetrical extension has the same problem in the case of motion. A solution is to filter over all frames, thus without grouping the frames into independent parts of consecutive frames. This reduces the number of boundaries to 1 (the first frame), but rules out the use of a refresh mode or a fast forward mode as in MPEG [MPEG-1]. Therefore, the Haar filter of length 2 is most often used for the filtering in the temporal direction. The length of this filter is equal to the downsampling rate and thus no temporal signal extension is needed. One might question if this form of subband decomposition in the time domain deserves the name subband coding, as it is almost equal to coding frame differences.

For these reasons, there is no decomposition in the temporal direction in most of the literature. The basic principle of temporal coding is that a frame is predicted by means of reconstructed frames. Because of moving objects in the scene, the problem is to compensate as well as possible for the motion.

The most simple motion estimation is that the predicted frame equals the reconstructed previous frame. Another prediction method is block-matching, for which the

images are divided into rectangular blocks. For each block in an image, the algorithm searches for the best matching block in the reconstructed previous image. Other prediction methods try to find a vector for every pixel, where the total vector field should fulfil some smoothness criteria [Drie92]. Such a motion vector field has to be reproduced by the decoder. These methods have a high computational complexity.

In video coding systems using a subband decomposition there are several options for the positions of the blocks in the coding scheme. The main choice that has to be made is the order in which the time and spatial correlation are removed. Coding standards like H.261 [CCIT90], and MPEG [MPEG-1][MPEG-2] are based on the discrete cosine transform (DCT). These schemes first remove the temporal correlation with frame prediction, after which the spatial correlation in the prediction error is removed. If this concept is translated into a scheme with a DWT we get the scheme shown in Figure 4.5. We have left out the bit allocation block to improve the visibility of the scheme structure.

Figure 4.5 Video coding scheme I; temporal coding followed by spatial coding.

Another type of scheme arises when the spatial correlation is removed first. The corresponding scheme is shown in Figure 4.6. The scheme has the ability to code the images in the video sequence by using the multiresolution principle. This implies that a decoder may work with less memory and computation power to generate a low res-

Figure 4.6 Video coding scheme II; spatial coding followed by temporal coding.

olution version of the original image by using only a subset of the total bit stream. Coding methods with this property are also called compatible or hierarchical coding methods. Another advantage of multiresolution coding is that the information of a full resolution signal (like HDTV) can be protected against transmission errors in a variable way. Thus, in the case of transmission over a channel with a variable bit rate, the low resolution can be sent in the high priority packets. Then packet losses will yield a graceful degradation of the image quality.

In scheme II, the temporal correlation in the subbands is used to encode the subband coefficients more efficiently. This can be performed by estimating the motion per subband and predicting the subbands with the reconstructed previous subbands (e.g. [Zhan92]). Theoretically, this in-band motion estimation is not able to achieve the same motion estimation accuracy as the full resolution motion estimation, because the motion information is spread over all subbands. In [Vial92] is shown that the in-band motion estimation is almost optimal if also the adjacent subbands are used in the estimation of the motion in a subband. In-band motion estimation has the advantage of low computational complexity when compared to full image size motion estimation, and it enables the hierarchical coding of the images. The suboptimal motion vectors will often reduce the bit rate needed for transmission of the sub-

band coefficients, even if the optimal motion field is not found. A performance comparison in [Ghar91] of the video coding schemes I and II showed that scheme II is slightly better suited to high bit rates and scheme I to low bit rates.

In Figure 4.7 a third kind of video coding scheme is shown that also has a hierarchical approach. It has the property that the quantization error of the already transmitted subbands are requantized when a high resolution image is used. Thus, for every resolution, the only quantization error is the error of the subbands with the highest frequencies. This scheme is extensively studied in [Bosv94]. It has a higher degree of

Figure 4.7 Video coding scheme III; hierarchical coding with requantization of the subbands.

complexity than scheme II and the requantization is theoretically suboptimal. However, the advantage of scheme III over scheme II is that, for a low resolution image, the bit rate can be chosen in proportion to the image size without affecting the quality of the image at any higher resolution. For scheme II, the bit rate used to code the low-pass subband restricts the overall quality that can be achieved for the full resolution image, as the information lost by quantization of the low-pass subband coefficients can never be recovered.

Again we mention that we do not aim to design the best video coding scheme. We only wish to have a good scheme with which we can perform our experiments on the wavelet decomposition. We choose for scheme II, because it has an inherent multiresolution structure and has a lower complexity than scheme III. The remaining parts of the coding scheme, like the quantization, are performed in the same manner as described for image coding in Section 4.1.

4.3 Conclusions

In this chapter we have introduced the image and video coding schemes that were used in the experiments reported in the following chapters. The image coding scheme has been described in detail, including the bit allocation algorithm and the quantization. Three important types of video coding schemes have been discussed. One of these video coding schemes was selected for the experiments described in the following chapters.

Chapter 5

Comparison of Wavelet Filters and Subband Coding Filters

In this chapter we compare wavelet filters with filters that are designed according to the subband coding theory. We restrict ourselves to finite impulse response filters, since these are most important for image coding. First, we give in Section 5.1 an overview of several properties which, ideally, a filter should have. As it is impossible for a filter to share all these properties, a filter design process consists of minimizing the deviation of a filter to a subset of the properties. The subband coding filter design and the wavelet filter design use different criteria for this minimization.

Section 5.2 contains an example of the subband coding filter design. We take the set of Johnston filters, which are popular in many applications. Some of these filters demonstrate a low objective coding performance for image coding applications. We give an explanation for this effect and design modified versions of those filters. Examples of wavelet filter design are shown in Section 5.3. An overview of the differences between the subband coding and wavelet filter design is presented in Section 5.4.

Next, the influence of the filter characteristics on the coding performance is evaluated in Section 5.5. We compare several filters with respect to their objective and subjective coding performance for image and video coding. Finally, we summarize the conclusions of this chapter in Section 5.6.

5.1 Criteria for filter design

In the subband coding theory the aim of a subband decomposition is to strictly divide the frequency band into non-intersecting bands. For a two-band splitting scheme, the magnitude of the ideal low-pass filter is shown in Figure 5.1a. Such a block-shaped magnitude response implies an infinite length filter, which is impractical. There are several methods to make the impulse response a finite length, for instance truncation or windowing of the impulse response. After such an operation the magnitude response will be an approximation of the ideal block shape (Figure 5.1b).

Figure 5.1 (a) The ideal magnitude response; (b) Approximation of the ideal magnitude response, obtained by symmetrically truncating the infinite impulse response to 41 coefficients.

Several criteria exist which evaluate the quality of the approximation. Almost all depend on the 4 parameters that are indicated in Figure 5.1b:

- δ_p the peak passband ripple,
- δ_s the peak stopband ripple,
- ω_p the passband edge, and
- ω_s the stopband edge.

A low peak passband ripple guarantees that the frequencies in the filtered signal are almost equally weighted. A low stopband ripple prevents high frequencies from appearing in the filtered signal. The high frequencies would show up as low frequencies after downsampling, which effect is called aliasing. Large aliasing causes severe degradation of the subjective image quality of the low-pass filtered signal. Also in subband coding systems with aliasing cancellation, the aliasing part should not be too large: the quantization of the subbands would partly destroy the cancellation. The larger the aliasing is, the larger the reconstruction error will be. The passband and stopband edge must be as close to $\pi/2$ as possible to have a good coverage of the passband and stopband region.

The filter characteristics are often measured by three other criteria, which are functions of the four parameters. The passband characteristic is usually measured by $A_p = -20 \log(1 - \delta_p)$, which is the peak passband ripple in dB. The minimum stopband attenuation is defined by $A_s = -20 \log(\delta_s)$, and is an alternative to the peak stopband ripple. The transition bandwidth is defined as the difference between the stopband and passband edge, $\Delta \omega = \omega_s - \omega_p$.

The question of whether a filter is good or not depends on the importance of the various criteria. Sometimes the stopband attenuation is most important, sometimes the maximal ripple has to be minimized.

In contrast, the filter design according to the wavelet theory does not try to approximate the ideal block shape. The wavelet filter design highly emphasizes flatness near $\omega = \pi$, which depends on the number of zeros of the low-pass filter at $\omega = \pi$. As explained in Section 2.5, this number of zeros is related to the degree of regularity. In Figure 5.2 we show a typical shape of the magnitude response for wavelet filters.

For ease of notation we define names for the two types of filters. Filters that are designed according to the subband coding theory are called SBT filters in the sequel, whereas the filters designed for regularity are called WT filters.

Figure 5.2 Typical example of a magnitude response of a wavelet filter.

Finally, there are two other filter characteristics that we wish to mention. They are important for both filter types. The first is the filter coefficient sum, which should be close or equal to 1,

$$\sum_n h_n = 1. \quad (5.1)$$

In the design process of wavelet filters, equality in (5.1) is guaranteed. For other design methods the coefficient sum must be close to 1, otherwise the transfer function of the input-output system is biased.

The second criterion is the coefficient energy, which should be 0.5,

$$\sum_n h_n^2 = 0.5. \quad (5.2)$$

If this criterion is not fulfilled, the subbands contain different fractions of the signal energy. Consequently, quantization errors in one subband will be more significant than errors in other bands.

With this overview of filter properties we close this section. In the next two sections we show two different examples of filter design. For a historical overview of filter design and a variety of techniques we refer to [Vaid93].

5.2 Subband coding filter design: Johnston filters

As an example of designing filters according to the subband coding theory we take Johnston's design procedure [John80]. These SBT filters are rather popular in several application areas among which is image coding. The most important constraint in the design process of the filters is the transition bandwidth.

We use the same notation as that in Section 3.2. The output of a two-band splitting scheme is related to the input by the relation of (3.4), which we repeat here for the sake of convenience,

$$T(\omega) = (H(\omega)G(\omega + \pi) - G(\omega)H(\omega + \pi)) e^{i\omega(p+r)}. \quad (5.3)$$

Here $T(\omega)$ is the transfer function, p and r are filter offsets, and $H(\omega)$ and $G(\omega)$ are the low-pass and high-pass filter, respectively. However, as in [Croi76] Johnston chooses another relation between the high-pass and low-pass filter than that shown in (3.7). This choice prohibits perfect reconstruction for filters with length greater than 2. The high-pass filter becomes the mirror of the low-pass filter with respect to the quarter of the sampling frequency, i.e.,

$$G(\omega) = H(\omega + \pi), \quad (5.4)$$

and for this reason the filters are called quadrature mirror filters (QMF). Then the transfer function becomes

$$T(\omega) = [H(\omega)H(\omega) - H(\omega + \pi)H(\omega + \pi)] e^{i\omega(p+r)}, \quad (5.5)$$

which for an even-length symmetric filter reduces to

$$T(\omega) = [|H(\omega)|^2 + H(\omega + \pi)^2] e^{i\omega(1-L+p+r)}, \quad (5.6)$$

where L is the length of the filter h_n . All filters in this QMF bank thus depend on one filter. The phase distortion has been removed by the symmetry property of the filter and the phase can be made zero by proper choice of the filter offsets p and r .

The only type of distortion that is left is the amplitude distortion, which has to be made minimal. Zero amplitude distortion is only possible for $L = 2$. Johnston imposes a certain value for the transition bandwidth. Then the filters coefficients are optimized by minimizing the error criterion

$$C = 2 \int_0^{\pi/2} (T(\omega) - 1) d\omega + \alpha \int_{\omega_s}^{\pi} |H^2(\omega)| d\omega, \quad (5.7)$$

where the parameter α weights the stopband attenuation. Equation (5.7) is implemented in discrete form by applying a 512 point FFT and replacing the integrals by summation.

Johnston designed filters of seven different lengths, but we examine only those filters which have a length that does not exceed 32. The larger filters are not practical for image coding applications. Johnston fixed five target transition bandwidths for his design algorithm. His combinations of target bandwidths and filter lengths are shown in Table 5.1, together with the choice for α .

Table 5.1 The length, transition bandwidth, and stopband attenuation parameter of the Johnston filters.

	$\Delta \omega$				
	0.14 (A)	0.10 (B)	0.0625 (C)	0.43 (D)	0.23 (E)
L	8	$\alpha = 1$	-	-	-
	12	$\alpha = 1$	$\alpha = 1$	-	-
	16	$\alpha = 1$	$\alpha = 1$	-	-
	24	-	$\alpha = 1$	$\alpha = 1$	-
	32	-	-	$\alpha = 2$	$\alpha = 2$

The five target bandwidths imply five classes of filters, which are denoted by capitals as in Table 5.1. In this thesis we refer to a Johnston filter by the prefix "joh" followed by the filter length and an upper case letter denoting the target transition bandwidth. For instance, joh16C is Johnston's filter of length 16, designed to have transition bandwidth 0.0625. All filter characteristics are tabulated in Appendix E.

Modification of some of the Johnston filters

The criteria used by Johnston do not produce filters with a coefficient sum equal to 1 and coefficient energy equal to 0.5. In Table 5.2 we show the coefficient sum and energy for all of the Johnston filters.

The coefficient energy is close enough to 0.5 to avoid any problems. However, the filters joh8A, joh12A, joh12B, joh16C, joh32E have a coefficient sum that deviates 0.1% or more from 1. This yields a transfer function that is biased. In other words,

Table 5.2 Filter coefficient sum and energy of Johnston filters

Johnston filter	Filter coefficient sum, $H(0)$	Transfer function at $\omega = 0$, $T(0)$	Filter coefficient energy
joh8A	0.99629	0.99259	0.49996
joh12A	1.00100	1.00200	0.50002
joh12B	1.00186	1.00372	0.49998
joh16A	0.99963	0.99926	0.50003
joh16B	0.99940	0.99879	0.50003
joh16C	0.99783	0.99566	0.49992
joh24B	0.99982	0.99965	0.50003
joh24C	0.99982	0.99965	0.50003
joh24D	0.99923	0.99845	0.49997
joh32C	0.99992	0.99985	0.50003
joh32D	1.00003	1.00006	0.50002
joh32E	0.99844	0.99687	0.49980

the average value of the reconstructed signal will not be equal to the average of the original signal.

In [West89][Bosv89] it had already been indicated that precisely those five filters cause a larger filter bank error variance, and they are therefore not recommended for image coding applications. We show below how the filters can be made unbiased with hardly any violation of the criteria Johnston imposed.

First, we show in Figure 5.3 what happens when an image is split into four bands and reconstructed without quantization. For this figure we took the joh16C filter and a 128x128 window of the Claire sequence. In Figure 5.3c-d the floating point error and integer error are shown, where the latter is the truncation of the first. Both images are contrast-stretched to reveal the features. In the floating point error we recognize four shifted copies of the original signal. Each copy is shifted 8 pixels in one of the four directions along the axes. The impulse response of the transfer function is shown in Figure 5.4. Here we see that the two largest peaks are at 16 pixels distance, explaining the shifted copies that appear in Figure 5.3c. The truncated error image shows that (without any quantization) roughly half of the pixels have changed in value because of the bias in the transfer function. The error pixels have value 1.

(a)

(b)

(c)

(d)

Figure 5.3 The filter bank error of the joh16C filter. (a) Original image; (b) Subband decomposition; (c) Floating point error of the reconstruction; (d) The reconstruction error truncated to integer.

Figure 5.4 The impulse response of the error in the transfer function,

$$t_n - \delta_0.$$

To have a good chance of there being no error after rounding of the reconstruction coefficients to the nearest integer, we must have

$$T(0) \max_n (|s_n|) < 0.5, \quad (5.8)$$

where s_n denotes the signal. That is, with the maximum equal to 255,

$$0.99804 < T(0) < 1.00196. \quad (5.9)$$

Roughly, also the filter coefficient sum $H(0)$ should be between these two numbers for the class of even length QMFs. Note that (5.9) only holds for a decomposition depth of 1. If the decomposition has greater depth, for instance 3 for a 10 band splitting, there is a stronger requirement for $T(0)$.

The filter coefficient sum can be made 1 by rescaling the filter. However, this induces completely different filter characteristics, and therefore, another method must be found. We choose for the option to change one of the $L/2$ different filter coefficients. The method is demonstrated for the joh32E filter.

In Figure 5.5a the magnitude response of the filter joh32E is shown. We can clearly see that the sum of the filter coefficients is not 1. In that case, the function should go through the point (0,1).

Because of the symmetry of the filter h_n we can write its Fourier transform as a sum of cosines,

$$\begin{aligned}
H(\omega) &= \sum_{n=0}^{L-1} h_n e^{-i\omega n} \\
&= \sum_{n=0}^{M-1} h_n (e^{-i\omega n} + e^{-i\omega (L-1-n)}) \\
&= \sum_{n=0}^{M-1} h_n e^{-i\omega (M-1/2)} (e^{i\omega (M-1/2-n)} + e^{-i\omega (M-1/2-n)}) \\
&= \sum_{n=0}^{M-1} 2h_n e^{-i\omega (M-1/2)} \cos((M - \frac{1}{2} - n)\omega),
\end{aligned} \tag{5.10}$$

where $M = L/2$.

The idea is to add an extra cosine to the definition of joh32E such that the deviation at $\omega = 0$ is cancelled. We can choose to match the curve near $\omega = 0$ as well as possible, but we have to keep in mind that the peak passband ripple should not also be enlarged. Therefore, we add a cosine $\cos((M - 1/2 - k)\omega)$ with as many oscillations between 0 and $\pi/2$ as $H(\omega)$. For the filter joh32E there are 3.5 periods from $\omega = 0$ to $\omega = 1.4$ and k is calculated from

$$(M - \frac{1}{2} - k)/2\pi \approx 3.5/1.4, \tag{5.11}$$

thus, $k = 0$. Thus, the cosine that has to be added has a frequency corresponding to the cosine of the first (and last) filter coefficient h_0 . In Figure 5.5 we show the addition process. The new filter is denoted by repeating the letter denoting the class. Thus, the modified filter joh32EE is the same as the filter joh32E, except for the changed first and last coefficient.

The new impulse response of the transfer function is shown in Figure 5.5c and the 7 filter criteria of the new filter are compared to the old filter in Table 5.3. The passband characteristics are improved, since we lowered the highest ripple. The stopband characteristics have become slightly worse, but we will see in Section 5.5.2 that the objective coding performance has increased. The other four filters joh8A, joh12A, joh12B, and joh16C are modified in the same way. The new filter coefficients and characteristics are tabulated in Appendix E.

(a)

(b)

(c)

Figure 5.5 Improvement of the magnitude response of one of Johnston's filters. (a) The magnitude response for the filter joh32E; (b) The cosine that is added; (c) Resulting magnitude response (joh32EE).

Table 5.3 Filter characteristics of joh32E and joh32EE

Filter	δ_p	δ_s	A_p (dB)	A_s (dB)	ω_p	ω_s	$\Delta \omega$
joh32E	0.0118	0.0577	0.103	24.78	1.435	1.752	0.317
joh32EE	0.0107	0.0591	0.093	24.57	1.432	1.751	0.319

5.3 Wavelet filter design: Daubechies filters and biorthogonal filter pairs

The design of WT filters is based on regularity constraints. From Section 2.5 on regularity we know that it is related to the number of zero moments of the high-pass filter and to the number of zeros at $\omega = \pi$ of the low-pass filter. In Section 5.3.1 we show how we can derive the filters introduced by Daubechies [Daub88] in an easy way. For this, we use the matrix description of the filtering [Pres92]. In Section 5.3.2 we show with a different method how biorthogonal filters can be designed [Cohe92].

5.3.1 Design of the Daubechies filters

The design procedure of Daubechies filters can be demonstrated in many ways. Here, we choose for the method of [Pres92]. The decomposition process is written as the matrix multiplication $\mathbf{v}^{j+1} = \mathbf{M}_j \mathbf{v}^j$ of Section 2.4, and we use the fact that

$$\mathbf{M}_j^T \mathbf{M}_j = \mathbf{I}, \quad (5.12)$$

where T means transpose and $\mathbf{I}$ is the identity matrix. If we take a filter h_n with length 4 we get the matrix of (2.51), which we repeat for convenience,

$$\mathbf{M}_j = \sqrt{2} \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & & & & \\ & h_0 & h_1 & h_2 & h_3 & & & \\ & & h_0 & h_1 & h_2 & h_3 & & \\ h_2 & h_3 & & & & h_0 & h_1 & \\ h_3 & -h_2 & h_1 & -h_0 & & & & \\ & h_3 & -h_2 & h_1 & -h_0 & & & \\ & & h_3 & -h_2 & h_1 & -h_0 & & \\ h_1 & -h_0 & & & & & h_3 & -h_2 \end{bmatrix}. \quad (5.13)$$

Equation (5.12) requires

$$\begin{aligned} h_0^2 + h_1^2 + h_2^2 + h_3^2 &= \frac{1}{2}, \\ h_0 h_2 + h_1 h_3 &= 0. \end{aligned} \quad (5.14)$$

These two equations do not give a unique solution for the filter h_n . But if we force the high-pass filter to have two zero moments,

$$\begin{aligned} h_3 - h_2 + h_1 - h_0 &= 0, \\ 0h_3 - 1h_2 + 2h_1 - 3h_0 &= 0, \end{aligned} \quad (5.15)$$

then (5.14) and (5.15) yield a unique solution (except for an additional reversed solution):

$$\begin{aligned} h_0 &= (1 + \sqrt{3})/8, & h_1 &= (3 + \sqrt{3})/8, \\ h_2 &= (3 - \sqrt{3})/8, & h_3 &= (1 - \sqrt{3})/8. \end{aligned} \quad (5.16)$$

All the Daubechies filters can be derived using this procedure. For a filter length of $L = 2N$, there are N equations for zero moments needed to obtain a unique solution. In general, the solution can only be obtained numerically. In the sequel we will refer to these filters by the prefix “dau” followed by the filter length, thus dau4 for the filter of (5.16).

5.3.2 Design of a biorthogonal filter pair

In this section the design process of a linear-phase biorthogonal wavelet filter pair with odd filter lengths is explained [Coh92]. The design process we describe here imposes a fixed number of zeros at $\omega = \pi$. This design criterion produces maximally flat filters. Other criteria also exist, but will not be mentioned here. We refer to [Vett92] [Coh92] [Vaid93] for other biorthogonal wavelet filter design methods.

As is explained in Appendix B, linear phase is an important property in the application of wavelet theory to image coding. In the same appendix we explain why linear phase and orthogonality cannot be combined for two-band splitting schemes. A linear-phase filter is either symmetric or antisymmetric.

The design process is described using the filter bank description. With the formulas of Section 3.2 and Appendix B it can easily be derived that the modulus of the transfer function equals

$$|T(\omega)| = |H(\omega)\tilde{H}(\omega) - H(\omega + \pi)\tilde{H}(\omega + \pi)| = 1, \quad (5.17)$$

where $H(\omega)$ and $\tilde{H}(\omega)$ are the low-pass filters in the analysis and synthesis bank, respectively. From this equation we design our filter pair. We use some imposed properties of the filters to find a frequency description for them. These descriptions are substituted in (5.17). We start with the symmetry property, combined with the odd length of the filters. In the second place we use the number of zero moments of the filters at half the sampling frequency.

Firstly, symmetric odd length filters can be written as polynomials in $\cos \omega$,

$$\begin{aligned}
 H(\omega) &= \sum_{n=0}^{L_H-1} h_n e^{-i\omega n} \\
 &= h_{M_H} e^{-i\omega M_H} + \sum_{n=0}^{M_H-1} h_n (e^{-i\omega n} + e^{-i\omega (L_H-1-n)}) \\
 &= e^{-i\omega M_H} \left(h_{M_H} + \sum_{n=0}^{M_H-1} h_n (e^{-i\omega (n-M_H)} + e^{-i\omega (M_H-n)}) \right) \\
 &= e^{-i\omega M_H} \left(h_{M_H} + 2 \sum_{n=0}^{M_H-1} h_n \cos((M_H-n)\omega) \right),
 \end{aligned} \tag{5.18}$$

with $M_H = (L_H - 1)/2$. Secondly, if $H(\omega)$ has l zero moments then we know that it can also be written as

$$H(\omega) = (1 + e^{-i\omega})^l Q_0(\omega) = \left(\cos\left(\frac{\omega}{2}\right)\right)^l Q_1(\omega), \tag{5.19}$$

for certain functions $Q_0(\omega)$ and $Q_1(\omega)$. But the only way that (5.19) becomes a polynomial in $\cos \omega$ is when l is even. In other words, a symmetric odd-length filter can only have an even number of zeros at $\omega = \pi$. Therefore, if $H(\omega)$ and $\tilde{H}(\omega)$ have $2k$ and $2\tilde{k}$ zeros at $\omega = \pi$, respectively, we can combine (5.18) and (5.19) to get

$$|H(\omega)| = (1 + \cos \omega)^k Q(\cos \omega), \quad |\tilde{H}(\omega)| = (1 + \cos \omega)^{\tilde{k}} \tilde{Q}(\cos \omega), \tag{5.20}$$

where $Q(\cos \omega)$ and $\tilde{Q}(\cos \omega)$ are polynomials in $\cos \omega$. Now we substitute (5.20) in (5.17) and get

$$(1 + \cos \omega)^m Q(\cos \omega) \tilde{Q}(\cos \omega) + (1 - \cos \omega)^m Q(-\cos \omega) \tilde{Q}(-\cos \omega) = 1, \tag{5.21}$$

with $m = k + \tilde{k}$. For convenience we write the equation above as a polynomial in $x = \sin^2(\omega/2)$, giving

$$(1-x)^m P(x) + x^m P(1-x) = 1. \quad (5.22)$$

Our aim is to find a polynomial $P(x)$ that fulfils (5.22) and to split the function $(1-x)^m P(x) = H(\omega)\tilde{H}(\omega)$ into two parts. Then we have found a biorthogonal wavelet filter pair. According to Bezout's theorem there is a unique polynomial $P(x)$ of degree less than m that is a solution to (5.22). By writing equation (5.22) as

$$P(x) = (1-x)^{-m} - x^m (1-x)^{-m} P(1-x), \quad (5.23)$$

we can expand the right-hand side in a Taylor series [Cohe92]. We need only the first m terms, since we know $P(x)$ has maximal degree $m-1$, and thus we get a solution of (5.22)

$$P(x) = \sum_{n=0}^{m-1} \binom{m-1+n}{n} x^n. \quad (5.24)$$

After substitution of this solution and rewriting x in $\sin^2(\omega/2)$, we get

$$H(\omega)\tilde{H}(\omega) = \left(\cos\left(\frac{\omega}{2}\right)\right)^{2(k+\tilde{k})} \sum_{n=0}^{m-1} \binom{m-1+n}{n} \sin^{2n}\left(\frac{\omega}{2}\right). \quad (5.25)$$

The design of our group of symmetric biorthogonal filters thus amounts to choosing two flatness degrees k and $\tilde{k}$ for the two filters $H(\omega)$ and $\tilde{H}(\omega)$ and splitting the right-hand side of (5.25) into two parts.

As an example, we choose $k = \tilde{k} = 2$ and try to create filters that have almost equal length. (Equal length is not possible, see [Nguy89] [Vett92].) Each of the two filters gets its own cosine terms and the summation of squared sines is split into two parts. The summation of squared sines becomes

$$1 + 4x^2 + 10x^4 + 20x^6, \quad (5.26)$$

and the only non-trivial way to factor it into two polynomials with real coefficients is

$$(a_0 + a_1 x^2) (a_2 + a_3 x^2 + a_4 x^4), \quad (5.27)$$

with

$$\begin{aligned} a_0 &= 1.53118822, & a_1 &= a_4 = 4.47213595, \\ a_2 &= 0.65308757, & a_3 &= 0.70487976. \end{aligned} \quad (5.28)$$

The trivial factorization has terms 1 and $1 + 4x^2 + 10x^4 + 20x^6$ and it yields a pair of filters with very different lengths (11 and 5). The solution of (5.28) yields a pair of filters with lengths 9 and 7, which are used frequently in the rest of this thesis. The pair is denoted by the prefix “cdf”, referring to the designers [Coh92], and followed by the filter lengths, thus cdf9-7. We refer to Appendix E for the filter coefficients and the regularity order.

Since a biorthogonal filter pair has two different low-pass filters, there are two possible ways to implement the filter pair in a two-band splitting scheme. To decide which low-pass filter should appear in the analysis and which in the synthesis filter bank, we have a look at the regularity. The regularity order of the longest and shortest filter are 1.04 and 1.70, respectively. This can also be seen from the scaling functions, as shown in Figure 5.6. A quantization error in the subband coefficients causes an error response that has the shape of the synthesis filter. If the decomposition is at least of depth 3 then the error in the small-sized subbands resembles the scaling function and the wavelet function. The HVS is most sensitive to errors in regions of slowly varying grey levels. Therefore, the synthesis of a low-pass should be as smooth as possible, and the short low-pass filter is recommended for the synthesis filter bank. As a consequence of this, the least regular high-pass filter appears also in the synthesis bank [Anto92].

Figure 5.6 The scaling functions of the biorthogonal filter pair cdf9-7.

(a) The scaling function with support 9; (b) The scaling function with support 7.

Other filters can be derived from (5.25) by choosing larger values for k and $\tilde{k}$. Then more solutions may be possible, each generating another set of filters. When we choose $k = \tilde{k} = 8$, for example, we have three solutions that produce filter lengths

differing by 2 (17 and 15). In the next section we use the solution where the energy of the filters is closest. In this case also the regularity is closest.

We remark that the filters above can also be derived with the matrix method. For example, the 12x12 matrix for a symmetric filter pair of lengths 9 and 7 is

$$\mathbf{M} = \begin{bmatrix} h_4 & h_3 & h_2 & h_1 & h_0 & & & & & h_0 & h_1 & h_2 & h_3 \\ h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & & & h_0 & h_1 & \\ h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & & & \\ & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & h_0 & & \\ h_0 & & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & h_2 & h_1 & & \\ h_2 & h_1 & h_0 & & & h_0 & h_1 & h_2 & h_3 & h_4 & h_3 & & \\ f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & & & f_0 & -f_1 \\ f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & & & \\ & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & & & \\ & & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 & & & \\ f_0 & & & & & f_0 & -f_1 & f_2 & -f_3 & f_2 & -f_1 & f_0 \\ f_2 & -f_1 & f_0 & & & & f_0 & -f_1 & f_2 & -f_3 \end{bmatrix}. \quad (5.29)$$

The location of the coefficients in the matrix is based on the filter bank description and offset definitions in Section 3.2. In the same way, the inverse matrix can be designed,

$$\mathbf{M}^{-1} = 2 \begin{bmatrix} f_3 & f_1 & & & f_1 & h_3 & h_1 & & h_1 & h_3 \\ f_2 & f_2 & f_0 & & f_0 & -h_4 & -h_2 & -h_0 & -h_0 & -h_2 \\ f_1 & f_3 & f_1 & & h_3 & h_3 & h_1 & & h_1 & \\ f_0 & f_2 & f_2 & f_0 & -h_2 & -h_4 & -h_2 & -h_0 & -h_0 & \\ & f_1 & f_3 & f_1 & h_1 & h_3 & h_3 & h_1 & & \\ & f_0 & f_2 & f_2 & f_0 & -h_0 & -h_2 & -h_4 & -h_2 & -h_0 \\ & & f_1 & f_3 & f_1 & h_1 & h_3 & h_3 & h_1 & \\ & & f_0 & f_2 & f_2 & f_0 & -h_0 & -h_2 & -h_4 & -h_2 & -h_0 \\ & & & f_1 & f_3 & f_1 & h_1 & h_3 & h_3 & h_1 \\ f_0 & & f_0 & f_2 & f_2 & -h_0 & -h_0 & -h_2 & -h_4 & -h_2 \\ f_1 & & & f_1 & f_3 & h_1 & h_1 & h_3 & h_3 & \\ f_2 & f_0 & & f_0 & f_2 & -h_2 & -h_0 & -h_0 & -h_2 & -h_4 \end{bmatrix}. \quad (5.30)$$

Note that the inverse is not the transpose, because the orthonormality property is lost. In order to have perfect reconstruction the matrix multiplication $\mathbf{M}^{-1}\mathbf{M}$ must

equal the identity matrix, which gives us 4 equations. Then, we require 4 zeros at $\omega = \pi$, that is, we impose 4 vanishing moments of the high-pass filters related to h_n and f_n . These vanishing moments give us another 4 equations. Finally, the coefficient sums of both filters must be equal to 1. The unique solution of this (redundant) set of 10 equations with 9 unknowns is equivalent to the filter pair found with the previous method. Note that the main difference between the two methods is that for the matrix method the filter lengths must be known.

The design process guarantees that the filter coefficient sum is 1, but the coefficient energy may be far from 0.5. Therefore, the design process can be extended in order to take into account the filter coefficient energy. For instance, the number of zero moments can be decreased in favour of a coefficient energy that is closer to 0.5.

5.4 Comparison of SBT and WT filter characteristics

The SBT filters and the WT filters differ in some aspects, as we saw in the previous three sections. Here we summarize the differences in the overview of Table 5.4. The three families of filters examined are the Johnston filters, the Daubechies filters and the biorthogonal filters designed in Section 5.2 and Section 5.3. The first filter family represents the class of SBT filters and the other two the class of WT filters. We continue with some comments on the table.

The filter length is important for two reasons. Firstly, in applications like image segmentation long filters cause inaccurate location of edges, because neighbouring edges fall in the support region of the filter. Secondly, short filters allow faster implementation. Symmetric even length filters allow a polyphase implementation, which reduces the computational complexity by a factor 2.

The subband coding theory allows all kind of splitting schemes and all sizes of filter banks. In contrast, the wavelet theory requires a two-band splitting scheme with recursive splitting of the low-pass subband. However, the ideas and terminology of wavelet theory have been used also for more dimensional filter banks and arbitrary splitting schemes (wave packets [Wick92]).

The orthogonality of the basis functions (and the corresponding filters) does not give rise to uncorrelated subband coefficients. In fact, a wavelet decomposition with orthogonal filters describes a signal as a weighted sum of orthogonal basis functions, whereas the subband coefficients are nothing more than the weights of the basis functions.

Table 5.4 Comparison between representatives of subband coding filters and wavelet filters

Property	Johnston	Daubechies	Biorthogonal
reconstruction	not perfect	perfect	perfect
basis functions	not orthogonal	orthogonal	biorthogonal
regularity w.r.t. filter length	low	maximal	high
magnitude response	good A_p , A_s , and $\Delta \omega$.	flat at half sampling frequency	flat at half sampling frequency
phase	linear	non-linear	linear
signal extension	free (symmetric)	periodic	free (symmetric)
filter length	relatively long	relatively short	relatively short
complexity	$\sim$ filter length / 2	$\sim$ filter length	$\sim$ filter length

In our opinion, the contribution of wavelet theory to image and video coding comprises two parts. Firstly, there is the link of the discrete filtering with the continuous theory of function spaces. This link is formed by the iterated filters and their limit functions, the scaling function and the wavelet. Secondly, there is the regularity of a filter. For image and video coding this regularity is the only new criterion the wavelet theory has added to the design process of filters. Whether this regularity is important is the subject of the next section, where we compare several filters with various orders of regularity.

We conclude with the remark that the distinction between the two types of filters is becoming less sharp than that shown in the examples above. Recent filter design techniques use a mixture of SBT and WT criteria (e.g. [Onno93]).

5.5 Comparison of various filters for image and video coding

Our purpose is to determine differences between the various filters that are used in subband coding schemes. In this section we wish to give an answer to the question whether the regularity order is important for a certain filter class, and, in the case of a biorthogonal filter pair, which of the two filters should be in the synthesis filter bank.

Further, we evaluate whether regularity is important by comparing filters of different design classes (SBT and WT). Additionally, we determine the difference in performance of the original and the modified Johnston filters, and give a general recommendation as to which are the good and the bad filters for image and video coding applications.

In our comparison we used various filters, test images, and test videos. Below we mention the choices we made in the experiments. The filters that we selected for our comparison are:

- the Daubechies filters *dau4* and *dau16* (Section 5.3.1),
- the biorthogonal filters *cdf5-7*, *cdf9-7*, *cdf15-17*, and *cdf17-15*, designed with the method of Section 5.3.2,
- the Johnston filters *joh16B*, *joh16C*, *joh16CC*, and *joh32D* (Section 5.2),
- a filter of length 16 designed by Smith and Barnwell [Smit86] denoted by *sb16*, which is known as a conjugate quadrature filter (CQF),
- a filter pair with lengths 9 and 7 of Kronander [Kron89], denoted by *kro9-7*, which was designed by minimizing a combination of step-response error, passband error, and stopband error.

The filter pairs *cdf17-15* and *cdf15-17* contain the same filters, but the analysis and synthesis filters have interchanged their function. They were designed to get filters of a length comparable to the Johnston filters of length 16. The filter *sb16* was added because it belongs to the earliest class of perfect reconstruction filters. It has non-linear phase. The last two filters were added because they were designed using different criteria than the SBT and WT criteria mentioned. Also for these filters the effect of regularity can be determined. For the filter characteristics of all filters we refer to Appendix E.

For the experiments we used the following test images:

Aerial	a 256x256 image containing an aerial photograph,
Baboon	a 512x512 image of a baboon, containing many high frequencies,
Camera-man	a 256x256 image of a man with a camera,
Clown	a 512x512 image of a clown in front of a mirror,
Lena	a 512x152 image of a woman with hat in front of a mirror, and
Mobile	first frame of a 720x576 sequence with a toy train and a calendar.

The video sequences that were used in the experiments are:

Claire	a 352x288 sequence containing a head-and-shoulder scene with a talking woman before a smooth background,
Vectra	a 352x288 sequence of an Opel Vectra driving in front of a building and where the camera follows the car.

Both video sequences are scanned on a progressive (non-interlaced) raster. The test images and the first frame of the test video sequences are shown in Appendix F. Only the luminance of the images was used.

Our comparison is divided into three parts. Firstly, in Section 5.5.1 we compare the subband coding gains of the filters for still images, and then we present the objective and subjective performance comparisons for still images in Section 5.5.2. Finally, Section 5.5.3 discusses the objective and subjective coding performance for video sequences.

5.5.1 The subband coding gain for still images

One way to compare various filters for signal decomposition and coding is to employ the energy compaction property, also called the subband coding gain. It is a theoretical measure of how much the subband decomposition of the signal can improve the coding efficiency. We calculated this coding gain for several filters in the subband decomposition of a test set of grey-value images. The method is partly theoretical, because the definition of the gain includes some assumptions which may not be true in practice. Still, the experiments give insight into the differences in the compaction capability of the filters. We start with the definition of the subband coding gain and then present the experimental results.

Definition of the subband coding gain

First, the original discrete signal x is assumed to be quantized directly with PCM. The quantizer is fitted to the pdf of the signal. The PCM quantization error variance of the signal is proportional to the variance σ_x^2 , i.e.,

$$\sigma_{r,PCM}^2 = \gamma_{PCM} 2^{-2R} \sigma_x^2, \quad (5.31)$$

where R is the bit rate in bits per pixel and γ_{PCM} is a number related to the pdf [Jaya84].

Then, the signal x with length N is decomposed into M subbands, which may be of unequal size. Subband k has length N_k . The intention of the decomposition is to reduce the number of bits needed to represent a signal when the same distortion is allowed. To which extent this is succeeded is expressed with the subband coding gain, which is defined as the ratio between the PCM quantization error variance and the subband coding error variance $\sigma_{r,SBC}^2$,

$$G_{SBC} = \frac{\sigma_{r,PCM}^2}{\sigma_{r,SBC}^2}. \quad (5.32)$$

If G_{SBC} equals 1 there is no gain and this implies that the decomposition does not give a better coding performance. In general, the gain is greater than 1. To find a formula for the subband coding gain, we have to express the subband error variance in terms of the subband variances, which is done below.

The subband coding error variance is the sum of the error variances of the subbands,

$$\sigma_{r,SBC}^2 = \sum_{k=0}^{M-1} \sigma_{rk}^2 = \sum_{k=0}^{M-1} \gamma_k 2^{-2R_k} \sigma_{xk}^2, \quad (5.33)$$

where subband k has variance σ_{xk}^2 and is coded with bit rate R_k . It assumes that the energy of the signal is divided over the subbands proportionally to their size. However, if the coefficient energy deviates from 0.5, this does not hold. It was shown in [Wood92] how to calculate weights w_k for each subband to correct for this effect.

Including this correction the formula above becomes

$$\sigma_{r,SBC}^2 = \sum_{k=0}^{M-1} \gamma_k 2^{-2R_k} w_k \sigma_{xk}^2. \quad (5.34)$$

The number of bits assigned to each subband must be allocated in such a way that the total bit rate equals R bit per pixel,

$$R = \sum_{k=0}^{M-1} \left(\frac{N_k}{N} \right) R_k. \quad (5.35)$$

To find the optimal bit allocation we have to minimize the error variance (5.34) under the constraint that the total bit rate equals R (5.35). This allocation is found by using the Lagrange multiplier method, which yields

$$R_k = R + \frac{1}{2} \log \left\{ \left(\gamma_k \left(\frac{N}{N_k} \right) w_k \sigma_{xk}^2 \right) / \prod_{l=0}^{M-1} \left(\gamma_l \left(\frac{N}{N_l} \right) w_l \sigma_{xl}^2 \right)^{N_l/N} \right\}, \quad (5.36)$$

(see Appendix C). After substituting (5.36) in (5.34) we obtain

$$\sigma_{r, SBC}^2 = (\gamma_{SBC} 2^{-2R}) \prod_{l=0}^{M-1} \left(\left(\frac{N}{N_l} \right) w_l \sigma_{xl}^2 \right)^{N_l/N}, \quad (5.37)$$

assuming the same pdf and, therefore, the same γ for each subband.

This last assumption is reasonable for all subbands except for the low-pass band, which has obviously different statistics. Values of $\gamma_k = 4.5$ and $\gamma_k = 5.7$ were reported in [Jaya84] for Laplacian and Gamma distributions. As the generalized Gaussian pdfs with scale parameters 0.5 - 0.7 are between the Laplacian and the Gamma pdfs, we do not expect the values to be very different. Another assumption is that the pdf of each subband is the same for every filter. The complete formula for the subband coding gain becomes

$$G_{SBC} = \frac{\gamma_{PCM}}{\gamma_{SBC}} \left(\sigma_x^2 \prod_{l=0}^{M-1} \left(\left(\frac{N}{N_l} \right) w_l \sigma_{xl}^2 \right)^{-N_l/N} \right). \quad (5.38)$$

Comparison of the subband coding gain for several filters and test images

The coding gain for several decompositions of our test images is shown in Figure 5.7, where we ignored the unknown constant $\gamma_{PCM}/\gamma_{SBC}$. We can conclude that the coding gain depends strongly on the image content. Highly detailed images like Aerial and Baboon have small coding gain, because their statistics are closer to Gaussian noise than are images with slowly varying intensity, for instance Lena. For all images the extra splitting of a low-pass subband is profitable when it is at level 1 or 2 in the

tree. After the third level the gain remains almost equal, which indicates that further splitting will not improve coding performance.

When comparing the two Daubechies filters the conclusion is that the most regular filter *dau16* always has a higher gain. Contrary to this, the *cdf15-17* filter pair always has lower gain than the shorter and less regular *cdf9-7* filter, although they originate from the same design procedure. This is an indication that the regularity order is not always an important filter property. Further, we see a clear difference when the analysis and synthesis banks are reversed, in the comparison of *cdf15-17* and *cdf17-15*. The analysis filter bank filters with the highest degree of regularity have a better energy compaction capability, which is unexpected from Section 5.3.2 and [Anto92]. Apparently, the place of the filter in either the analysis or synthesis bank is no longer determined by the regularity if both filters have a sufficiently high degree of regularity, and both filters in the *cdf15-17* pair have a regularity above 2.5. Also in [Zett90] it is argued that a regularity of 2 is sufficient for images, because images are, in general, of a polynomial nature with a degree not higher than 2. Therefore, more vanishing moments of the filter are not needed.

The subband coding gains of the original and modified Johnston filters *joh16C* and *joh16CC* are almost equal. This was to be expected as the modification has only influence on the mean of the subband coefficients.

In general, we conclude that the biorthogonal filter pair *cdf9-7* and the four Johnston filters have the highest coding gain, among which the *joh32D* is the best. The *kro9-7* filter shows a very different behaviour for highly detailed and smooth images and clearly is more suitable for the last class. The only filter with rational coefficients, *cdf5-7*, has a low gain for all images.

Finally, to measure the effect of regularity on the subband coding gain in a straightforward way, we plot them in Figure 5.8 for two test images and a 13-band decomposition. The regularity of the filter pairs is taken to be the regularity of the synthesis filters. We can conclude that there is no correlation between regularity and subband coding gain.

5.5.2 The objective and subjective coding performance for still images

We continue our evaluation of the differences between the filters mentioned at the beginning of Section 5.5 by an objective comparison of the coding performance. As our distortion measure we take the peak signal-to-noise ratio (PSNR), which is defined as

Figure 5.7 Subband coding gains for several filters, decompositions, and test images.

Figure 5.8 Subband coding gain versus regularity. For the filter pairs the regularity of the synthesis filters is used.

$$PSNR = 10^{10} \log\left(\frac{255^2}{MSE}\right), \quad (5.39)$$

where MSE is the mean square error of the compressed image. Although the PSNR does not completely correspond to the subjective visual qualification of the image, it is a commonly used measure for the comparison of different coding techniques. There is no distortion measure yet that models the working of the HVS in a proper way.

The bit rate at which we code our test images ranges from 0.2 to 1.4 bits per pixel (bpp). In Figure 5.9 we show the objective performance of the filters for each image.

We can come to several conclusions. Firstly, the comparison of the Daubechies filters reveals that there is no strict relation between regularity and performance, unlike the outcome of the subband coding gain comparison. In the case of Camera-man the dau16 filter even has a 0.3 dB worse performance than the shorter filters. An explanation for this might be that the non-linear phase of a filter is a larger drawback for very sharp edges. The short Daubechies filter has less non-linear phase than the long filter.

Secondly, we compare the biorthogonal filters. Except for the lowest bit rate, the performance of cdf5-7 with rational coefficients is ca. 1 dB lower than cdf9-7 for both Lena and Clown, and ca. 0.3 dB lower for the other four images. The filter cdf15-17 has always a slightly lower performance than the filter cdf9-7, which is of the same family, but with a lower regularity. Therefore, we conclude that increasing the regularity does not automatically increase the objective coding performance. When the highly regular filters of cdf15-17 are reversed the objective coding per-

formance decreases slightly, which is not expected from Section 5.3.2 and [Anto92]. However, our experiments with the reversed implementation of the cdf5-7 and cdf9-7 filter confirmed the conclusion in [Anto92] that for these filters the most regular filters should be in the synthesis bank. We conclude that when the degree of regularity is high enough, as for the cdf15-17 filters, other filter characteristics than regularity determine which filter should be in the synthesis bank. Note that the analysis filters of kro9-7 even have a negative regularity and, therefore, generate a non-continuous scaling function and wavelet (Appendix E).

Next, the comparison of the Johnston filters clearly demonstrates the decrease in objective coding performance of the joh16C filter. The higher the bit rate, the larger the difference becomes. The difference is most pronounced for images with large smooth areas (5 dB difference with joh16CC). High-detailed images like Aerial do not suffer from the bias in this filter.

Finally, when comparing all filters we can conclude that the best filter is cdf9-7 and the worst are the sb16 filter and the joh16C filter. In general, the differences between the filters are not very big. A high regularity order does not say anything about the objective coding performance. A similar conclusion has been drawn in [Hart93].

The subjective coding performance for still images is evaluated for only two test images and a subset of the filters. In Figure 5.10a/b we compare the filters dau4 and dau16 for a 10-band decomposition and a bit rate of 0.2 bpp. The overall quality of the dau4 filter is higher than of the dau16 filter. The latter suffers more from its obligatory periodic signal extension (Section 3.2), which is reflected by the errors near the image boundaries. Further, there are a lot of artefacts in the face. The edges of the image coded with dau4 are more blocked, but this has less visual importance than the other coding errors.

In Figure 5.10c/d/e/f the compressed Lena image is shown for the filters cdf5-7, cdf9-7, cdf15-17, and cdf17-15 for a 10-band decomposition and a bit rate of 0.2 bpp. The best filter is cdf9-7, although the difference between it and the other two filters is not very large. The cdf17-15 filter gives a lower quality than the reversed filter pair cdf15-17 despite its more regular synthesis filters. We do not show in Figure 5.10 the images coded with the reversed pairs of cdf5-7 and cdf9-7, but the outcomes of our experiments confirmed the conclusion in [Anto92] that for these pairs the most regular filters should be in the synthesis bank.

Figure 5.9 Objective coding performance comparison for the test images. To facilitate good visual interpretation the twelve filter performances are shown in two graphs in juxtaposition.

Figure 5.9 (Continued).

(a)

(b)

Figure 5.10 Comparison of the compressed Lena image with a 10-band decomposition at 0.2 bpp. (a) dau4; (b) dau16; (c) cdf5-7; (d) cdf9-7; (e) cdf15-17; (f) cdf17-15; (g) joh16B; (h) joh16CC.

(c)

(d)

Figure 5.10 (Continued).

Figure 5.10 (Continued).

(g)

(h)

Figure 5.10 (Continued).

When comparing the joh16B and joh16C filter in Figure 5.10g/h we conclude that the first one is slightly more preferable. Although the objective coding performance differs very much for the joh16C and joh16CC filter, the subjective coding performance was ranked equally by all of four experienced test persons. The most probable explanation is that the bias in the original filter just lowers all grey values with an almost fixed value, which causes a slightly darker image, but it has the same coding artefacts.

Comparing the compressed Lena images shown in Figure 5.10, we conclude that the Daubechies filters have the lowest quality, and the cdf9-7 filter the best quality.

Up to now only the Lena image has been considered. For a more general conclusion we show in Figure 5.11 the compression results of the Camera-man image for the filters cdf5-7, cdf9-7, cdf15-17, and joh16B. All four experienced test persons ranked the images also in this order, with the cdf5-7 filter as best. The compression result of the cdf17-15 filter was again ranked below the cdf15-17 filter and there was no agreement on whether the modified Johnston filter joh16CC had a better coding performance than its original. We conclude that except for the interchanged ranking of cdf5-7 and cdf9-7 there is no different conclusion for the two test images.

5.5.3 The objective and subjective coding performance for video sequences

In Figure 5.12 we show the objective coding performances of the filters for video sequences. The two video sequences are compressed using a 10-band decomposition. The presented performances have been derived by taking the average bit rate and PSNR over frame number 2 to 15. The bit rate is given in bits per pixel and ranges from 0.2 to 1.4. For the frame format used, this amounts to ca. 0.5 to 3.5 Mbit/s, that is, medium bit rates. More details about the coding scheme can be found in Section 4.2.

We can conclude that for the Claire sequence all filters perform, except for the joh16C and sb16 filter, equally well. For the first, the less satisfactory performance is caused by its bias, and for the second by its non-linear phase, which is more non-linear than the phase of the dau4 and dau16 filters (Appendix E). For low bit rates the cdf15-17 filter has a lower performance, probably because the energy of the filter coefficients deviate from 0.5 (0.62 and 0.41). For high bit rates the performance of the joh16B filter is worse than that of the others, but we did not find a good explanation for this effect. The Vectra sequence shows an almost equal performance for all

(a)

(b)

Figure 5.11 Comparison of the compressed Camera-man image with 10-band decomposition at a bit rate of 0.55 bpp. (a) cdf5-7; (b) cdf9-7; (c) cdf15-17; (d) joh16b.

(c)

(d)

Figure 5.11 (Continued).

Figure 5.12 Objective coding performance comparison for the test video sequences. The decomposition consisted of 10 bands and the values in the graphs are the averages over frame 2-15. The twelve filters are shown in two graphs in juxtaposition to facilitate a good visual interpretation.

filters used, except for the filters sb16 and joh16C, for the same reason as for the Claire sequence.

When the Vectra sequences coded with 0.2 and 1.0 bpp were displayed on a television screen the four experienced test persons ranked the cdf5-7 filter as best, followed by the cdf9-7 filter. Further, there was a slight preference for the cdf15-17 filter over the Johnston filters, among which no difference in subjective quality was observed. The sb16 filter showed the lowest coding performance.

5.6 Conclusions

Subband coding theory and wavelet theory are two theories that underlie the two-channel analysis/synthesis filter scheme. The contribution of the newer wavelet theory is that it added the link between digital filtering and the theory of continuous function spaces by means of introducing the limit functions (wavelet and scaling function) of the iterated filters. Further, it added the criterion of regularity to the design of filters. For filter design, the wavelet theory concentrates on the regularity criterion, which is closely related to the flatness of the filter frequency response at half the sampling frequency. Contrary to this, filter design according to subband coding theory concentrates on approximating the ideal block shape of the filter frequency response.

To compare the performances of several filters of both categories for image and video coding, we determined the objective and subjective coding performance. Additionally, the subband coding gain was calculated for still images.

No direct relation between regularity and coding performance emerged. The regularity was only useful in deciding where the filters in a filter pair should be placed in the analysis/synthesis scheme: for the filter pairs of which one of the filters has a low regularity, the implementation with the most regular filters in the synthesis bank showed a better performance. The cdf15-17 filter pair has a higher regularity than the filter pair cdf9-7 of the same family, but an overall less satisfactory performance.

The modified filter joh16CC performs much better than its original considering the objective coding performance, but it showed hardly any difference with respect to subband coding gain and subjective coding performance.

In general, the biorthogonal wavelet filters have performed best, followed by the Johnston filters, the Daubechies filters and the Smith and Barnwell filter. The filter pair of Kronander has been shown to be most suitable for smoothly varying images. When all three criteria for performance are taken together, the cdf9-7 showed the best performance, although in some cases the subjective coding performance of cdf5-7 was better.

Chapter 6

The Two-Dimensional Discrete Wavelet Transform

In the previous chapters we only applied the one-dimensional DWT. For the two dimensions in images, the transform consisted of applying twice a 1-D DWT, one for the rows and one for the columns of the images, as was introduced in [Mall89a]. This manner of applying a DWT on a two-dimensional signal that is sampled on a rectangular lattice is here called the dyadic 2-D DWT. The word dyadic refers to the scale factor 2 in both directions. In this chapter, we describe a more general 2-D discrete wavelet transform which uses non-separable filters in general, and which is also defined on signals that are sampled on a hexagonal lattice. Thus, we can make better use of the two-dimensional correlation in the signal and, therefore, this transform is expected to achieve a higher coding gain than the dyadic 2-D DWT.

In this chapter we concentrate on one variant, the quincunx 2-D DWT. It has a scale factor that is equal in all directions. If the filters have circular symmetry, then this transform has the additional advantage that it does not emphasize a particular direction in the plane, which is useful for some image coding applications like the storage of aerial photographs. The quincunx 2-D DWT is very similar to the 1-D DWT as it uses only one wavelet function, as opposed to other variants of the general 2-D DWT.

We wish to investigate whether this alternative two-dimensional transform will lead to a better image representation and a better decomposition in comparison to the popular dyadic transform. The dyadic 2-D DWT has two advantages. Firstly, it is

easy to implement. The separate processing of rows and columns requires the same computational complexity and hardware realization as needed for the processing of one-dimensional signals. Secondly, the transform has the property that it emphasizes the directions along the axes in comparison to all other directions in the plane. As the human visual system is more sensitive to the horizontal and vertical directions in comparison to other directions [Marr82], the dyadic 2-D DWT is suited to most image coding applications.

In Section 6.1, we give the description of the dyadic 2-D DWT in terms of basis functions. Then we define in Section 6.2 the general 2-D DWT together with the multiresolution analysis for two dimensions, after which in Section 6.3 the particular case of the quincunx 2-D DWT is worked out. Section 6.4 contains experiments on the comparison of the quincunx 2-D DWT with the dyadic 2-D DWT. The chapter is closed with conclusions in Section 6.5.

6.1 The dyadic two-dimensional discrete wavelet transform

The dyadic two-dimensional implementation of the discrete wavelet transform was introduced in [Mall89a]. We have already applied it in Chapter 4 and Chapter 5. It is based on the same multiresolution analysis as that in one dimension, with scaling function $\phi(x)$ and wavelet $\psi(x)$ defined as in Chapter 2. The extension to two dimensions is performed by the defining the scaling function as

$$\phi_{LL}(x, y) = \phi(x)\phi(y), \quad (6.1)$$

and three wavelet functions as

$$\psi_{LH}(x, y) = \psi(x)\phi(y), \quad \psi_{HL}(x, y) = \phi(x)\psi(y), \quad \psi_{HH}(x, y) = \psi(x)\psi(y). \quad (6.2)$$

The dilated and shifted versions of the scaling function $\phi_{LL}(x)$ build up the multiresolution approximations V_m of $L^2(R^2)$. These approximations can be described as a tensor product of two identical multiresolution approximations of $L^2(R)$, which we denote by V_m' in this chapter,

$$V_m = V_m' \otimes V_m'.$$

The three wavelets of the scale parameter m form together the orthonormal bases for the orthogonal complements of V_m and all wavelets of all scales build up $L^2(R^2)$.

The dyadic 2-D DWT decomposes the original image into four subbands, as shown in Figure 6.1. One of the subbands contains all the low-pass information (LL) and another all the high-pass information (HH). The other two subbands are a horizontal high-pass band containing vertical low-pass information (LH) and a vertical high-pass band containing horizontal low-pass information (HL). For a full DWT the decomposition is repeated several times on the LL part.

LL	LH
HL	HH

Figure 6.1 First stage of the 2-D dyadic DWT. The LL part is decomposed iteratively in the same way.

6.2 The general two-dimensional discrete wavelet transform

In this section we extend the multiresolution analysis to two-dimensional signals [Cohe93a] [Kova92]. We define the two-dimensional equivalent of Mallat's one-dimensional multiresolution analysis in a way analogous to the one in Section 2.2. The general 2-D DWT follows directly from this multiresolution analysis as in the one-dimensional case.

The multiresolution analysis is a description of the space $L^2(R^2)$ as a hierarchy of embedded subspaces V_m which have intersection $\{0\}$ and for which the limit of the union is $L^2(R^2)$. That is,

$$\begin{aligned} \dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots \\ \bigcap_{m \in \mathbb{Z}} V_m = \{0\}, \quad \overline{\bigcup_{m \in \mathbb{Z}} V_m} = L^2(R^2) \end{aligned} \quad (6.3)$$

Further, these V_m have the property that for each function $f(x) \in V_m$, $x \in R^2$, it holds that

$$f(x) \in V_m \Leftrightarrow f(Dx) \in V_{m-1}, \quad (6.4)$$

where the 2×2 matrix $\mathbf{D}$ has integer elements and its eigenvalues have absolute value greater than 1. Finally, in a multiresolution analysis, there is a function $\phi(x) \in V_0$ such that its shifted versions $\phi_{0,\mathbf{n}}(\mathbf{x}) = \phi(\mathbf{x} - \mathbf{n})$, $\mathbf{n} \in \mathbb{Z}^2$ constitute an orthonormal basis of V_0 .

The matrix $\mathbf{D}$ is the analog of the downsampling factor 2 in one dimension. The absolute value of the determinant of the matrix, denoted by $|\mathbf{D}|$, is the downsampling factor, whereas the elements of the matrix indicate which samples are retained and which are discarded. The lattice of retained sample points is written as $\mathbf{D}\mathbb{Z}^2$, which is shorthand for the set $\{\mathbf{D}\mathbf{x} | \mathbf{x} \in \mathbb{Z}^2\}$. As the elements of the matrix $\mathbf{D}$ are integer, the lattice $\mathbf{D}\mathbb{Z}^2$ is a sublattice of $\mathbb{Z}^2$, which is important in practical applications. Also theoretically, it seems that the elements must be integer in order to create a multiresolution analysis based on a single, real-valued, compactly supported scaling function [Coe93a]. The eigenvalues of the matrix $\mathbf{D}$ represent the scaling factors in two directions. Both eigenvalues must be greater than 1 to ensure a contraction in all directions. Note that for $\mathbf{D} = 2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix, we get the dyadic transform of the previous section, with total downsampling rate 4 (2 per dimension). Then the matrix is diagonal and defines, in fact, a degenerate 2-D transform, because both dimensions are handled separately.

With the definition of a two-dimensional multiresolution analysis, we derive some properties for the scaling function and wavelets. First, because the functions $\phi_{0,\mathbf{n}}(\mathbf{x})$ are an orthonormal basis for V_0 , it follows that the functions

$$\phi_{m,\mathbf{n}}(\mathbf{x}) = |\mathbf{D}|^{-m/2} \phi(\mathbf{D}^{-m}\mathbf{x} - \mathbf{n}) \quad (6.5)$$

constitute an orthonormal basis for V_m . Since $\phi_{0,0}(\mathbf{x}) \in V_0 \subset V_{-1}$, we can write $\phi_{0,0}$ as a linear combination of basis functions of V_{-1} , i.e.,

$$\phi(\mathbf{x}) = \sqrt{|\mathbf{D}|} \sum_{\mathbf{n} \in \mathbb{Z}^2} h_{\mathbf{n}} \phi_{-1,\mathbf{n}}(\mathbf{x}) = |\mathbf{D}| \sum_{\mathbf{n} \in \mathbb{Z}^2} h_{\mathbf{n}} \phi(\mathbf{D}\mathbf{x} - \mathbf{n}) \quad (6.6)$$

which becomes in the frequency domain

$$\Phi(\omega) = H(\mathbf{D}^{-1}\omega) \Phi(\mathbf{D}^{-1}\omega) = \prod_{n=1}^{\infty} H(\mathbf{D}^{-n}\omega). \quad (6.7)$$

Here we again have normalized the filter $h_{\mathbf{n}}$ such that $H(\mathbf{0}) = 1$, that is, the sum of the filter coefficients equals 1.

A set of $|\mathbf{D}| - 1$ wavelets is related to the scaling function $\phi(\mathbf{x}) = \phi_{0,0}(\mathbf{x})$. Each wavelet $\psi_k(\mathbf{x}) = \psi_{k,0,0}(\mathbf{x})$ is a basis function of the space $W_{k,0}$, where

$$V_{-1} = V_0 \oplus W_{1,0} \oplus \dots \oplus W_{|\mathbf{D}|-1,0}. \quad (6.8)$$

The wavelets $\psi_k(\mathbf{x})$ can be written as linear combinations of basis functions of V_{-1} , i.e.,

$$\psi_k(\mathbf{x}) = |\mathbf{D}| \sum_n g_{k,n} \phi(\mathbf{D}\mathbf{x} - \mathbf{n}), \quad k = 1, 2, \dots, |\mathbf{D}| - 1, \quad (6.9)$$

and the set

$$\{|\mathbf{D}|^{-m/2} \psi_k(\mathbf{D}^{-m}\mathbf{x} - \mathbf{n}) \mid k = 1, \dots, |\mathbf{D}| - 1, m \in \mathbb{Z}, n \in \mathbb{Z}^2\} \quad (6.10)$$

constitutes an orthonormal basis for $L^2(\mathbb{R}^2)$ [Kova92].

The 2-D DWT is performed by filtering the signal $|\mathbf{D}|$ times with the two-dimensional filters h_n and $g_{k,n}$, $k = 1, \dots, |\mathbf{D}| - 1$. Each of the filtered signals is downsampled by retaining only the samples on the lattice $\mathbf{D}\mathbb{Z}^2$. For a full DWT decomposition, this procedure can be repeated on the low-pass subband. An example of such a filtering and downsampling is shown in the next section on the quincunx 2-D DWT.

From (6.5) we see that the scaling functions of two consecutive levels are related by a combined rotation and dilation, and a translation. If we wish the dilation to be the same in all directions, we should separate the rotation and scaling, which means that we have

$$\mathbf{D} = a \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}. \quad (6.11)$$

There are only two basic possibilities for the angle θ that lead to a matrix $\mathbf{D}$ with integer elements, and the corresponding lattices are the square and the triangular lattice. The angle θ is either a multiple of $\pi/4$, with smallest scale factor $a = \sqrt{2}$, or a multiple of $\pi/6$, with smallest scale factor $a = \sqrt{3}$. In both cases, integer multiples of π must be excluded, otherwise one dimension is not scaled. The two downsampling lattices with the smallest possible scale factors are shown in Figure 6.2. In the following sections we concentrate on the square lattice, as our two-dimensional signals are sampled on a rectangular grid.

Figure 6.2 Two regular scalable lattices in R^2 , defined by a rotation matrix R_θ and a scale factor a . The original lattice is represented by the corners of the dotted lines and the rotated/scaled lattice is represented by the circles.

(a) $\theta = \pi/4, a = \sqrt{2}$; (b) $\theta = \pi/6, a = \sqrt{3}$.

6.3 The quincunx two-dimensional discrete wavelet transform

In this section, we examine the special case of the quincunx 2-D DWT. The name quincunx originates from the lattice corresponding to the downsampling matrix D , in which the points are positioned relative to each other as the five spots on one side of a die (see also Figure 6.4). Quincunx downsampling has been introduced earlier in the subband coding literature [Vett84]. It has also been introduced as a special sampling for television signals in the temporal domain [Tong81]. We remark that the quincunx 2-D DWT is closely related to the 2-D CTWT described in Appendix A.

If we compare the multiresolution analyses of the dyadic and quincunx case, we see that the scale factors are 2 and $\sqrt{2}$, respectively. This means that the levels in the multiresolution scheme are closer for the quincunx 2-D DWT. This is shown schematically in Figure 6.3. With the intermediate resolution levels, we might expect to be able to analyse a signal more accurately than for the dyadic transform, since there are more resolution levels. Also, in a hierarchical coding scheme, the resolution for displaying an image can be adjusted better to the available bit rate.

Figure 6.3 Low-pass image pyramid of a 2-D multiresolution analysis. (a) Dyadic; (b) Quincunx.

The downsampling matrix

The quincunx 2-D DWT downsampling matrix has downsampling rate 2, which is the square of the scale factor. The value of 2 means that there is one wavelet function, as in the 1-D case. There are two basic variants for the downsampling matrix D ,

$$D = R = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}, \quad D = S = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad (6.12)$$

[Kova92] [Coh93a]. The scaling functions and wavelets may be completely different for both matrices, even if the filter coefficients are the same. In other words, the choice of the downsampling matrix influences the shape of the basis functions

[Groc92]. The quincunx multiresolution analysis and its application to image coding were first published in [Feau90], where the matrix S was used.

If we take, for example, the matrix R , then the scaling functions and wavelets rotate over $\pi/4$ and dilate by factor $\sqrt{2}$ for increasing m . In the frequency domain the frequency responses of the scaling functions rotate over $-\pi/4$ while contracting by factor $\sqrt{2}$. If the filters are symmetric in the horizontal and vertical directions then there is no difference between the decompositions produced by the matrices R and S .

The downsampling is performed by retaining only the points on the lattice DZ^2 , which amounts to halving the number of pixels in each line and alternating the place of the retained sample per line. This is visualized in Figure 6.4. Note that the quin-

Figure 6.4 The quincunx downsampling procedure.

The downsampling rate is $\sqrt{2}$ per dimension, thus 2 in total.

cunx lattice is essentially a rectangular lattice which has been rotated over $\pi/4$. Therefore, the low-pass subband can be repeatedly filtered to perform a complete DWT.

The design of the filters

For the quincunx 2-D DWT we have one low-pass filter and one high-pass filter as in the one-dimensional case. We can design the set of filters in several ways. We choose

for the McClellan transform [McCl73] which directly derives the two-dimensional filter from a one-dimensional filter. This method has also been suggested in [Kova92] [Coh92] and has been applied to image coding in [Anto91]. Other approaches to the design of two-dimensional filters can be found in [Visc91][Vaid93]. In these approaches, the concept of regularity is not addressed.

The method of the McClellan transform starts with rewriting the frequency domain description of a zero-phase filter h_n with odd length $2K + 1$ by

$$H(\omega) = \sum_{n=-K}^K h_n e^{-i\omega n} = \sum_{k=0}^K a_k \cos k\omega = \sum_{k=0}^K a_k T_k(\cos \omega), \quad (6.13)$$

for some sequence a_k . Here, $T_k(\cdot)$ is the k -th Chebyshev polynomial of the first kind [Abra65]. These polynomials have the property that $T_k(\cos \omega) = \cos k\omega$. Analogously,

$$H(\omega_1, \omega_2) = \sum_{n_1=0}^M \sum_{n_2=0}^N b_{n_1, n_2} T_{n_1}(\cos \omega_1) T_{n_2}(\cos \omega_2) \quad (6.14)$$

defines a two-dimensional zero-phase filter for arbitrary positive M and N .

The McClellan transform consists of substituting $\cos \omega$ in (6.13) by a linear combination of 1, $\cos \omega_1$, $\cos \omega_2$, and $\cos \omega_1 \cos \omega_2$, denoted by $f(\omega_1, \omega_2)$. To guarantee perfect reconstruction, $f(\omega_1, \omega_2)$ must be real-valued and bounded between -1 and 1. Then (6.13) can always be rewritten in the form of (6.14).

If we choose

$$f(\omega_1, \omega_2) = (\cos \omega_1 + \cos \omega_2)/2, \quad (6.15)$$

then the two-dimensional filter inherits all zeros of the one-dimensional filter at $\omega = \pi$ on the position $(\omega_1, \omega_2) = (\pi, \pi)$. Further, the support in the frequency domain becomes almost circular symmetric for the low frequency region. The regularity order does not transfer identically to the two-dimensional filter. In fact, the regularity estimation of two-dimensional filters is very complicated [Vill93][Coh93a].

In Appendix E, we show the filter coefficients of the two-dimensional filter that is related by (6.15) to the biorthogonal filter cdf9-7 defined in Section 5.3.2. This 2-D filter has been used in [Anto91] and is evaluated in our comparison of dyadic and quincunx 2-D filters in Section 6.4.

The implementation in a coding scheme

The two-channel scheme of the quincunx 2-D transform is analogous to the one-dimensional case. The scheme is shown in Appendix C together with the input-output relation. The implementation of the high-pass filter must be performed with different delays in the x and y directions, which has also been noted in [Vett84]. In [Feau90] the same idea is expressed when the author states that the downsampling after high-pass filtering should take place on the alternative lattice, that is, on

$$DZ^2 + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (6.16)$$

in our notation. This fact can also be seen from the orthogonality relation between the low-pass and high-pass filter shown in (6.17), which is the two-dimensional equivalent of (2.34),

$$0 = \langle \phi_{0,0}, \psi_{0,0} \rangle = \sum_n \sum_p h_n g_p \langle \phi_{-1,n}, \phi_{-1,p} \rangle = \sum_n h_n g_n, \quad (6.17)$$

where the high-pass filter g_n is defined by

$$\psi_{0,0}(x) = |D| \sum_n g_n \phi_{-1,n}(x). \quad (6.18)$$

A solution for (6.17) is

$$g_n = (-1)^{1^T n} h_{L - \begin{bmatrix} 1 & 0 \end{bmatrix}^T - n}, \quad (6.19)$$

with the vector L containing the lengths of the filter h_n in both directions.

Here we clearly see the difference between the x and y directions. The relation between filter delay and sampling on the alternative lattice is explained in more detail in Appendix C.

Boundary problems in the implementation are solved by extending the image either periodically or symmetrically, depending on the filter type used. The direction of extension is always parallel to the x and y axes, also for odd-numbered levels in the decomposition, otherwise perfect reconstruction is not possible. This can be easily verified by the condition in Section 3.3, which states that filtering outside the signal borders may not introduce new subband samples.

An example of a quincunx decomposition

In Figure 6.5, we show the three-level decomposition of the test image Camera-man, which produces four subbands. For an efficient storage of the odd-level subbands, the relative position of the subband coefficients on the quincunx lattice are changed. This is performed by shifting half of the rows of the quincunx lattice horizontally over one pixel. Then we have obtained a subband on a rectangular lattice, which has half the number of columns and the same number of rows as the subband it originates from, as shown in Figure 6.5b/c [Feau90]. From this representation, we clearly see that the decomposition has the same number of subband coefficients as there are pixels in the original image. However, the most proper way of showing odd-level subbands is by rotating them over $\pi/4$. Then, the relative distances between the subband coefficients are represented correctly on a rectangular lattice as is shown in Figure 6.5d/e.

The application to video coding

For video coding, the quincunx downsampling is most often used for the temporal direction [Tong81]. This is especially useful in the case of interlaced transmission. However, we investigate the quincunx 2-D DWT only for the spatial coding, for the same reasons as explained in Section 4.2 on video coding.

The main difference between a dyadic and a quincunx coding scheme is the motion estimation and compensation part. Of course, this difference appears only when scheme II or III shown in Section 4.2 are used. All other parts in the coding scheme, like the bit allocation and the quantization, are essentially the same.

If the decomposition has an even number of levels, it ends up with a low-pass subband on a rectangular (non-quincunx) lattice. This enables all conventional motion estimation techniques to be applied on the low-pass subband. In most applications, it will not be necessary to explicitly have an odd number of levels in the decomposition and, therefore, motion estimation techniques for a quincunx lattice do not have to be considered.

If the motion vectors of the low-pass subband are passed to the other subbands and used for compensation, then we must know how the coefficients in the intermediate (odd-level) subbands are related to the low-pass subband. We show in Figure 6.6 how the samples in every subband are located with respect to the original signal, so that the motion compensation can be performed in the best way. These locations can be obtained by accurate observation of the centres of the filters. Figure 6.6 shows for a 2-D symmetric filter pair with odd lengths, the relative position of the sample information in the subbands with respect to the original sampling lattice.

(a)

(b)

(c)

(d)

(e)

Figure 6.5 Quincunx two-dimensional decomposition of the test image Camera-man. (a) Original; (b) Schematic three-level decomposition; (c) The three-level decomposition of the image; (d) (e) The low-pass and high-pass subband of level 3 enlarged with factor 2 and properly displayed.

Figure 6.6 The relative position of the sample information in the subbands with respect to the pixel positions in the original image. These positions hold for symmetric odd-length 2-D filters.

6.4 Comparison of the dyadic and quincunx wavelet transform

In this section, we compare the quincunx with the dyadic 2-D DWT. First, we explain our choice of filters, images, and decompositions. Then we compare the subband coding gain, the objective coding performance in PSNR, and the subjective coding performance (visual quality).

For our comparison, we take the quincunx filter that is derived from the cdf9-7 filter of Section 5.3.2 with the McClellan transform, as explained in the previous section. This filter is here denoted by cdf9-7Q and its performance is compared to the dyadic filter cdf9-7 . By experiments, we verified that these two filters yield subband coefficients that have the same pdf. Therefore, differences in coding performances cannot be caused by our quantization, which is based on a certain pdf model as explained in Section 4.1.

Our set of test images consists of four images. It contains two high-detailed images (Aerial and Baboon) and one image with vertical and diagonal edges (Camera-man). The fourth image has large areas of low frequency (Lena). Information about the image contents and sizes can be found in Section 5.5 and Appendix F.

The applied decompositions are the 10- and 7-band splitting for the dyadic case, and the 7- and 5-band splitting for the quincunx case, which are shown in Figure 6.7. We compared a decomposition of the quincunx scheme with the decomposition of the dyadic scheme that contains a low-pass subband of the same size. This seems the fairest way of comparison, as it corresponds to the same resolution in the image pyramids of Figure 6.3. However, we also added the possibility to compare decompositions with the same number of subbands (seven).

We started our comparison with the calculation of the subband coding gain (Section 5.5). The gains are presented in Table 6.1 for the four test images and the two subband decompositions. The conclusion is that the quincunx subband coding gains are less than the dyadic subband coding gains. Even the gain of the 7-band dyadic splitting of Baboon is higher than the 7-band quincunx splitting. Note that the quincunx subband coding gains are relatively better (more close to the dyadic) for the Aerial and Camera-man images than for the other two images. This was to be expected, as these images have no direction preference (Aerial) and sharp diagonal edges (Camera-man), two properties for which the quincunx 2-D DWT should be suited better.

Figure 6.7 The four subband decompositions used in the comparison.
(a) Dyadic 10 band; (b) Quincunx 7 band; (c) Dyadic 7 band; (d) Quincunx 5 band.

Table 6.1 Subband coding gain for dyadic and quincunx decompositions

	Bands	Aer	Bab	Cam	Len
cdf9-7 (dyadic)	10	3.03	4.41	13.92	54.22
cdf9-7Q (quincunx)	7	3.00	3.97	13.64	49.30
cdf9-7 (dyadic)	7	2.94	4.11	12.51	47.92
cdf9-7Q (quincunx)	5	2.91	3.70	12.27	43.70

Then, we determined the objective coding performance of the two types of filters. Comparing the decompositions with the same size of low-pass subband, the performance of the dyadic scheme was always better than the quincunx scheme. Only for low bit rates the performance of the quincunx 7-band splitting was higher than the dyadic 7-band splitting (see Figure 6.8). This is the direct consequence of the larger size of the low-pass subband in the dyadic case. Then the DPCM coding of this band is provided with too few bits and the overall PSNR decreases. The expectation that the diagonal edges of the Camera-man image would be better coded with the quincunx transform turned out quite differently now. Probably the diagonal edges are so sharp that the coefficients in the vertical and horizontal subbands of the dyadic transform are also significant and therefore are quantized relatively accurately. For the non-directional Aerial image, the quincunx transform showed only a slightly lower performance. Therefore, our general conclusion is that the objective coding performance of the quincunx 2-D DWT is lower than the dyadic transform.

Finally, Figure 6.9 shows two coded images for the evaluation of the subjective coding performance (visual quality). Again we can conclude that the quincunx 2-D DWT performs less well. The coding errors near the edges of the shoulder of Lena and the roads at the top of Aerial are much larger than for the dyadic transform.

We remark that the 2-D filters of [Feau90] have not been taken into consideration, although these were the first used in a 2-D DWT based on a multiresolution analysis. These truncated infinite impulse response filters do not allow perfect reconstruction and perform worse than the cdf9-7Q filter, both objectively and subjectively.

6.5 Conclusions

We have derived a general two-dimensional discrete wavelet transform from the two-dimensional extension of the multiresolution analysis. The dyadic and quincunx 2-D DWT are two special cases that have been considered in more detail. The dyadic transform is a degenerate 2-D DWT as it does not make full use of the two-dimensional characteristics in a signal. However, it has a simple implementation with low computational complexity because of the separability of the filters. The quincunx transform is similar to the 1-D DWT in that it uses one basic wavelet function. The computational complexity is high, because the filters are not separable.

For the application to video coding, the number of subbands should be odd. Then the low-pass subband is defined on a rectangular (non-quincunx) lattice, which enables all conventional motion estimation techniques. We have shown how the samples

Figure 6.8 Objective comparison of the coding performances of the dyadic and quincunx 2-D DWT for some images.

in every subband are located with respect to the original signal, so that the motion compensation in subbands can be performed in the best possible way.

We have compared the quincunx 2-D DWT with the dyadic 2-D DWT for the coding of four test images. The performance of the quincunx transform has been lower with respect to subband coding gain, objective coding performance, and subjective coding performance. Since also the computational complexity is high in comparison to the dyadic 2-D DWT, the quincunx transform is not recommended for the filter used in the experiments. Further research on the quincunx 2-D DWT should be dedicated to the design of better filters, which might perform better than the filter considered here. Then the coding performance of the quincunx 2-D DWT should again be compared with the dyadic 2-D DWT.

(a)

(b)

(c)

(d)

Figure 6.9 Subjective coding performance for two image coding examples. (a) Aerial, dyadic 10 bands, 0.87 bpp, 27.2 dB; (b) Aerial, quincunx 7 bands, 0.85 bpp, 26.5 dB; (c) Lena, dyadic 10 bands, 0.21 bpp, 32.3 dB (shown partially); (d) Lena, quincunx 7 bands, 0.22 bpp, 31.7 dB (shown partially).

Chapter 7

A Region-Based Discrete Wavelet Transform

In this chapter we introduce a region-based discrete wavelet transform (RBDWT), which is a new method for coding the texture in a contour-texture coding technique.

Contour-texture, or region-based, coding techniques have a different approach to coding images than the transform coding methods, like DCT and DWT. The idea of region-based coding is to divide the image into regions, described by their contours and texture, which are coded separately. This type of coding method has been extensively discussed in [Kunt85]. With region-based coding, the edges in an image are preserved better than with transform coding.

The most common method for coding the texture of a region is to fit a polynomial up to some order to the grey-values in a region. Here, we propose a new method for coding the texture: the DWT. Up to now, the DWT as described in literature could not be used to code the texture. The reason for this is that a d -level decomposition requires the regions to consist of square blocks of size $2^d \times 2^d$. In general, such regions will not correspond to the shape of the objects in a scene. However, the efficient signal extension method described in Section 3.4 enables us to apply a DWT on arbitrarily sized regions. To distinguish this implementation of the DWT from the standard DWT, we use the name region-based DWT (RBDWT). From another point of view, the RBDWT forms a special kind of DWT, for which the subbands are divided into regions.

In Section 7.1, we explain in brief the principle of contour-texture coding methods. Next, the general idea and the details of the implementation of the RBDWT are described in Section 7.2. In Section 7.3, the coding scheme of the RBDWT is worked out and in Section 7.4 we present an overview of the properties, the advantages, and the disadvantages to be expected for the new method. Section 7.5 contains a comparison of the RBDWT with the standard DWT for the coding of two test images. We discuss the application of the RBDWT to video coding in Section 7.6. The chapter ends with Section 7.7 on conclusions and subjects for further research.

7.1 Contour-texture coding

Before introducing the principle of the RBDWT as a method for coding texture in a contour-texture coding method, we first explain the general principle of contour-texture coding.

Contour-texture coding techniques attempt to segment the image into regions surrounded by contours such that the contours correspond, as much as possible, to those of the objects in the image [Kunt85]. The transmission of the image information takes place in two stages: first the contours, also called the segmentation information, and then the texture, which is equal to the segment contents. This is a natural approach, because contours and texture coincide with psychological concepts of vision.

For the segmentation, several algorithms have been proposed during the last decades, for instance, region-growing and split-and-merge methods [Kunt87]. We use in our experiments the split-and-merge method described in [Stra91], which we summarize below.

The split-and-merge algorithm starts with a quad-tree splitting of the image up to a certain level. Then each group of four 'child' blocks with the same 'parent' is examined to decide if they should be merged according to the similarity of the grey-value mean and variance inside the blocks. This process is repeated recursively. After the merging step, a recursive splitting process is started. For every block, it is evaluated whether it should be split into four child blocks, depending on the variation in grey-values. Finally, the segmentation is obtained by a grouping procedure that merges blocks from different subtrees if they have similar characteristics.

The segmentation information can be coded with a chain-code [Free61], which under certain assumptions may only cost 1.27 bits per contour node [Eden85].

Most often, the texture information is transmitted by the coefficients of a 2-D polynomial that is fitted to the segment data. Generally, this polynomial is chosen to be of degree 0, 1, 2, or 3, depending on the bit rate and the region properties, such as variance and size. Another method for texture coding is the use of generalized orthogonal transforms, introduced in [Gilg89]. Here, the basis images of the transform are calculated by orthogonalizing standard basis images with respect to the shape of the region. The weights of the basis images are transmitted after they have been quantized to get at the desired bit rate.

We end this section by remarking that the coding performance of the contour-texture method relies heavily on the quality of the segmentation. Therefore, the segmentation process should always be optimized with regard to the texture coding method.

7.2 Principle and implementation of the RBDWT

Region-based coding methods are based on the division of the image into regions. Each region can be coded with an amount of bits that reflects its relative importance in respect to the other regions. In contrast, transform methods like the DWT decompose the image into subbands, each of which gets from the bit allocation algorithm a number of bits corresponding to the importance of the subband. Figure 7.1 illustrates how a contour-texture coding scheme based on the RBDWT combines both the division into regions and the division into subbands. Now, the bit allocation algorithm can distribute bits over all subband segments, such that important features in the spatial as well as the frequency domain can be represented accurately.

Therefore, in addition to considering the RBDWT as the texture coding part in a contour-texture coding method, one can also view the RBDWT as a DWT for which all subbands are segmented as well. Any subband segment can be filtered, quantized, and reconstructed independently of all other subband segments of the same level. This viewpoint can be found back in our description of the coding scheme in Section 7.3.

In the following, we describe in detail the practical implementation. We assume that the segmentation into regions has already been performed by a proper split-and-merge method, taking into account that the texture is coded by the RBDWT. We start in Section 7.2.1 by explaining how to decompose the regions of the original image, the *segmentation mask*, into four subbands, while preserving the possibility of perfect reconstruction. In Section 7.2.2 we describe how the filtering inside regions takes place.

Figure 7.1 Schematic illustration of the RBDWT being a combination of the DWT and a region-based coding method. (a) Original image, split up into: (b) subbands (as used by DWT); (c) regions (as used in region-based coding); (d) subbands and regions (as used in RBDWT).

7.2.1 Decomposition of the segmentation mask

The standard (dyadic) 2-D discrete wavelet transform for images decomposes an image into four subbands (called LL, LH, HL, and HH), after which this splitting can be repeated on the subbands. In Figure 7.2 such a decomposition is shown.

Figure 7.2 A splitting into four subbands. The LH subband contains horizontal high-pass and vertical low-pass information, and so on.

Even when the image is divided into regions, we still wish to be able to decompose the image into four subbands, where each subband is subdivided into regions as well. Four child regions of decomposition level 1 correspond to one parent region of the original image. The division of the subbands into regions has the restriction that each parent region in the original image must be perfectly reconstructible from the four corresponding child regions in the subbands. This means that the total number of subband pixels in the four child regions must be at least equal to the size of the parent region, otherwise some information will be lost. Since the total number of subband pixels remains the same, equality is required.

The above requirement for the segmentation of the subbands can be achieved with the method as demonstrated in Figure 7.3. The segmentation mask of the original image is subdivided into 2x2 blocks and for each block the upper-left part is put into the LL band. In the same way, the upper-right part of each block falls in the LH band, the lower-left into the HL band, and finally, the lower-right part is copied into the HH band. This procedure guarantees that the total number of pixels in one region remains the same. The procedure can be viewed as an algorithm to downsample in four ways a segmentation by a factor 2 per dimension. We see in every subband in Figure 7.3c a kind of jig-saw puzzle in which the pieces exactly fill up the rectangular shape of the subband. Note that the two congruent regions in the centre are divided over the sub-

(a)

(b)

(c)

Figure 7.3 The process of segmentation of the subbands; (a) The segmentation of the original image; (b) The subdivision into 2×2 blocks; (c) The segmentation of the subbands.

bands in a different way, because of the different parity of the pixel coordinates. Of course, the decomposition can be repeated on any of the subbands.

7.2.2 The filtering and downsampling of the regions

We continue with the description of the filtering and downsampling process of the RBDWT. We have the requirement that the wavelet decomposition of a region must be totally independent of neighbouring regions. In order to enable this, we have to apply the efficient signal extension method as explained in Section 3.4, which preserves the perfect reconstruction property. The filtering and downsampling process is explained for the rows of the image. The columns are handled in an analogous manner.

Processing of segment lines of length greater than one

The row filtering starts at the upper left corner of the image and the rows are scanned from top to bottom, as usual. We do not filter per image row, as in ordinary wavelet filtering, but we filter per segment line, where a segment line is defined to be a horizontal (or, vertical in the case of column filtering) sequence of connected pixels from the same region. Thus, our first signal to be filtered in the example given in Figure 7.3a is a signal of length 6 and the next two signals are of lengths 7 and 5. The segment lines can be divided into four classes that each require their own implementation of the filtering and downsampling. The four classes are distinguished by the parity of the column number at which the segment line starts and by the parity of the segment line length. Below, we explain how the filtering and downsampling take place in each of the four classes.

1. The segment line starts at an even-numbered column (0, 2, ...) and has even length. In this case the filtering and downsampling take place as usual (Section 3.2). The signal is extended at the boundaries and the downsampling consists of keeping the even-numbered samples and discarding the odd-numbered samples.
2. The segment line starts at an odd-numbered column (1, 3, ...) and has even length. Now the downsampling consists of keeping the odd-numbered samples of the segment line, which are located on the even-numbered columns of the row. Then we retain as much correlation as possible between two successive rows in the subbands, which will improve the decomposition of the columns.
3. The segment line starts at an even-numbered column (0, 2, ...) and has odd length. In this case, the efficient signal extension introduced in Section 3.4 is

used, and the additional sample at the right-hand side in the extension is chosen such that the right-most high-pass subband coefficient becomes zero, i.e., can be neglected for transmission. The downsampling corresponding to this method is the normal downsampling of keeping the even-numbered samples. This means that the low-pass signal is one sample longer than the high-pass signal. This corresponds to the way in which we defined the transferring of the segmentation to the subbands (Figure 7.3).

4. The segment line starts at an odd-numbered column (1, 3, ...) and has odd length. This is the most difficult case. Now, the resulting low-pass signal must be one sample shorter than the high-pass signal. Further, the downsampling has to take place by keeping the odd-numbered samples of the segment line, which are located on even-numbered columns of the row. This implies that the efficient signal extension has to be implemented in such a way that the additional sample in the extension appears at the *left*-hand side. The determination of this extra sample differs from that of the previous case, because now the pre-set sample is positioned in the low-pass subband. It will be clear that the choice 0 will not do, as it will alter the statistics of the subbands in general. Therefore, we define the pre-set sample in the low-pass subband to be equal to its (right) neighbour. This choice is expected to preserve the statistics of both the low-pass and high-pass signal. Then the calculation of the additional boundary sample requires two inner products of the filter with a part of the signal. This procedure is shown in Figure 7.4.

The four classes are illustrated in Figure 7.5. The coefficients in a grey background do not have to be transmitted, i.e., the total number of subband coefficients is equal to the number of pixels in the original signal.

For the columns, the filtering and downsampling procedures are the same. The downsampling depends on the parity of the row number where the vertical segment line starts. Note that the signal extension method can be chosen either periodically or symmetrically and that there are no restrictions on the filter class to be used.

Processing of segment lines of length one

We now have described in general terms the filtering and downsampling of a segmented signal. However, there is one exception and this is the situation of a segment line of length one. It will be clear that the information in such a signal of length one is always of low-pass nature and that it cannot be divided into a low-pass and high-pass component.

Figure 7.4 The low-pass filtering and downsampling in the case of a segment line starting at an odd-numbered column and having an odd length. A symmetric odd-length filter pair and the efficient symmetric extension are used. The signal is represented by 'b' - 'h'. The additional sample 'a' is chosen such that 'k' equals 'l'. Then the sample 'k' does not have to be transmitted. Note that in comparison to Figure 3.4b the downsampling is different, which can be concluded from the shifted position of the filter and the symmetry type in the signal extension of the subband.

If this segment line of length one is situated at an even-numbered position, there is no problem. It must be filtered by the low-pass filter to produce a coefficient in the low-pass subband. Thus, it is extended, filtered, and downsampled in the way defined above, which, of course, amounts to just the copying of the pixel value in the low-pass subband. However, if the segment line of length one is located at an odd-numbered position, it should give a coefficient in the high-pass subband. This coefficient will be zero after the extension, filtering, and downsampling, because of the zero mean of the high-pass filter. As the value of this subband coefficient is independent of the pixel value, we will have lost some information. Our solution to this problem is to keep the pixel value and copy it directly into the high-pass subband. We call these subband coefficients *singles* in the sequel. In other words, singles are low-pass coefficients that are positioned in a high-pass subband.

The singles are handled as if they together form a new region, which consists of non-connected points. They are quantized separately and the variance calculation and bit allocation of the remaining subband coefficients is performed without the singles. This guarantees that the singles are processed in such a way that they do not disturb any statistics of the remaining regions. If the high-pass subband were split up again,

Figure 7.5 Schematic illustration of the four classes of filtering. In each class only the analysis filtering is shown, where H and G are the low-pass and high-pass filter, respectively. The upper line of every block represents the (extended) signal and the lower line the subband. The circles indicate the special samples, whereas the subband coefficients with grey background do not need to be transmitted.

the non-connected region formed by the singles would be processed separately. But in this thesis we do not consider the splitting of high-pass subbands. Note that the separate processing of singles is possible because their location is determined by the segmentation mask, which is known both at the encoder and the decoder.

At this point we meet a difference between row and column processing. We assume that at the encoder first the rows are processed and then the columns. Then, after the row filtering not all segment columns contain low-pass information, because half of the columns are related to the high-pass subband of the row filtering. New singles can only appear when processing the low-pass subband of the row filtering. The coefficients of the row high-pass subbands are all of a high-pass nature and thus a direct copying of a segment part of length one in the column high-pass signal will hardly disturb the statistics of this subband, even when the region is located in an odd pixel position. Therefore, the singles appearing in the HH band are, in fact, singles caused by row filtering only. In Figure 7.6 we show the singles of the segmented image of Figure 7.3c.

Figure 7.6 The singles in the decomposition of Figure 7.3c, indicated by white circles. In this example none of the singles of the row processing falls in the HH band.

The slight asymmetry in the row and column processing has the consequence that the reconstruction filtering at the decoder must be performed in reversed order. In our case, this means that the column filtering at the decoder must precede the row filtering, otherwise perfect reconstruction is not possible.

Figure 7.7 The region-based discrete wavelet transform image coding scheme. The segmentation controls the transform (DWT), the variance calculation (V), and for the low-pass subband also the quantization (Q). The transmitted data consists of the segmentation information, the variances, and the quantizer output.

7.3 The coding scheme

In this section we describe the image coding scheme used in the experiments reported in the following sections. The scheme is shown in Figure 7.7 and it is an extension of the image coding scheme of Section 4.1. At the encoder, the image is segmented into several regions of similar variance and grey-value. The number of regions depends on the image content and is a compromise between accurate segmentation and transmission costs for the contours. The segmentation controls the discrete wavelet transform of the image, the calculation of the variances, and the quantization. At the decoder, the inverse quantization and inverse transform are controlled by the segmentation as well. Therefore, it is important to have a good segmentation and the segmentation information should be protected against channel errors.

The segmentation information consists of the separation lines between the regions. These can be transmitted using a 4-connected chain code that follows the links in the separation lines. An alternative method is to use an 8-connected chain code addressing the edge pixels in the regions. In that case the chain is formed by pixels. Note that now not all regions need to be coded with their own edge pixels, because a separation

line is completely determined by coding only the chain of pixels on one side of it. In our experiments we did not implement a chain coding, but we assumed that the transmission of the segmentation information requires a bit rate of 1.5 bit per link. In Appendix D we show that this bit rate is a fair estimate for a 4-connected chain code.

The bit allocation is performed with the algorithm of [West89], which has been summarized in Section 4.1. The variance of every subband segment is calculated and is used to control the bit allocation process. The variances are transmitted to enable the decoder to reproduce the bit allocation process of the encoder.

The quantization method is PCM for the coefficients in all high-pass subbands. The coefficients of the low-pass subband are DPCM quantized per region. The DPCM is implemented in the following way. The regions are line scanned from top to bottom. The first subband coefficient of a region is PCM quantized with 6 bpp. All other coefficients are predicted with 3, 2, or 1 neighbouring coefficients depending on the number of neighbours in the upper left area of the coefficient. The prediction coefficients are fixed, as opposed to the method used for standard DPCM coding of the low-pass band as in Section 4.1. This avoids the transmission of several prediction coefficients per region. If a subband coefficient has no neighbours, the previous coded sample of the same region is used as prediction. In Figure 7.8 we show the way the fixed prediction is defined.

Figure 7.8 The fixed prediction $\hat{x}$ of the subband coefficients x in the low-pass subband, depending on the number of neighbouring coefficients in the upper left area (denoted by a , b , c) belonging to the same region.

The singles are coded in a special manner. As they are low-pass coefficients, they can easily be predicted from the other low-pass coefficients located in the low-pass subband. We choose to code the singles by the average coefficient value of their corresponding region in the low-pass subband. If this region does not exist, or contains less than 3 coefficients, we apply a PCM quantization with 6 bpp.

The quantizers are uniform threshold quantizers which are designed for generalized Gaussian distributions. As the regions are smaller than the subbands in the standard image coding scheme, the shape parameter tends to a higher value. In Figure 7.9 we show the relation between segment size and shape parameter of the pdf for the Vectra image (Section 7.5). The pdf fitting is based on the minimal Kolmogorov-Smirnov distance, which was evaluated for 13 shape parameters in the range 0.4 – 6.0.

Figure 7.9 Relation between segment size and shape parameter of the pdf, here shown for the high-pass subbands of a 10-band decomposition of the Vectra image (Section 7.5), which is segmented into 52 regions.

The quantizer output is entropy-coded in our implementation. The entropy can be reached by an adaptive variable length coding. For instance, an adaptive arithmetic coding can be used by coding a large group of coefficients simultaneously, where each coefficient has its own subdivision of the $[0, 1]$ interval. This is possible because the pdf models are defined both at the encoder and the decoder.

7.4 Properties, advantages, and disadvantages of the RBDWT

In this section we present an overview of the properties, as well as the expected advantages and disadvantages of the RBDWT in comparison to polynomial coding [Eden86] and the generalized orthogonal transform [Gilg89]. Further, we describe the expected advantages and disadvantages of the complete RBDWT coding scheme

(Section 7.3) compared to the DWT coding scheme used in Chapter 5 and Chapter 6 (Section 4.1). We refer to the DWT of the previous chapters by 'standard' DWT to distinguish it from the DWT as used for the segments.

The RBDWT has the ability to distribute bits over subbands as well as regions, which has been illustrated by Figure 7.1. The standard DWT distinguishes only subbands, whereas polynomial coding has only regions. The generalized orthogonal transform has also the ability to consider frequency information as well as regions.

The segmentation process should be adapted to the method that is used for coding the texture. If the texture is coded by the RBDWT, the regions should be relatively large, because the RBDWT is able to represent multiple edges inside regions very well. When the regions are too small, the RBDWT is of no use, as the transmitted coefficients will only consist of the mean values of the segments. In that case, the advantage of the RBDWT in that it decomposes the signal into different subbands is not used. Contrary to this, the polynomial coding method, for example, is not able to represent many edges or variations inside a region. Therefore, a good segmentation for polynomial coding contains many regions, most of which will have a small size. Of course, such a segmentation causes a high bit rate for the transmission of the contours. The generalized orthogonal transform also requires small segments, because of the computational complexity of the orthogonalization procedure.

Another property of the RBDWT is its ability to support progressive image build-up. For example, this can be applied in image databases. The human observer decides from a full-sized contour image with little texture whether he wishes to see more detail or not. If he is not interested, this progressive image build-up has saved transmission time. Otherwise, additional texture information is used to upgrade the image quality by transmitting more RBDWT coefficients and corrections on coefficients already transmitted. The progressive transmission is also a property of the standard DWT and the method of the generalized orthogonal transform. The polynomial coding has no progressive option, because polynomial coefficients for different image qualities are not related to each other.

Comparing the computational complexity of the RBDWT coding scheme to the standard DWT, the RBDWT has the extra cost of finding a proper segmentation for the image. However, the difference in complexity for the filtering process can be neglected, because only one extra inner product is needed for roughly one fourth of the segment lines and columns (class 4 in Figure 7.5). Further, the solving of the Yule-Walker equations is no longer needed in the DPCM coding of the low-pass sub-

band coefficients. In comparison with the generalized orthogonal transform the RBDWT has a lower complexity. The orthogonalization of the basis images with respect to the shape of a region requires a considerable computational effort that increases with the size of the region.

As a last point we compare the RBDWT with the standard DWT. There are four advantages of the RBDWT. Firstly, after the segment boundaries have been determined, the filtering takes place inside relatively homogeneous regions. Consequently, the high-pass subbands are expected to contain less energy than for the standard DWT, since filtering across sharp edges is avoided. Another advantage of the RBDWT is that the DPCM coding of the low-pass subband has a better prediction, because the spatial correlation inside each region will be higher than the spatial correlation over the low-pass subband as a whole. Therefore, the energy of the prediction error of the low-pass subband will be lower. In the third place, quantization errors have only local effect, because the reconstruction filtering depends only on one region. That is, quantization errors do not influence neighbouring regions and the ringing artefacts near sharp edges will be much less than for standard DWT. Finally, the bit allocation should be able to favour important regions in the image, as bits can be assigned to each subband segment separately.

A disadvantage of the RBDWT in comparison to the standard DWT is the time-consuming generation of the segmentation. Another disadvantage is the lower bit rate available for the transform coefficients because of the need for transmitting segmentation information.

7.5 Experimental comparison of the RBDWT and the standard DWT

In this section we present the coding results of the RBDWT and compare them to the standard DWT. First, the full resolution image coding is considered, and then the multiresolution idea is investigated.

Two test images are used: Camera-man and the first frame of the Vectra video sequence. To facilitate the split-and-merge process with quad-trees we use a 256x256 part of the Vectra image. Both images are selected because of the presence of sharply defined objects, for which region-based coding methods are most suited.

The segmentation of the Camera-man image uses 11 regions, whereas the 256x256 sized part of the Vectra image has been segmented into 52 regions. These segmentations were selected from several different segmentations, obtained by trial

and error using the split-and-merge algorithm of [Stra91]. The choices were made by considering the number of segments and the representation of some objects in the scene. For a complete coding scheme, the segmentation process should be performed automatically, taking into account the texture coding by the RBDWT. However, here we are mainly interested in the evaluation of the texture coding by the RBDWT. For the same reason, we kept the segmentation mask fixed in all experiments.

Figure 7.10 and Figure 7.11 show both images, their segmentation, the 10-band decomposition of the segmentation mask, and the location of the singles.

7.5.1 Full resolution coding of images

We start with the description of the coding scheme settings that we choose for the RBDWT. Then, we examine the variances of the high-pass subbands and the prediction error variance of the low-pass subband in comparison to the standard DWT. Finally, we compare the objective and subjective coding performances of the RBDWT to those of the standard DWT.

Coding scheme settings

For bit rates of 1.0 bpp and higher the differences between the two methods will not be visible. For low bit rates that are only slightly above the number of bits needed for transmitting the segmentation information, the costs of the chain code will completely consume the channel capacity, such that for the texture only one quantized low-pass value per region remains. Therefore, we choose a bit rate of 0.20 bpp for the coding of the texture information of the RBDWT. Then, we make an estimation of the additional bits that are needed to code the contour information and we allow the standard DWT to use a bit rate of 0.20 bpp plus those additional bits. Thus, we obtain a fair comparison of both methods.

The bit rate needed for the coding of the contour information is image dependent. Our segmentation of the Camera-man image into 11 regions consists of 2694 separation links, which is 0.06 bpp if they would be coded with 1.5 bit per link (Section 7.3). The segmentation of the Vectra image into 52 regions consists of 6397 separation links, which would require 0.15 bpp. Therefore, for a fair comparison of the image quality we allow the standard DWT to spend 0.06 bpp and 0.15 bpp more than the RBDWT, for the Camera-man and Vectra image, respectively.

We have also calculated the entropy of the contours of the Camera-man and Vectra images. This entropy is obtained by walking along the segmentation lines, and counting the number of times the direction was straight on, to the left, and to the right.

Figure 7.10 (a) The test image Camera-man; (b) The segmentation of the image into 11 regions; (c) The 10-band decomposition of the segmentation mask; (d) The locations of the 199 singles indicated by black dots.

(a)

(b)

(c)

(d)

Figure 7.11 (a) The 256x256 sized test image Vectra; (b) The segmentation of the image into 52 regions; (c) The 10-band decomposition of the segmentation mask; (d) The locations of the 481 singles indicated by black dots.

Then, the entropy is calculated by $-f_S^2 \log(f_S) - f_L^2 \log(f_L) - f_R^2 \log(f_R)$, where f_L is the frequency of left turns, and so on. For the Camera-man segmentation the entropy is 1.47, which is close to our choice of 1.5 bits per link, whereas the entropy of the Vectra segmentation is only 1.10.

For both coding methods the image is decomposed into 10 bands. The filter bank used is the biorthogonal wavelet filter pair of lengths 9 and 7 [Coh92], and a symmetric boundary extension is applied. The filter pair has emerged in Chapter 5 as one of the better filters for image coding.

For the coding with the RBDWT we adjust the weights of the subbands in the bit allocation. Normally, all subbands are only weighted with the energy of the synthesis filters [Wood92], which ranges from 0.88 to 1.11 for the 10-band decomposition with the selected filter. However, for a better PSNR and visual quality the low-pass subband of the Camera-man and Vectra image are weighted 20 and 30 times higher than the high-pass subbands, respectively. These weights depend on the average size of the subband segments and have been determined experimentally. For the standard DWT the normal weighting of the low-pass subband emerged as best.

Comparison of the variances

Before considering the objective and subjective coding performance we first compare the subband variances. In Section 7.4 we expected a decrease in variance for the RBDWT with respect to standard DWT.

The variances are calculated assuming a zero mean for each high-pass subband and subband segment, which is theoretically correct, because of the zero mean of the high-pass filter. For the low-pass subband and subband segments, we take the variance of the prediction error, again assuming zero mean.

For the RBDWT, we calculate the average variance for every subband. The average variance for subband s_k with size N_k and variance σ_k^2 , which has been divided into r_k segments of size $N_{k,j}$ and variance $\sigma_{k,j}^2$, is calculated by

$$\sum_{j=0}^{r_k-1} (N_{k,j}/N_k) \sigma_{k,j}^2 = \sum_{j=0}^{r_k-1} (N_{k,j}/N_k) \left(\frac{1}{N_{k,j}} \sum_{n=0}^{N_{k,j}-1} (s_{k,n})^2 \right) = \sigma_k^2, \quad (7.1)$$

i.e., just the variance of subband k .

In Table 7.1 we put the average variances of the subbands of the RBDWT and the subband variances of the standard DWT. We clearly see that for the low-pass subband the prediction error variance has decreased enormously, as had been expected. How-

ever, most of the high-pass subbands have an unexpected higher variance (indicated by the italic numbers). The explanation for this effect is most probably that the fourth type of extension reported in Section 7.2.2 is not preserving the statistics, but is increasing the variance instead. This is caused by the choice of the sample 'l' to be equal to 'k' in Figure 7.4, which assumes a small variation near the edge. As most edges are not ideal, this will not always be valid. This effect is inherent to our RBDWT.

Table 7.1 Subband variances for the RBDWT and the DWT

Subband number	Subband variances DWT Camera-man	Subband variances RBDWT Camera-man	Subband variances DWT Vectra	Subband variances RBDWT Vectra
0	509.5	301.2	361.2	82.8
1	105.1	<i>125.1</i>	36.9	17.9
2	65.2	<i>104.6</i>	209.9	53.9
3	28.7	<i>101.6</i>	22.4	21.2
4	112.8	105.8	34.0	17.0
5	48.2	<i>79.1</i>	117.2	63.4
6	22.8	89.9	11.0	23.6
7	69.7	98.8	10.6	<i>21.2</i>
8	70.8	<i>93.6</i>	69.1	<i>108.0</i>
9	12.7	<i>96.0</i>	4.2	27.8

In the following, we compare the objective and subjective quality of the compressed Camera-man and Vectra images, obtained by the RBDWT and the standard DWT.

Comparison of the objective and subjective coding performance

In Figure 7.12a/c and Figure 7.13a/c we compare the RBDWT with the standard DWT for comparable bit rates, when the contour coding is assumed to require 1.5 bits per link. Two other images are added for comparison. Firstly, in Figure 7.12b and Figure 7.13b we added Gaussian noise to the reconstructed image, with a variance that equals a fixed percentage of the variance of the error image (the difference between the original and the reconstructed image). This method has been suggested in [Kunt85], to reduce the artefact inherent to region-based image coding schemes

which is that the regions seem to be artificially coloured ('painted by numbers'). We chose a noise variance percentage of 60. Secondly, the images in Figure 7.12d and Figure 7.13d are coded with standard DWT at a bit rate of 0.20 bpp, which would be a fair comparison to the RBDWT if the contour information would not need to be transmitted.

Note that the face in the Camera-man image is reconstructed very well with the RBDWT method (Figure 7.12a). This accords with our expected advantage of the RBDWT in that it is able to divide its bits not only over the subbands, but also over the spatial domain. Further, the edges are much sharper for the RBDWT and the ringing artefacts are smaller, as had been expected. The standard DWT yields a more natural texture inside the regions.

Four test persons evaluated the images. A series of two images were presented and each time they had to choose the one they liked most. If they had no preference at all, they were allowed to make no choice.

Two of the test persons appreciated the noise addition in the images. A cautious conclusion is that if noise is added, the variance should be a small percentage of the variance of the difference image. Further, all test persons ranked the RBDWT images above the images coded with standard DWT at 0.20 bpp.

The most important comparison is the comparison of the RBDWT with the standard DWT at a corresponding bit rate, i.e., 0.26 bpp and 0.35 bpp for the Camera-man and Vectra image, respectively. Both pairs of images were presented twice in the series. The RBDWT coded Camera-man image was twice preferred by all test persons over the standard DWT coded image. However, for the Vectra image two persons chose the standard DWT once. For the remaining choices, the RBDWT coded image was ranked better.

Of course, this experiment does not allow us to reach firm conclusions. However, we may say that the RBDWT is a competitive alternative to the standard DWT in terms of subjective visual quality. The sharpness of the edges and the small ringing artefacts are often considered more important than the lack of texture in some segments.

For the sake of completeness, we show in Figure 7.14 both images coded with RBDWT, where the low-pass subband is weighted normally. Clearly, the low-pass subbands get too few bits and this is caused by the extremely good prediction of the DPCM inside the segments. The prediction error variances are then so low that the segments in the low-pass subband do not get enough bits.

(a)

(b)

(c)

(d)

Figure 7.12 Comparison of the coding performance of RBDWT versus DWT. (a) The Camera-man image coded with RBDWT at 0.20 bpp (without counting segmentation overhead of ca. 0.06 bpp); (b) The same image with noise added; (c) and (d) The image coded with the standard DWT at 0.26 and 0.20 bpp, respectively.

Figure 7.13 Comparison of the coding performance of RBDWT versus DWT. (a) The Vectra image coded with RBDWT at 0.20 bpp (without counting segmentation overhead of ca. 0.15 bpp); (b) The same image with noise added; (c) and (d) The image coded with the standard DWT at 0.35 and 0.20 bpp, respectively.

Figure 7.14 The test images coded with the RBDWT coding method, where the low-pass subband is weighted equally in respect to the high-pass subbands.

For comparison of the objective coding performance we put in Table 7.2 the PSNR of the compressed images of Figure 7.12, Figure 7.13, and Figure 7.14. In all cases the standard DWT has a better PSNR, and the weighting of the low-pass subband for the RBDWT generates a higher PSNR than the normal weighting.

Table 7.2 Objective coding performance of RBDWT and standard DWT

PSNR (dB)	Texture of RBDWT (0.20 bpp), weighted low-pass subband	Texture of RBDWT (0.20 bpp), non-weighted low-pass subband	Standard DWT (0.26 bpp / 0.35 bpp)	Standard DWT (0.20 bpp)
Camera-man	25.2	24.5	26.4	25.4
Vectra	28.2	27.8	29.3	27.2

7.5.2 Multiresolution coding of images

In some applications, it is a useful feature to build up the image hierarchically using several resolutions. First, a low-resolution image is produced, and when a high-resolution is required, additional information is added to reconstruct the image at medium

or full resolution. In this section, we compare the capability of both the RBDWT and the DWT to perform such a multiresolution coding, which is useful for the compatible coding of various image formats.

We show in Figure 7.15 and Figure 7.16 a straightforward implementation: the original image is decomposed into 10 subbands and only the four subbands of level 3 are transmitted at the first stage. We coded this first stage with 0.07 bpp and did not take into account the coding of the segmentation information. For the RBDWT, the reconstructed low-resolution image is shown in Figure 7.15a and Figure 7.16a. We clearly see a blocking effect, because of the downsampling of the segmentation. The image coded with standard DWT does not suffer from this type of error (Figure 7.15b and Figure 7.16b). It shows a normal smoothing of all edges and it has the usual low-pass features. The images coded by the standard DWT are usually appreciated more. It is to be expected that for larger size images the downsampled segmentation will have less effect on the visual quality, which is of importance for HDTV applications.

Figure 7.15 Reconstructed Camera-man images using only the low-pass subband of a 10-band decomposition, coded at 0.06 bpp. The images have been enlarged by factor 4. (a) RBDWT; (b) DWT.

For the comparison above, we neglected the coding of the segmentation information. However, we need in a hierarchical scheme only the segmentation of the subbands that are transmitted. Therefore, it would be logical to transmit the segmentation per subband, instead of the segmentation of the total image. As the segmentation

Figure 7.16 Reconstructed Vectra images using only the low-pass subband of a 10-band decomposition, coded at 0.06 bpp. The images have been enlarged by factor 4. (a) RBDWT; (b) DWT.

information consists of (one-dimensional) segmentation lines, the chain codes of all four child subbands in a splitting are roughly *half* the size of the chains codes of the parent subband. This causes a higher total bit rate for the full resolution image. How this increase in total segmentation information can be diminished by using the correlation between the segmentations of the four subbands is an interesting question.

Our overall conclusion is that the multiresolution coding with the RBDWT coding scheme is, at first sight, not competitive to the multiresolution coding with the standard DWT. The downsampling of the segmentation causes blocking effects, which grow with the decrease in resolution.

7.6 Application to video coding

The RBDWT can be applied to video coding in several ways. In this section we mention three possibilities and discuss whether these are promising for further research.

The first, trivial, method is to code all frames independently and then the RBDWT can be applied as before. A drawback is that for every frame the segmentation information has to be transmitted, which is quite expensive. As no temporal correlation is used at all, this method will not be optimal. Further, the differences in the segmenta-

tion from frame to frame will generate a jerky scene. Therefore, this method is not recommended.

A better method is to estimate the motion of the segments in the scene. If the moving objects are small, probably large parts of the background segmentation will be equal to the segmentation in the previous frame. Thus, the segmentation transmission cost per frame is expected to decrease, because only the modifications to the segmentation have to be transmitted. Research can be directed to find the matching of the segmentations of two successive frames and to code the differences efficiently.

In this approach, the coding of texture information in the regions with RBDWT can be performed in two ways. Firstly, we make use of the temporal correlation before applying the RBDWT, i.e., the same order of removing temporal and spatial correlation as for scheme I, shown in Section 4.2. Image regions that have a corresponding (part of a) region in the previous frame can be predicted by the corresponding region (parts) of the previous frame. Then, the prediction error as well as the non-predicted areas can be coded by the RBDWT. Secondly, we can think of using the temporal correlation after applying the RBDWT, as in scheme II and III explained in Section 4.2. However, the motion estimation between the coefficients of subband segments is even more complex than for subbands as a whole. The reason is that the shapes of the segments depend on the position of the corresponding region in the frame (compare the different decompositions of the two central segments in Figure 7.3). Because the usual aliasing problems of motion estimation in subbands also show up, this type of coding scheme is not expected to be useful. The first method of coding the texture is expected to be the most promising. Further research must indicate whether it is competitive with existing video coding schemes or not.

A third method has been suggested in [Musm89] and used in [Gilg89] for the application of the generalized orthogonal transform to video coding, but it can also be used for the RBDWT. A new frame is predicted from the previous one in exactly the same way as in scheme I shown in Figure 4.5, thus without using any segmentation information. From the prediction error image the areas are determined for which the motion compensation has failed. The pixels that are masked by these areas have a high-pass characteristic, i.e., the correlation between the pixels is rather low and intra-frame coding methods like the RBDWT are not useful. Instead, the corresponding regions in the original frame are coded by the RBDWT.

From the three possible methods mentioned above, the third method seems to be the most promising, since it does not make any assumptions about the correspondence between the segmentations of two consecutive frames.

7.7 Conclusions and further research

In this section, we have presented a new texture coding method for contour-texture coding techniques which is called the region-based discrete wavelet transform (RBDWT). It has the advantage that it is able to distribute the available bit rate over subbands as well as regions. We have presented a coding scheme, in which all parts have been specified except for the segmentation, which has been chosen fixed for the two test images.

For image compression, the subjective and objective coding performance of the RBDWT have been compared with the standard DWT. Although the standard DWT has a higher PSNR than the RBDWT, the subjective coding performance of the RBDWT has emerged to be quite competitive. In particular, the edges are sharper and ringing artefacts are smaller. For multiresolution coding, the low-resolution image has shown a lower subjectively rated visual quality for the RBDWT than for the standard DWT. For the application of the RBDWT to video coding, the most promising option seems to be the coding of those areas for which motion compensation has failed.

Further research should be directed to the segmentation process, taking into account the characteristics of the RBDWT method for texture coding. Then a complete coding scheme can be built, the coding performance of which should be compared to other contour-texture coding techniques. Another part of the coding scheme that deserves more attention is the quantization of the subband coefficients. Some gain is to be expected by using another pdf model for small-sized regions. Another research area is the development of an efficient hierarchical transmission of the segmentation information, which makes the coding of the subbands less dependent on the segmentation of the original image. Finally, the possible application of the RBDWT in a video coding scheme deserves further research.

Bibliography

- [Abra65] M. Abramowitz, and I. A. Stegun (Eds.), *Handbook of Mathematical Functions*, Dover Publications Inc., New York, 1965.
- [Akan92] A. N. Akansu, and R. Haddad, *Multiresolution Signal Decomposition*, Academic Press Inc., Boston, 1992.
- [Ante92] J.- P. Antoine, "Wavelets in Image Processing," *Signal Processing VI: Theories and Applications*, pp. 23-30, Elsevier, 1992.
- [Ante93] J.-P. Antoine, P. Carrette, R. Murenzi, and B. Piette, "Image Analysis with Two-Dimensional Continuous Wavelet Transform," *Signal Processing*, Vol. 31, No. 3, pp. 241-272, April 1993.
- [Anto90] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding Using Vector Quantization in the Wavelet Transform Domain," *Proceedings ICASSP 1990*, pp. 2297-2300, April 1990.
- [Anto91] M. Antonini, *Transformée en ondelettes et compression numerique des images*, Ph. D. thesis, Université de Nice-Sophia Antipolis, 1991.
- [Anto92] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding Using Wavelet Transform," *IEEE Transactions on Image Processing*, Vol. 1, No. 2, pp. 205-220, April 1992.

- [Asla68] E. W. Aslaksen, and J. R. Klauder, "Unitary representations of the affine group," *J. Math. Phys.*, Vol. 9, pp. 206-211, 1968.
- [Barn92] H. J. Barnard, B. Sankur, and J. C. A. van der Lubbe, "Statistics of DCT Coefficients in a Hybrid Video Codec," *Proceedings of the Thirteenth Symposium on Information Theory in the Benelux*, pp. 39-46, June 1992.
- [Barn93] H. J. Barnard, J. H. Weber, and J. Biemond, "Efficient Signal Extension for Subband/Wavelet Decomposition of Arbitrary Length Signals," *SPIE Vol. 2094 Visual Communications and Image Processing '93*, pp. 966-975, November 1993.
- [Barn94] H. J. Barnard, J. H. Weber, and J. Biemond, "A Region-Based Discrete Wavelet Transform for Image Coding," To appear in: *Signal Processing VII: Theories and Applications (Proceedings of Eusipco '94)*, Elsevier Science Publishers B.V., September 1994.
- [Blan88] L. Blanc-Feraud, M. Barlaud, and P. Mathieu, "Amélioration de la restauration d'images floues par un filtrage adaptif de Kalman utilisant une image miroir," *Traitement du Signal*, Vol. 5, No. 4, pp. 249-261, 1988.
- [Bosv89] F. Bosveld, *Hierarchisch subband coderen van HDTV in B-ISDN*, M. Sc. thesis, Delft University of Technology, Department of Electrical Engineering, Information Theory Group, Delft, The Netherlands, November 1989, (in Dutch).
- [Bosv94] F. Bosveld, *Hierarchical Video Compression*, Ph. D. thesis, Delft University of Technology, Department of Electrical Engineering, Information Theory Group, Delft, The Netherlands, to appear June 1994.
- [Bouc92] F. Boucherok, and J.-F. Vial, "Compatible Multi-Resolution Coding Scheme," *Proceedings of the International Workshop on HDTV '92*, November 1992.

-
- [Burt83] P. Burt, and E. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Transactions on Communication*, Vol. 31, pp. 482-540, 1983.
- [CCIT90] CCITT, "Draft Revision of Recommendation H.261: Video Codec for Audiovisual Services at px64 kbit/s," *Image Communication*, Vol. 2, No. 2, pp. 221-239, August 1990.
- [Chui92a] C. K. Chui, *An Introduction to Wavelets*, Academic Press Inc., Boston, 1992.
- [Chui92b] C. K. Chui, editor, *Wavelets: A Tutorial in Theory and Applications*, Academic Press Inc., Boston, 1992.
- [Cohe92] A. Cohen, I. Daubechies, and J. -C. Feauveau, "Biorthogonal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. XLV, pp. 485-560, 1992.
- [Cohe93a] A. Cohen, and I. Daubechies, "Non-Separable Bidimensional Wavelet Bases," *Revista Matemática Iberoamericana*, Vol. 9, No. 1, pp. 51-137, 1993.
- [Cohe93b] A. Cohen, I. Daubechies, and P. Vial, "Wavelets on the Interval and Fast Wavelet Transforms," *Applied and Computational Harmonic Analysis*, Vol. 1, No. 1, December 1993.
- [COST93] COST, *Proceedings of the COST 211^{ter} European Workshop on New Techniques for Coding of Video Signals at Very Low Bitrates*, University of Hannover, Germany, December 1993.
- [Croi76] A. Croisier, D. Esteban, and C. Galand, "Perfect Channel Splitting by Use of Interpolation/Decimation/Tree Decomposition Techniques," *Proceedings of the International Conference on Information Science and Systems*, Patras, Greece, pp.443-446, August 1976.
- [Daub88] I. Daubechies, "Orthonormal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. XLI, pp. 909-996, 1988.

- [Daub90] I. Daubechies, "The Wavelet Transform, Time-Frequency Localization and Signal Analysis," *IEEE Transactions on Information Theory*, Vol. 36, No. 5, pp. 961-1005, September 1990.
- [Drie92] J. N. Driessen, *Motion Estimation for Digital Video*, Ph. D. thesis, Delft University of Technology, Department of Electrical Engineering, Information Theory Group, Delft, The Netherlands, September 1992.
- [Eden85] M. Eden, and M. Kocher, "On the Performance of A Contour Coding Algorithm in the Context of Image Coding; Part 1: Contour Segment Coding," *Signal Processing*, Vol. 8, pp. 381-386, 1985.
- [Eden86] M. Eden, M. Unser, and R. Leonardi, "Polynomial Representation of Pictures," *Signal Processing*, Vol. 10, No. 4, pp. 385-393, 1986.
- [Egge86] J. D. Eggerton, and M. D. Srinath, "Statistical Distributions of Image DCT coefficients," *Comput. & Elect. Engng.*, Vol. 12, No. 3/4, pp. 137-145, 1986.
- [Feau90] J. -C. Feauveau, "Analyse multirésolution pour les images avec un facteur de résolution $\sqrt{2}$," *Traitement du Signal*, Vol. 7, No. 2, pp. 117-128, 1990.
- [Free61] H. Freeman, "On the Encoding of Arbitrary Geometric Configurations," *IRE Transactions on Electronic Computers*, Vol. EC-10, No. 2, pp. 260-268, 1961.
- [Gabo46] D. Gabor, "Theory of Communication," *J. Inst. Elect. Eng. (London)*, Vol. 93, No. III, pp. 429-457, 1946.
- [Gaid92] T. Gaidon, M. Barlaud, and P. Mathieu, "Image Sequence Coding Using Quincunx Wavelet Transform, Motion Compensation and Lattice Vector Quantization," *SPIE Vol. 1818 Visual Communications and Image Processing '92*, pp. 241-252, 1992.

-
- [Ghar91] H. Gharavi, "Subband Coding Algorithms for Video Applications: Videophone to HDTV-Conferencing," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 1, No. 2, pp. 174-183, June 1991.
- [Gilg89] M. Gilge, T. Engelhardt, and R. Mehlan, "Coding of Arbitrarily Shaped Image Segments Based on a Generalized Orthogonal Transform," *Signal Processing: Image Communication*, Vol. 1, No. 2, pp. 153-180, October 1989.
- [Groc92] K. Gröchenig, and W. R. Madych, "Multiresolution Analysis, Haar Bases, and Self-Similar Tilings of $\mathbb{R}^n$," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, pp. 556-568, March 1992.
- [Gros84] A. Grossmann, and J. Morlet, "Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape," *SIAM Journal of Mathematical Analysis*, Vol. 15, pp. 723-736, 1984.
- [Haar10] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," *Math. Ann.*, Vol. 69, pp. 331-371, 1910.
- [Hart93] F. Hartung, and J. H. Husøy, "Wavelet and Subband Coding of Images - A Comparative Study," *SPIE International Symposium on Optical and Applied Science and Engineering*, San Diego, CA, July 1993.
- [ISO93] ISO/IEC JTC1/SC29, *Proposal for a New Work Item: Very-Low Bitrate Audio-Visual Coding*, March 1993.
- [Jawe93] B. Jawerth, and W. Sweldens, *An Overview of Wavelet Based Multiresolution Analyses*, Research Report 1993:1, Industrial Mathematics Initiative, Dept. of Mathematics, University of South Carolina, February 1993.
- [Jaya84] N. S. Jayant, and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, 1984.

- [John80] J. D. Johnston, "A Filter Family Designed for Use in Quadrature Mirror Filter Banks," *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 291-294, April 1980.
- [Karl87] G. Karlsson, and M. Vetterli, "Sub-Band Coding of Video Signals for Packet-Switched Networks," *SPIE Vol. 845 Visual Communications and Image Processing II*, pp. 446-456, 1987.
- [Koch86] M. Kocher, and R. Leonardi, "Adaptive Region Growing Technique Using Polynomial Functions for Image Applications," *Signal Processing*, Vol. 11, No. 1, pp. 47-60, July 1986.
- [Kova92] J. Kovačević, and M. Vetterli, "Nonseparable Multidimensional Perfect reconstruction Filter Banks and Wavelet Bases for R^n ," *IEEE Transactions on Information Theory*, Vol. 38, No. 2, pp. 533-555, March 1992.
- [Kova93] J. Kovačević, and M. Vetterli, "Nonseparable Two-Dimensional Orthogonal Wavelets with Symmetries," *Proceedings of the Eighth Workshop on Image and Multidimensional Signal Processing*, Cannes, pp. 150-151, September 1993.
- [Kron89] T. Kronander, *Some Aspects of Perception Based Image Coding*, Ph. D. thesis, Linköping University, Linköping, Sweden, 1989.
- [Kunt85] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-Generation Image Coding Techniques," *Proceedings of the IEEE*, Vol. 73, No. 4, pp. 549-574, April 1985.
- [Kunt87] M. Kunt, M. Bénard, and R. Leonardi, "Recent Results in High-Compression Image Coding," *IEEE Transactions on Circuits and Systems*, Vol. CAS-34, No. 11, pp. 1306-1336, November 1987.
- [Mall89a] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 674-693, July 1989.

-
- [Mall89b] S. G. Mallat, "Multiresolution Approximations and Wavelet Orthonormal Bases of $L_2(\mathbb{R})$," *Transactions of the American Mathematical Society*, Vol. 315, No. 1, pp. 69-87, September 1989.
- [Mall89c] S. G. Mallat, "Multifrequency Channel Decomposition of Images and Wavelet Models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 37, No. 12, pp. 2091-2110, December 1989.
- [Marr82] D. Marr, *Vision*, Freeman, New York, 1982.
- [Mart91] S. A. Martucci, "Signal Extension and Noncausal Filtering for Sub-band Coding of Images," *SPIE Vol. 1605 Visual Communications and Image Processing '91: Visual Communication*, pp. 137-148, 1991.
- [McCl73] J. H. McClellan, "The Design of Two-Dimensional digital filters by transformations," *Proceedings of the 7th Annual Princeton Conference on Information Sciences and Systems*, pp. 247-251, 1973.
- [Mey85] Y. Meyer, "Principe d'incertitude, bases hilbertiennes et algebres d'operateurs," *Bourbaki seminar*, No. 662, 1985-1986.
- [Mey90] Y. Meyer, *Ondelettes et Operateurs*, Vol. 1. Hermann, Paris, 1990.
- [Mey93] Y. Meyer, *Wavelets - Applications and Algorithms*, SIAM, Philadelphia, 1993.
- [MPEG-1] ISO/IEC DIS 11172, *Information Technology - Coding of Moving Pictures and Associated Audio for Digital Storage Media up to about 1.5 Mbit/s*, October 1992.
- [MPEG-2] ISO/IEC 13818, Recommendation H.262, *Generic Coding of Moving Pictures and Associated Audio*, Committee Draft, November 1993.
- [Musm89] H. G. Musmann, M. Hötter, and J. Ostermann, "Object-Oriented Analysis-Synthesis Coding of Moving Images," *Signal Processing: Image Communication*, Vol. 1, No. 2, pp. 117-138, 1989.

- [Nave92] T. Naveen, J.W. Woods, "Motion Compensated Multiresolution Transmission of High Definition Video," *Submitted to IEEE Transactions on Circuits and Systems for Video Technology*, 1992.
- [Nguy89] T. Q. Nguyen, and P. P. Vaidyanathan, "Two-Channel Perfect-Reconstruction FIR QMF Structures which yield Linear-Phase Analysis and Synthesis Filters," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 37, pp. 676-690, May 1989.
- [Ohm92] J. R. Ohm, "Temporal Domain Sub-Band Video Coding with Motion Compensation," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. III, pp. 229-232, March 1992.
- [Onno93] P. Onno, and C. Guillemot, "Tradeoffs in the Design of Wavelet Filters for Image Compression," *SPIE Vol. 2094 Visual Communications and Image Processing '93*, pp. 1536-1547, November 1993.
- [Oppe83] A. V. Oppenheim, A. S. Willsky, and I. T. Young, *Signals and Systems*, Prentice-Hall, London, 1983.
- [Penn93] W. B. Pennebaker, and J. L. Mitchell, *JPEG - Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, 1993.
- [Pres92] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in {C}: The Art of Scientific Computing*, Second edition, Cambridge, 1992.
- [Rein83] R. C. Reininger, and J. D. Gibson, "Distributions of the Two-Dimensional DCT-Coefficients for Images," *IEEE Transactions on Communications*, Vol. COM-31, No. 6, pp. 835-839, June 1983.
- [Riou91] O. Rioul, and M. Vetterli, "Wavelets and Signal Processing," *IEEE Signal Processing Magazine*, pp. 14-38, October 1991.
- [Riou92a] O. Rioul, "Simple, Optimal Regularity Estimates for Wavelets," *Signal Processing VI: Theories and Applications (Proceedings of Eusipco '92)*, Elsevier Science Publishers B.V., 1992.

-
- [Riou92b] O. Rioul, "Simple Regularity Criteria for Subdivision Schemes," *SIAM Journal of Mathematical Analysis*, Vol. 23, No. 6, pp. 1544-1576, November 1992.
- [Sank91] B. Sankur, private communication, 1991.
- [Smit86] M. J. T. Smith, and T. P. Barnwell, "Exact Reconstruction for Tree-Structured Subband Coders," *IEEE Transactions on Acoustics Speech and Signal Processing*, Vol. ASSP-34, pp. 434-441, June 1986.
- [Stra91] K. C. Strasters, and J. J. Gerbrands, "Three-Dimensional Image Segmentation Using a Split, Merge and Group Approach," *Pattern Recognition Letters*, Vol. 12, No. 5, pp. 307-325, May 1991.
- [Tong81] G. J. Tonge, *The Sampling of Television Images*, Experimental and Development Report 112/81 of the Independent Broadcasting Authority, Winchester, May 1981.
- [Vaid93] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, 1993.
- [Vett84] M. Vetterli, "Multi-Dimensional Sub-Band Coding: Some Theory and Algorithms," *Signal Processing*, Vol. 6, No. 2, pp. 97-112, April 1984.
- [Vett86] M. Vetterli, "Filter Banks Allowing Perfect Reconstruction," *Signal Processing*, Vol. 10, pp. 219-244, 1986.
- [Vett92] M. Vetterli, and C. Herley, "Wavelets and Filter Banks: Theory and Design," *IEEE Transactions on Signal Processing*, Vol. 40, No. 9, pp. 2207-2232, September 1992.
- [Vial92] J.-F. Vial, M. Pécot, P. J. Tourtier, and Y. Thomas, "In-Band Interpolation Applied To Motion-Compensated Subband Coding," in: *Signal Processing of HDTV, III*, H. Yasuda and L. Chiariglione (Eds.), Elsevier, pp. 207-214, 1992.
- [Vill93] L. F. Villemoes, "Continuity of Non-Separable Quincunx Wavelets," *submitted to Appl. Comp. Harmonic Anal.*, 1993.

- [Visc91] E. Viscito, and J. P. Allebach, "The Analysis and Design of Multidimensional FIR Perfect Reconstruction Filter Banks for Arbitrary Sampling Lattices," *IEEE Transactions on Circuits and Systems*, Vol. CAS-38, No. 1, January 1991.
- [West89] P. H. Westerink, *Subband Coding of Images*, Ph. D. thesis, Delft University of Technology, Department of Electrical Engineering, Information Theory Group, Delft, The Netherlands, October 1989.
- [Wick92] M. V. Wickerhauser, "Acoustic Signal Compression with Wavelet Packets," in [Chui92b], pp. 679-700, 1992.
- [Wood86] J. W. Woods, and S. D. O'Neil, "Subband Coding of Images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-34, No. 5, pp. 1278-1288, October 1986.
- [Wood92] J. W. Woods, and T. Naveen, "A Filter Based Bit Allocation Scheme for Subband Compression of HDTV," *IEEE Transactions on Image Processing*, Vol. 1, No. 3, pp. 436-440, July 1992.
- [Zett90] W. R. Zettler, J. Huffman, and D. C. P. Linden, "Application of Compactly Supported Wavelets to Image Compression," *SPIE Vol. 1244 Image Processing Algorithms and Techniques*, pp. 150-160, 1990.
- [Zhan92] Y.-Q. Zhang, S. Zafar, "Motion-Compensated Wavelet transform Coding for Color Video Compression", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 2, No. 3, September 1992.

Appendix A

Overview of Wavelet Transforms and Series

In this appendix we present all formulas for the wavelet transforms and series. The formulas are classified and named analogous to the formulas of Fourier theory as used in [Oppe83]. The corresponding formulas of Fourier and wavelet theory are shown in juxtaposition in two tables, thus producing a clear overview. After giving the overview in Section A.1, we summarize in Section A.2 the derivation of the 2-D continuous-time wavelet transform, which has not been discussed in Chapter 6.

A.1 Overview of the formulas for wavelet and Fourier transforms and series

In Table A.1 and Table A.2 we present an overview of the formulas for wavelet and Fourier transforms and series for continuous and discrete signals, respectively. The Fourier theory formulas are placed on the left side and those of the wavelet theory on the right side. Each time both the transform and its inverse are given and we assume that the signals fulfil the necessary requirements. We start with an explanation of the notation used and conclude with some comments on the tables.

As in Chapter 2, continuous signals are represented in function notation with the argument between parentheses, whereas discrete signals are written in index notation. Whenever the domain of the argument or index is not given, it is assumed to be R and Z , respectively. The symbol N denotes the length of the finite discrete signal, and is

assumed to be a power of 2 for the DTWS and DWT. The notation $\langle N \rangle$ is used to indicate a range of N consecutive integers and, analogously, 2π below an integral means integrating over an arbitrary interval of length 2π .

An infinite length input signal is denoted either by $f(x)$ or c_k . If a finite length signal is made periodic we denote it by $\tilde{c}_k$, where the period is equal to the length of the finite signal, i.e., $\tilde{c}_{k+rN} = c_k$, $r \in \mathbb{Z}$.

There are no formulas given for the DTWT. In other words, there is no wavelet analog to the DTFT that transforms a discrete signal into a continuous transform domain.

Another remark is that, in the comparison, we put the constants of the transform pairs according to the positions as appear in Chapter 2. This means that the Fourier theory constants appear in one of the two formulas of a pair (as is usual in signal processing), whereas the constants for the wavelet formulas are divided equally over both formulas in a pair (as is usual in mathematics). Of course, this is only a matter of definition.

The DWT is defined for finite length signals, but it needs a periodic signal in the implementation. Our formulas correspond to the periodic extension of Section 3.3. As we assume the signal to be of length $N = 2^d$, the decomposition is complete and there is one low-pass coefficient, usually called the DC value. In the inverse formula this DC value does not need to weight a basis function, because a constant signal appears when a signal with length one is made periodic.

Finally, note that the formulas for the DTWS and DWT are indexed by $k - 2^m n$. In fact, these indices should be $2^{-m}k - n$ in order to have a full analogy to the formulas of continuous theory. However, to get the common integer indexing of discrete signals, we multiplied the indices by 2^m .

Table A.1 Comparison between the continuous versions of the Fourier transform and series with the wavelet transform and series

Continuous	Fourier	Wavelet
Transform	$CTFT\{f(\omega) = \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx$	$CTWTF(a, b) = \int_{-\infty}^{\infty} a^{-1/2} \psi\left(\frac{x-b}{a}\right) f(x) dx, \quad a > 0$
Inverse	$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} CTFT\{f(\omega) e^{i\omega x} d\omega$	$f(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} CTWTF(a, b) a^{-1/2} \psi\left(\frac{x-b}{a}\right) \frac{da db}{a^2}$
Series	$(CTFS)_p{}_n = \frac{\omega_0}{2\pi} \int_{2\pi/\omega_0} f(x) e^{-in\omega_0 x} dx$	$(CTWS)_p{}_m, n = \int_{-\infty}^{\infty} a_0^{-m/2} \psi(a_0^{-m} x - nb_0) f(x) dx$
Inverse	$f(x) = \sum_{n=-\infty}^{\infty} (CTFS)_p{}_n e^{in\omega_0 x}$	$f(x) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} (CTWS)_p{}_m, n a_0^{-m/2} \psi(a_0^{-m} x - nb_0)$

Table A.2 Comparison between the discrete versions of the Fourier transform and series and the wavelet transform and series

Discrete	Fourier	Wavelet
Transform	$DTFT_c(\omega) = \sum_{k=-\infty}^{\infty} c_k e^{-i\omega k}$	(not defined)
Inverse	$c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} DTFT_c(\omega) e^{i\omega k} d\omega$	(not defined)
Series	$(DTFS_c)_n = \frac{1}{N} \sum_{k=\langle N \rangle} c_k e^{-in(2\pi/N)k}$	$(DTWS_c)_{m,n} = \sum_{k=-\infty}^{\infty} 2^{m/2} g_{k-2^m n} c_k, \quad m \geq 0$
Inverse	$c_k = \sum_{n=\langle N \rangle} (DTFS_c)_n e^{in(2\pi/N)k}$	$c_k = \sum_{m=0}^{\infty} \sum_{n=-\infty}^{\infty} 2^{m/2} (DTWS_c)_{m,n} g_{k-2^m n}^m$
Finite Transform	$(DFT_c)_n = \frac{1}{N} \sum_{k=0}^{N-1} c_k e^{-in(2\pi/N)k}$	$(DWT_c)_{m,n} = \sum_{k=-\infty}^{\infty} 2^{m/2} g_{k-2^m n}^m \tilde{c}_k, \quad \begin{cases} 1 \leq m < {}^2\log N \\ 0 \leq n < N2^{-m} \end{cases}$ $(DWT_c)_{m,n} = \sum_{k=-\infty}^{\infty} 2^{m/2} h_{k-2^m n}^m \tilde{c}_k, \quad \begin{cases} m = {}^2\log N \\ n = 0 \end{cases}$
Inverse	$c_k = \sum_{n=0}^{N-1} (DFT_c)_n e^{in(2\pi/N)k}$	$c_k = N^{1/2} (DWT_c)_{\lceil \log(N), 0} + \sum_{m=1}^{{}^2\log(N)-1} \sum_{n=0}^{N2^{-m}-1} 2^{m/2} (D\tilde{W}T_c)_{m,n} g_{k-2^m n}^m$

A.2 The two-dimensional continuous-time wavelet transform

In this section we consider the extension of the continuous-time wavelet transform to two dimensions [Ante92][Ante93]. The 1-D CTWT transforms a one-dimensional signal into a two-dimensional signal, with parameters a for scale and b for location. In analogy, the 2-D CTWT creates a four-dimensional signal and the basis functions will have four parameters. Two parameters are for the location and are denoted by $\mathbf{b}$, which is a vector of length two. The third parameter is the scale parameter a , which is assumed to be the same in all directions. The last parameter is for the orientation of the basis functions and is denoted by the angle θ . We may consider a and θ as the spatial frequency in polar coordinates. Now all basis functions are shifted, dilated and rotated versions of one function. With the definition

$$\mathbf{R}_\theta = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (\text{A.1})$$

the basis functions become

$$\psi_{a, \mathbf{b}, \theta}(\mathbf{x}) = a^{-1} \psi((a\mathbf{R}_\theta)^{-1}(\mathbf{x} - \mathbf{b})), \quad (\text{A.2})$$

where the wavelet $\psi(\mathbf{x})$ is a function of two-dimensional variable $\mathbf{x}$. Equation (A.2) is the 2-D analog to (2.3). The amplitude correction is in fact $(a^N)^{-1/2}$, with N the dimension of the space. These wavelets are basis functions for the CTWT and they form a redundant set. If the wavelets have no rotational symmetry then it is possible to analyse signals with respect to directions.

The CTWT of a function $f(\mathbf{x})$, $\mathbf{x} \in R^2$ is defined as

$$CTWT(a, \mathbf{b}, \theta) = \langle \psi_{a, \mathbf{b}, \theta}(\mathbf{x}), f(\mathbf{x}) \rangle = a^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi((a\mathbf{R}_\theta)^{-1}(\mathbf{x} - \mathbf{b})) f(\mathbf{x}) d^2\mathbf{x}, \quad (\text{A.3})$$

where $d^2\mathbf{x} = dx dy$. The function $f(\mathbf{x})$ can be reconstructed by calculating the inverse,

$$f(\mathbf{x}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_0^{2\pi} \frac{1}{a} \psi((a\mathbf{R}_\theta)^{-1}(\mathbf{x} - \mathbf{b})) CTWT(a, \mathbf{b}, \theta) \frac{d\theta da d^2\mathbf{b}}{a^2}, \quad (\text{A.4})$$

where we have assumed that an admissibility condition analogous to (2.5) is fulfilled and that the functions $\psi_{a,b,\theta}(x)$ form an orthogonal set. The factor $1/a^2$ is the Jacobian.

In [Ante92][Ante93] this 2-D CTWT is applied to discrete signals by transforming the discrete signal into (continuous) Fourier domain and sampling the 2-D CTWT so fine as to guarantee an accurate result. As in these references the subject is signal analysis, it is not important that there is a lot of redundancy in the transform.

Appendix B

Linear Phase and Biorthogonality

Filters for two-band splitting schemes cannot be orthogonal and linear phase simultaneously when the filters have length greater than 2. If the orthogonality requirement is replaced by the weaker biorthogonality requirement, then linear phase filters can be constructed. In this appendix, we show in Section B.1 why the property of linear phase is important in image coding. Section B.2 contains the formulas for the biorthogonal extension to the wavelet theory. The derivations are analogous to those of Section 2.2, and therefore we limit ourselves to the formulas only.

B.1 The importance of linear phase for image coding

In image coding applications, it is important that the filters used are of linear phase. If they are not of linear phase, the quantization of the subband coefficients not only introduces amplitude distortion, but also phase distortion. In Figure B.1 we show schematically why a phase distortion is more important than an amplitude distortion. The first graph displays a sum of sines, and in the second graph the amplitude of one of the sines is distorted. It can be observed that the distortion has hardly any influence on the overall shape of the total signal. In the third graph, one of the signals has a phase distortion instead of an amplitude distortion. Clearly, the degradation of the sum of sines is more severe, as the overall shape of the signal has changed [Sank91].

The importance of the phase information is also clear from Figure 4.35 in [Oppe83]. This figure shows what happens when the amplitude and the phase of two

Figure B.1 The importance of phase in comparison to amplitude. (a) The starting function, which is the sum of the three grey sines; (b) The sum, after one of the amplitudes has been distorted by factor 1.5; (c) The sum, after one of the sines has been shifted over $2\pi/5$.

different images are combined and together are transformed back to the spatial domain. The reconstructed image only shows features of the image which phase information it contains.

Therefore, in image coding, linear phase filters are recommended. Linear phase cannot be combined with the property of orthogonality, except for the pair of filters with length 2 (the example in Section 2.2, the Haar wavelet). This filter is too short for practical purposes. Also, when looking at the filter characteristics of this filter, we conclude that it has no regularity (the basis function is not even continuous) and that the passband ripple, transition bandwidth, and stopband attenuation are very bad (Appendix E). Therefore, there is a need for longer linear phase filters, which can be designed by the extension of the wavelet theory to the biorthogonal case.

B.2 Biorthogonality

The idea of multiresolution can be extended to describe also the biorthogonal case. Biorthogonal filters themselves were known before the introduction of the wavelet theory ([Vett86]) although they did not have a special name. The biorthogonality replaces the stronger orthogonality requirement, which enables the construction of linear phase filters longer than 2.

The derivation of the formulas is analogous to the derivations described in Section 2.2. In the biorthogonal case, the space of square integrable functions $L^2(R)$ is divided two times into a series of embedded subspaces,

$$\dots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \subset \dots, \quad (\text{B.1})$$

$$\dots \subset \tilde{V}_2 \subset \tilde{V}_1 \subset \tilde{V}_0 \subset \tilde{V}_{-1} \subset \tilde{V}_{-2} \subset \dots. \quad (\text{B.2})$$

The hierarchies of subspaces V_m and $\tilde{V}_m$ have the properties

$$\overline{\bigcup_{m \in \mathbb{Z}} V_m} = L^2(R), \quad \bigcap_{m \in \mathbb{Z}} V_m = \{0\}, \quad (\text{B.3})$$

$$\overline{\bigcup_{m \in \mathbb{Z}} \tilde{V}_m} = L^2(R), \quad \bigcap_{m \in \mathbb{Z}} \tilde{V}_m = \{0\}, \quad (\text{B.4})$$

$$f(x) \in V_m \Leftrightarrow f(2x) \in V_{m-1}, \quad (\text{B.5})$$

$$f(x) \in \tilde{V}_m \Leftrightarrow f(2x) \in \tilde{V}_{m-1}. \quad (\text{B.6})$$

The sets of scaling functions $\phi_{m,n}(x)$ and $\tilde{\phi}_{m,n}(x)$ are basis functions of V_m and $\tilde{V}_m$, respectively. As in Section 2.2 we introduce the spaces W_m and $\tilde{W}_m$ as the complement of V_m and $\tilde{V}_m$ respectively, but now the orthogonality has been interchanged,

$$V_{m-1} = V_m \oplus W_m, \quad \tilde{V}_{m-1} = \tilde{V}_m \oplus \tilde{W}_m, \quad (\text{B.7})$$

$$V_m \perp \tilde{W}_m, \quad \tilde{V}_m \perp W_m. \quad (\text{B.8})$$

The sets of wavelets $\psi_{m,n}(x)$ and $\tilde{\psi}_{m,n}(x)$ are defined as linear combinations of the scaling functions,

$$\psi_{m,n}(x) = 2^{1/2} \sum_p g_p \phi_{m-1,p}(x), \quad (\text{B.9})$$

$$\tilde{\psi}_{m,n}(x) = 2^{1/2} \sum_p \tilde{g}_p \phi_{m-1,p}(x). \quad (\text{B.10})$$

These sets do not constitute an orthogonal basis of W_m and $\tilde{W}_m$ respectively, but they do possess the property

$$\langle \psi_{m,n}(x), \tilde{\psi}_{j,k}(x) \rangle = \delta_{m,j} \delta_{n,k}, \quad (\text{B.11})$$

and therefore the name biorthogonality is used.

Each function $f(x) \in L^2(R)$ can now be written as a weighted sum of basis functions $\tilde{\psi}_{m,n}(x)$,

$$f(x) = \sum_m \sum_n d_{m,n} \tilde{\psi}_{m,n}(x). \quad (\text{B.12})$$

Taking the inner product of both sides with $\psi_{j,k}(x)$, we get

$$\langle \psi_{j,k}(x), f(x) \rangle = \sum_m \sum_n d_{m,n} \langle \psi_{j,k}(x), \tilde{\psi}_{m,n}(x) \rangle = d_{j,k}, \quad (\text{B.13})$$

which defines the synthesis equation

$$f(x) = \sum_m \sum_n \langle \psi_{m,n}(x), f(x) \rangle \tilde{\psi}_{m,n}(x). \quad (\text{B.14})$$

Of course it is possible to interchange the role of $\psi(x)$ and $\tilde{\psi}(x)$.

If we stop at a certain scale m , we get

$$f(x) = f_m(x) + \sum_{j=-\infty}^m e_j(x), \quad (\text{B.15})$$

where $f_m(x)$ is the projection of $f(x)$ on $\tilde{V}_m$,

$$f_m(x) = \sum_n \langle \phi_{m,n}(x), f(x) \rangle \tilde{\phi}_{m,n}(x), \quad (\text{B.16})$$

and $e_j(x)$ is a projection on $\tilde{W}_m$,

$$e_j(x) = \sum_n \langle \psi_{j,n}(x), f(x) \rangle \tilde{\psi}_{j,n}(x). \quad (\text{B.17})$$

It is straightforward to derive the following relations analogous to the orthogonal case,

$$\phi_{0,0}(x) = 2 \sum_n h_n \phi_{0,0}(2x-n), \quad \Phi(\omega) = H\left(\frac{\omega}{2}\right) \Phi\left(\frac{\omega}{2}\right), \quad (\text{B.18})$$

$$\tilde{\phi}_{0,0}(x) = 2 \sum_n \tilde{h}_n \tilde{\phi}_{0,0}(2x - n), \quad \tilde{\Phi}(\omega) = \tilde{H}\left(\frac{\omega}{2}\right) \tilde{\Phi}\left(\frac{\omega}{2}\right), \quad (\text{B.19})$$

$$g_n = (-1)^{n+L_H-1} \tilde{h}_{L_H-1-n}, \quad G(\omega) = \tilde{H}(-\omega + \pi) e^{-i\omega(L_H-1)}, \quad (\text{B.20})$$

$$\tilde{g}_n = (-1)^{n+L_H} h_{L_H-1-n}, \quad \tilde{G}(\omega) = -H(-\omega + \pi) e^{-i\omega(L_H-1)}, \quad (\text{B.21})$$

$$H(\omega)\tilde{H}(\omega) - H(\omega + \pi)\tilde{H}(\omega + \pi) = 1. \quad (\text{B.22})$$

In Figure B.2 we show the filter bank description corresponding to the biorthogonal wavelet theory.

Figure B.2 Schematic filter bank scheme for the biorthogonal wavelet transform with the corresponding scaling functions and wavelets.

We wish to make two remarks on the derivations above. First, note that the synthesis high-pass filter $\tilde{g}_n$ is related to the *analysis* low-pass filter h_n (B.21), whereas the synthesis wavelet $\tilde{\psi}(x)$ is related to the *synthesis* scaling function $\tilde{\phi}(x)$ (B.10). This means that the regularity of $\tilde{\psi}(x)$, which is equal to the regularity of $\tilde{\phi}(x)$, is related to the number of zero moments of g_n (and not $\tilde{g}_n$) [Anto92].

Our second remark is that equation (B.22) appears most often in literature with a plus sign instead of a minus sign. For example, in [Cohe92], [Jawe93], and [Anto91] the plus sign appears, but only in [Cohe92] the authors mention that they did shift one of the filters over one sample to achieve this. We prefer the notation above, as the filters are always defined as causal filters starting with the first non-zero sample at $n = 0$. Note that the notation method of [Cohe92] is equivalent to defining all odd-

length filters as even-length filters that start or end with a 0. Maybe it would provide a more uniform description in wavelet theory if such a definition were introduced.

This concludes our listing of formulas that extend the multiresolution theory in the case of biorthogonal basis functions. The corresponding biorthogonal filters are frequently used in the experiments.

Appendix C

Derivations of Formulas

We show the derivations of several formulas and relations in the order in which they are encountered in the text. For every derivation we indicate the section in which is referred to this appendix.

C.1 Some wavelet theory equations (Section 2.2)

Equation (2.25) is the equivalent of the orthonormality of the scaling function $\phi(x)$, which is derived by using Parseval's relation:

$$\begin{aligned}
 \delta_{n,0} &= \int_{-\infty}^{\infty} \phi(x) \phi(x-n) dx \\
 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-in\omega} |\Phi(\omega)|^2 d\omega \\
 &= \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \int_{2\pi k}^{2\pi(k+1)} e^{-in\omega} |\Phi(\omega)|^2 d\omega \\
 &= \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \int_0^{2\pi} e^{-in\zeta} |\Phi(\zeta + 2\pi k)|^2 d\zeta \\
 &= \frac{1}{2\pi} \int_0^{2\pi} e^{-in\zeta} \sum_{k=-\infty}^{\infty} |\Phi(\zeta + 2\pi k)|^2 d\zeta.
 \end{aligned} \tag{C.1}$$

From this, it is clear that (2.25) holds. Then, equation (2.26) can be derived from (2.25), in the following way:

$$\begin{aligned}
 1 &= \sum_k |\Phi(2\omega + 2k\pi)|^2 \\
 &= \sum_k |H(\omega + k\pi)\Phi(\omega + k\pi)|^2 \\
 &= \sum_k (|H(\omega + 2k\pi)\Phi(\omega + 2k\pi)|^2 + |H(\omega + 2k\pi + \pi)\Phi(\omega + 2k\pi + \pi)|^2) \quad (\text{C.2}) \\
 &= |H(\omega)|^2 \sum_k |\Phi(\omega + 2k\pi)|^2 + |H(\omega + \pi)|^2 \sum_k |\Phi(\omega + 2k\pi + \pi)|^2 \\
 &= |H(\omega)|^2 + |H(\omega + \pi)|^2.
 \end{aligned}$$

In Table C.1 we show properties of filters, scaling functions, and wavelets in both their time-domain and their frequency-domain representation. The chosen formulas are the most important of those appearing in Section 2.2 and Section 2.3.

Table C.1 Some wavelet theory formula in their time-domain and their frequency-domain description

Time domain	Frequency domain
$\phi(x) = 2 \sum_n h_n \phi(2x - n)$	$\Phi(\omega) = H(\frac{\omega}{2})\Phi(\frac{\omega}{2})$
$\psi(x) = 2 \sum_n g_n \phi(2x - n)$	$\Psi(\omega) = G(\frac{\omega}{2})\Phi(\frac{\omega}{2})$
$\langle \phi(x), \phi(x - k) \rangle = \delta_{0,k}$	$\sum_k \Phi(\omega + 2k\pi) ^2 = 1$
$g_n = (-1)^n h_{-n+2l+1}$	$G(\omega) = -H(-\omega + \pi)e^{-i\omega(2l+1)}$
$2 \sum_n h_n h_{n+2k} = \delta_{0,k}$	$ H(\omega) ^2 + H(\omega + \pi) ^2 = 1$
$2 \sum_n g_n g_{n+2k} = \delta_{0,k}$	$ G(\omega) ^2 + G(\omega + \pi) ^2 = 1$
$2 \sum_n h_n g_{n+2k} = 0$	$H(-\omega)G(\omega) + H(-\omega + \pi)G(\omega + \pi) = 0$
$2 \sum_k (h_{n+2k} h_{p+2k} + g_{n+2k} g_{p+2k}) = \delta_{n,p}$	$ H(\omega) ^2 + G(\omega) ^2 = 1$

C.2 The input-output relation of a two-band splitting scheme (Section 3.2)

For the derivation of the input-output relation in (3.1) we repeat here the two-band splitting scheme shown in Figure C.1 together with a notation for the signals at different points in the scheme. Again we assume that there is neither quantization distortion nor any channel errors. The index position in the notation is reserved for the branch, where 0 indicates the low-pass and 1 the high-pass branch. Therefore, the discrete time (spatial) domain variable n is written between parentheses.

Figure C.1 Two-channel analysis/synthesis scheme. The filters H and F are low-pass filters and the filters G and E are high-pass filters.

We have the following relations

$$U_0(\omega) = H(\omega)S(\omega)e^{i\omega p}, \quad (\text{C.3})$$

$$U_1(\omega) = G(\omega)S(\omega)e^{i\omega q}. \quad (\text{C.4})$$

The downsampling is defined as retaining all even-numbered samples and throwing away the odd-numbered samples,

$$v_k(n) = u_k(2n), \quad k = 0, 1, \quad (\text{C.5})$$

which is in the frequency domain

$$V_k(\omega) = \frac{1}{2} \left[U_k\left(\frac{\omega}{2}\right) + U_k\left(\frac{\omega}{2} + \pi\right) \right]. \quad (\text{C.6})$$

The upsampling consists of stretching the signal by factor 2 and putting a zero at the open, odd-numbered positions,

$$\begin{aligned} w_k(2n) &= v_k(n) \\ w_k(2n+1) &= 0 \end{aligned} \quad k = 0, 1, \quad (\text{C.7})$$

which is in the frequency domain

$$W_k(\omega) = V_k(2\omega), \quad k = 0, 1. \quad (\text{C.8})$$

Finally, the synthesis filterings are described by

$$X_0(\omega) = F(\omega)W_0(\omega)e^{i\omega r}, \quad (\text{C.9})$$

$$X_1(\omega) = E(\omega)W_1(\omega)e^{i\omega s}. \quad (\text{C.10})$$

With these formulas it is easy to verify that

$$\begin{aligned} O(\omega) &= X_0(\omega) + X_1(\omega) \\ &= [F(\omega)H(\omega)e^{i\omega(p+r)} + E(\omega)G(\omega)e^{i\omega(q+s)}] S(\omega) + \\ &\quad [F(\omega)H(\omega+\pi)e^{i\omega(p+r)}e^{ip\pi} + E(\omega)G(\omega+\pi)e^{i\omega(q+s)}e^{iq\pi}] S(\omega+\pi), \end{aligned} \quad (\text{C.11})$$

and we have derived (3.1).

In the equations above, we defined the downsampling as keeping the even-numbered samples and throwing away the odd-numbered samples, which is usual. This is also reflected in our diagrams of Figure 3.3, Figure 3.4, and Figure 3.5 by the open squares in the subbands. However, the odd-length symmetric filter $G(\omega)$ of Figure 3.4b is centred on the odd positions of the signal. This suggests that it is theoretically also correct to define the high-pass filtered signal in the analysis filter bank to be downsampled on the alternative grid. Consequently, the filter offsets become different. Below, we describe the derivation of the input-output relation with this alternative viewpoint.

After high-pass filtering in the analysis bank, i.e.,

$$\tilde{U}_1(\omega) = G(\omega)S(\omega)e^{i\omega\tilde{q}}, \quad (\text{C.12})$$

we would have

$$\tilde{v}_1(n) = \tilde{u}_1(2n+1), \quad (\text{C.13})$$

which is in the frequency domain

$$\tilde{V}_1(\omega) = \frac{1}{2}e^{i\frac{\omega}{2}} \left[\tilde{U}_1\left(\frac{\omega}{2}\right) - \tilde{U}_1\left(\frac{\omega}{2} + \pi\right) \right]. \quad (\text{C.14})$$

The upsampling is then defined as

$$\begin{aligned} w_1(2n+1) &= v_1(n) \\ w_1(2n) &= 0, \end{aligned} \tag{C.15}$$

which is in the frequency domain

$$\tilde{W}_1(\omega) = \tilde{V}_1(2\omega)e^{-i\omega}. \tag{C.16}$$

With the synthesis filtering

$$\tilde{X}_1(\omega) = E(\omega)\tilde{W}_1(\omega)e^{i\omega\tilde{s}} \tag{C.17}$$

the input-output relation becomes

$$\begin{aligned} O(\omega) &= X_0(\omega) + \tilde{X}_1(\omega) \\ &= [F(\omega)H(\omega)e^{i\omega(p+r)} + E(\omega)G(\omega)e^{i\omega(\tilde{q}+\tilde{s})}]S(\omega) + \\ &\quad [F(\omega)H(\omega+\pi)e^{i\omega(p+r)}e^{ip\pi} - E(\omega)G(\omega+\pi)e^{i\omega(\tilde{q}+\tilde{s})}e^{i\tilde{q}\pi}]S(\omega+\pi). \end{aligned} \tag{C.18}$$

This is equivalent to (C.11) when the filter offsets $\tilde{q}$ and $\tilde{s}$ are chosen as

$$\tilde{q} = q - 1, \quad \tilde{s} = s + 1, \tag{C.19}$$

thus both descriptions yield the same implementation.

C.3 The optimal bit rate (Section 5.5.1)

We have to minimize

$$\sum_{k=0}^{M-1} \gamma_k w_k \sigma_{xk}^2 2^{-2R_k} \quad (\text{C.20})$$

under the constraint

$$R = \sum_{k=0}^{M-1} \left(\frac{N_k}{N} \right) R_k. \quad (\text{C.21})$$

Using the method of Lagrange multipliers this means we must solve

$$\frac{\partial}{\partial R_k} \left\{ \sum_{k=0}^{M-1} \gamma_k w_k \sigma_{xk}^2 2^{-2R_k} - \lambda \left(R - \sum_{k=0}^{M-1} \left(\frac{N_k}{N} \right) R_k \right) \right\} = 0, \quad (\text{C.22})$$

which yields

$$-2({}^e\log 2) \gamma_k w_k \sigma_{xk}^2 2^{-2R_k} + \left(\frac{N_k}{N} \right) \lambda = 0, \quad (\text{C.23})$$

so,

$$R_k = \frac{1}{2} {}^2\log \left(\gamma_k \left(\frac{N}{N_k} \right) w_k \sigma_{xk}^2 \right) - \frac{1}{2} {}^2\log \left(\frac{\lambda}{2({}^e\log 2)} \right). \quad (\text{C.24})$$

Substitution of (C.24) into (C.21) gives

$$\frac{1}{2} {}^2\log \left(\frac{\lambda}{2({}^e\log 2)} \right) = \frac{1}{2} \sum_{k=0}^{M-1} \left(\frac{N_k}{N} \right) {}^2\log \left(\gamma_k \left(\frac{N}{N_k} \right) w_k \sigma_{xk}^2 \right) - R, \quad (\text{C.25})$$

and after using this in (C.24) for the right-hand term, we obtain

$$R_k = R + \frac{1}{2} {}^2\log \left(\frac{\gamma_k \left(\frac{N}{N_k} \right) w_k \sigma_{xk}^2}{\prod_{l=0}^{M-1} \left(\gamma_l \left(\frac{N}{N_l} \right) w_l \sigma_{xl}^2 \right)^{N_l/N}} \right). \quad (\text{C.26})$$

C.4 The input-output relation of a quincunx two-band splitting scheme (Section 6.3)

For the derivation of the input-output relation for two-dimensional filtering we show in Figure C.2 the two-band splitting scheme, together with the notation of the signals at different points in the scheme. Again we assume that there is neither quantization distortion nor any channel errors. The discrete time (spatial) domain variable $\mathbf{n}$ is written between parentheses, as the index is used to indicate the branch.

Figure C.2 Two-channel analysis/synthesis scheme for two-dimensional filtering. The filters H and F are low-pass filters and the filters G and E are high-pass filters.

We have the following relations

$$U_0(\omega) = H(\omega)S(\omega)e^{i\omega^T p}, \quad (\text{C.27})$$

$$U_1(\omega) = G(\omega)S(\omega)e^{i\omega^T q}, \quad (\text{C.28})$$

where the symbol ω now represents a vector, and T means transposition. The down-sampling is defined as keeping all samples on position $\mathbf{m}$ for which $\mathbf{1}^T \mathbf{m}$ is even. Those points can be written as

$$\mathbf{m} = \mathbf{R}^T \mathbf{n}, \quad \mathbf{n} \in \mathbb{Z}^2, \quad (\text{C.29})$$

where we choose the matrix $\mathbf{R}$ as

$$\mathbf{R} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}. \quad (\text{C.30})$$

Note that this matrix is equivalent to the scaled rotation matrix of Section A.2,

$$a_0 \mathbf{R}_{\pi/4} = \sqrt{2} \begin{bmatrix} \cos(\pi/4) & -\sin(\pi/4) \\ \sin(\pi/4) & \cos(\pi/4) \end{bmatrix}. \quad (\text{C.31})$$

Thus, the downsampling is defined as

$$v_k(n) = u_k(\mathbf{R}^T \mathbf{n}), \quad k = 0, 1, \quad (\text{C.32})$$

which is in the frequency domain

$$V_k(\omega) = \frac{1}{2} [U_k(\mathbf{R}^{-1} \omega) + U_k(\mathbf{R}^{-1} \omega + \pi)], \quad k = 0, 1, \quad (\text{C.33})$$

where $\pi = [\pi, \pi]^T$. The upsampling consists of putting a zero at the open, odd numbered positions,

$$\begin{aligned} w_k(\mathbf{R}^T \mathbf{n}) &= v_k(n) \\ w_k(\mathbf{R}^T \mathbf{n} + \begin{bmatrix} 1 \\ 0 \end{bmatrix}) &= 0 \end{aligned} \quad k = 0, 1, \quad (\text{C.34})$$

which is in the frequency domain

$$W_k(\omega) = V_k(\mathbf{R} \omega), \quad k = 0, 1. \quad (\text{C.35})$$

Finally, the synthesis filterings are described by

$$X_0(\omega) = F(\omega)W_0(\omega)e^{i\omega^T r}, \quad (\text{C.36})$$

$$X_1(\omega) = E(\omega)W_1(\omega)e^{i\omega^T s}. \quad (\text{C.37})$$

With these formulas it follows that the input-output relation becomes (with the Greek symbols denoting vectors)

$$O(\omega) = X_0(\omega) + X_1(\omega)$$

$$= [F(\omega)H(\omega)e^{i\omega^T(p+r)} + E(\omega)G(\omega)e^{i\omega^T(q+s)}] S(\omega) + \quad (\text{C.38})$$

$$[F(\omega)H(\omega + \pi)e^{i\omega^T(p+r)} e^{i\pi^T p} + E(\omega)G(\omega + \pi)e^{i\omega^T(q+s)} e^{i\pi^T q}] S(\omega + \pi).$$

Appendix D

On the Entropy of a 4-Connected Chain Code

In this appendix we describe a method that reaches a bit rate of 1.5 bit per link for coding a 4-connected chain by using a fixed length code. The chain represents the segmentation of an image divided into regions. Every *link* in the chain is the border between two pixels, and every link connects two *nodes*, which are positioned between four pixels. The method does not include the coding of the starting points, but this can be neglected for a relatively small number of regions.

A 4-connected chain code starts with the coding of the address of a certain node in the chain, usually a node that is on the border of the image. Then each next node can be coded relative to the previous node, and there are, in general, three choices possible: turn left, turn right, and going in the same direction. When this information is coded straightforwardly, we have an entropy of $^2\log 3 = 1.58$, but a fixed length coding requires 2 bits per link.

We now consider groups of k consecutive links and code them with a fixed length code. At first sight, the number of possible groups seems to be 3^k , but close examination reveals that not all are possible: the chain of k links may not cross itself (except for very small regions, which will be handled separately). We denote the number of possible chain codes by $c(k)$. In Figure D.1 we show an example for $k = 3$. Here, only 25 of the $3^3 = 27$ chain codes are possible.

Figure D.1 The 25 possible groups of chains of length 3, $c(3)$. The dotted line represents the last link of the previously coded group of links.

Figure D.2 The coding of the residual after dividing into groups of k links (here we use $k = 8$). The dot represents the starting point. (a) A new starting point has to be coded; (b) The chain continues.

If we aim at a bit rate of 1.5 bits per link we should find a k for which $c(k) \leq 2^{1.5k}$. In Table D.1 we present the values of $c(k)$ for k up to 16. We can conclude that a length of 8 gives a bit rate of 1.5 bit per link, where 29 code words of length 12 are not used.

Table D.1 Entropy and bit rate of a fixed length coding for 4-connected chains

k	Nr. of theoretical chains, 3^k	Nr. of possible chains, $c(k)$	Entropy, $(^2\log(c(k)))/k$	Bit rate with fixed length coding	Nr. of unused code words with fixed length coding
1	3	3	1.585	2.000	1
2	9	9	1.585	2.000	7
3	27	25	1.548	1.667	7
4	81	71	1.537	1.750	57
5	243	195	1.521	1.600	61
6	729	543	1.514	1.667	481
7	2187	1479	1.504	1.571	569
8	6561	4067	1.499	1.500	29
9	19683	11025	1.492	1.556	5359
10	59049	30073	1.488	1.500	2695
11	177147	81233	1.483	1.545	49839
12	531441	220375	1.479	1.500	41769
13	1594323	593611	1.475	1.538	454965
14	4782969	1604149	1.472	1.500	493003
15	14348907	4311333	1.469	1.533	4077275
16	43046721	11616669	1.466	1.500	5160547

Theoretically, we can reach a higher bit rate by taking large values of k . For instance, if we take $k = 24$ we can use code words of length 30 (of which more than two hundred million are not used) to get a bit rate of 1.25 bpp. Of course, the number of code words requires too much memory space to be applicable.

Now we have achieved a chain coding with 1.5 bits per link for $k = 8$. However, we made three assumptions. Firstly, after splitting the contours into parts of k links, there is in general a residual r links. This residual is coded with the same code words, where we have a choice for the remaining $k - r$ links in the chain. Either we can

choose the last $k - r$ links in such a way that they match completely links that are already coded. At the decoder, they are neglected and the next code words are considered to be a new starting point. Alternatively, we can attempt to match the last $k - r$ links with an uncoded part of the contour. Then, the coding can be continued without the coding a starting point (Figure D.2).

The second assumption is that there are more than k links for every chain. If this is not the case, the remaining $2^{1.5k} - c(k)$ code words can be used to code these small regions. In Figure D.3 we show for $k = 8$ the 10 small regions that have at most 8 links. These 10 can easily be coded with the remaining 29 code words.

Thirdly, in the calculations we did not take into account the coding of the starting points. For our segmentation of the Camera-man and Vectra images, the number of starting points are 10 and 51, respectively, in our implementation.

Of course, the bit rate with a fixed code word length is suboptimal. A variable length coding can be used to exploit the statistical dependency between two successive links.

Figure D.3 The 10 possible regions that have a perimeter not exceeding 8.

Appendix E

Filters

In literature, many filters appear that are designed to perform a subband/wavelet decomposition. This appendix contains a schematic overview of the filters that are used in this thesis. For each filter we mention its origin, the design criteria, and whether the filter enables perfect reconstruction or not. Where possible, we give the filter characteristics, such as peak passband ripple, stopband attenuation, transition bandwidth, and regularity. Graphs with the magnitude and phase response, and the limit functions of the iterated filters (scaling functions and wavelets) are added. For some of the filters, the coefficients are tabulated. However, the most well-known filter coefficients are left out, since they have been published in many places.

E.1 The Johnston filters

These filters were published by Johnston in [John80]. The design technique comes from the subband coding theory and the filters are found by minimizing an error criterion while the transition bandwidth is fixed. The error criterion weights the peak passband ripple and the peak stopband ripple. All filters have linear phase and do not enable perfect reconstruction. More information about these filters can be found in [John80] and Section 5.2.

The filter coefficients of the modified Johnston filters are presented in Table E.1. The filter characteristics of the most important Johnston filters and the modified versions are shown in Table E.2 and Table E.3. Figure E.1, Figure E.2, and Figure E.3 show the magnitude and phase response, and the limit functions of the iterated filters of joh16C, joh16CC, and joh32D, respectively.

Table E.1 Filter coefficients of the modified Johnston filters

joh8AA	joh12AA	joh12BB	joh16CC	joh32EE
0.01124276	-0.0043096449	-0.0073726620	0.0076118200	0.00590535010
-0.07065183	0.0188565910	0.0274553890	-0.0204875100	-0.01127565000
0.06942827	-0.0027103261	-0.0075816400	0.0019911500	-0.00096245919
0.48998080	-0.0846959430	-0.0913825040	0.0464768410	0.01568082000
	0.0884699230	0.0980852170	-0.0262755990	-0.00261204110
	0.4843894000	0.4807962000	-0.0992954970	-0.02103823100
			0.1178666000	0.00737988620
			0.4721122000	0.02812259100
				-0.01456900000
				-0.03830612800
				0.02662431100
				0.05570720900
				-0.05138257100
				-0.09768378700
				0.13876420000
				0.45964550000

Figure E.1 Frequency responses of *joh16C*, and the limit functions of the iterated low-pass and high-pass filter. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) ψ .

Figure E.2 Frequency responses of *joh16CC*, and the limit functions of the iterated low-pass and high-pass filter. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) ψ .

Figure E.3 Frequency responses of *joh32D*, and the limit functions of the iterated low-pass and high-pass filter. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) ψ .

Table E.2 Filter characteristics of the Johnston filters

Filter	δ_p	δ_s	A_p (dB)	A_s (dB)	ω_p	ω_s	$\Delta \omega$	Regularity
joh8A	0.0034	0.0266	0.029	31.51	0.766	2.534	1.770	0.82
joh12A	0.0010	0.0041	0.009	47.65	0.873	2.481	1.608	0.87
joh12B	0.0023	0.0201	0.020	33.93	0.999	2.261	1.261	0.70
joh16B	0.0008	0.0062	0.007	44.17	1.045	2.230	1.185	1.21
joh16C	0.0040	0.0349	0.035	29.14	1.204	2.017	0.813	0.86
joh32E	0.0118	0.0577	0.103	24.78	1.435	1.752	0.317	0.73

Table E.3 Filter characteristics of the modified Johnston filters

Filter	δ_p	δ_s	A_p (dB)	A_s (dB)	ω_p	ω_s	$\Delta \omega$	Regularity
joh8AA	0.0026	0.0298	0.023	30.52	0.702	2.517	1.815	0.80
joh12AA	0.0013	0.0040	0.012	48.10	0.879	2.479	1.600	0.85
joh12BB	0.0024	0.0211	0.021	33.51	0.984	2.255	1.271	0.69
joh16CC	0.0034	0.0364	0.030	28.77	1.188	2.013	0.825	0.84
joh32EE	0.0107	0.0591	0.093	24.57	1.432	1.751	0.319	0.71

E.2 The Daubechies filters

These orthonormal wavelet filters were published by Daubechies in [Daub88]. Their magnitude responses have maximal flatness at $\omega = \pi$ with respect to their length. This family of filters has infinitely many members. All filters have even length and non-linear phase and they enable perfect reconstruction. The Daubechies filter of length 2 is known as the Haar filter, which appeared for the first time in [Haar10].

In Table E.4 the regularity of the two filters used in the experiments are given. Figure E.4 and Figure E.5 show the magnitude and phase response, and the limit functions of the two filters. For the filter coefficients of the Daubechies filters refer to [Daub88].

Table E.4 Regularity of two Daubechies filters

Filter	Regularity
dau4	0.55
dau16	2.76

Figure E.4 Frequency responses, scaling function, and wavelet of the filter dau4. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) ψ .

Figure E.5 Frequency responses, scaling function, and wavelet of the filter *dau16*. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) ψ .

E.3 The biorthogonal wavelet filters of Cohen, Daubechies and Feauveau

The biorthogonal filter pairs cdf5-7, cdf9-7 were designed in [Cohe92]. The filters of the cdf17-15 pair are designed with the same criteria as cdf9-7 and are published here for the first time. For details, about the design process refer to Section 5.3.2 or [Cohe92]. All these filters enable perfect reconstruction and have linear phase and odd length. The design criterion of the cdf5-7 filter is that the filter coefficients are integer and that the analysis low-pass filter is of the class of filters used in [Burt83]. The design process of the other two filters concentrated on the maximal flatness of both filters in the bank, with the restriction that the filter lengths are as close as possible.

In Table E.5 we list the regularity of the filters. The magnitude and phase responses as well as the scaling functions and wavelets are shown in Figure E.6, Figure E.7, and Figure E.8. For the filter coefficients of the cdf5-7 and cdf9-7 filter refer to [Cohe92]. The coefficients of the cdf17-15 filter are presented in Table E.6.

Table E.5 Regularity of the biorthogonal filter pairs

Filter	Regularity of analysis low-pass filters	Regularity of synthesis low-pass filters
cdf5-7	0.51	0.57
cdf9-7	1.07	1.70
cdf17-15	2.60	2.89

Table E.6 Filter coefficients of the low-pass filters of cdf17-15

0	0.50899981607	0.60980581216
1	0.27261042715	0.32946843957
2	-0.02905910110	-0.06383907747
3	-0.02710419682	-0.09898186984
4	0.03387560846	0.00977042376
5	0.00499186475	0.02182717956
6	-0.01069785249	-0.00083425237
7	-0.00049809509	-0.00231374929
8	0.00138143709

Figure E.6 Frequency responses (analysis filters), scaling functions, and wavelets of the filter pair cdf5-7. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) $\tilde{\phi}$; (f) ψ ; (g) $\tilde{\psi}$.

Figure E.7 Frequency responses (analysis filters), scaling functions, and wavelets of the filter pair cdf9-7. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) $\tilde{\phi}$; (f) ψ ; (g) $\tilde{\psi}$.

Figure E.8 Frequency responses (analysis filters), scaling functions, and wavelets of the filter pair cdf17-15. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) $\tilde{\phi}$; (f) ψ ; (g) $\tilde{\psi}$.

E.4 The filter pair with lengths 9 and 7 of Kronander

The biorthogonal filter pair kro9-7 was published first in [Kron89] by Kronander. The filter pair enables perfect reconstruction and the filters have linear phase and odd length. The design criterion is based on the combined optimization of step-response and frequency error.

In Table E.7 we show the filter characteristics of the filter pair. Figure E.9 shows the frequency responses, the scaling functions, and the wavelets. Note the extreme irregularity of the analysis filter pair, which is not clearly influencing the coding performance (Section 5.5).

Table E.7 Filter characteristics of the Kronander filter pair

kro9-7	δ_p	δ_s	A_p (dB)	A_s (dB)	ω_p	ω_s	$\Delta \omega$	Regularity
length 9	0.1127	0.2403	1.039	12.38	1.755	2.264	0.509	-.02
length 7	0.0318	0.0436	0.281	27.21	0.932	2.124	1.192	1.54

Figure E.9 Frequency responses (analysis filters), scaling functions, and wavelets of the filter pair kro9-7. (a) Magnitude response; (b) Magnitude response in dB; (c) Phase response; (d) ϕ ; (e) $\tilde{\phi}$; (f) ψ ; (g) $\tilde{\psi}$.

E.5 The Smith and Barnwell filter of length 16

This filter is one of the first published quadrature mirror filters that enabled perfect reconstruction. In literature these filters are known as conjugate quadrature filters (CQF), and were introduced in [Smit86]. These filters have non-linear phase and even length. The design method concentrates on optimizing the stopband attenuation, and other criteria that are typical for the subband coding theory.

In Table E.8 we show the filter characteristics of the filter pair. Figure E.10 shows the frequency responses, the scaling function, and the wavelet.

Table E.8 Filter characteristics of the Smith and Barnwell filter

Filter	δ_p	δ_s	A_p (dB)	A_s (dB)	ω_p	ω_s	$\Delta \omega$	Regularity
sb16	0.000	0.0101	0.000	39.95	1.052	2.070	1.018	-0.02

Figure E.10 Frequency responses of the filter sb16, and the limit functions of the iterated low-pass and high-pass filter.

(a) Magnitude response; (b) Magnitude response in dB;

(c) Phase response; (d) ϕ ; (e) ψ .

Appendix F

Test Images and Video Sequences

In this appendix we show the test images and the first frames of the video sequences used in the experiments. The relative sizes of the images is preserved, except for the Mobile image, which is shown at half resolution.

Aerial (256x256)

Baboon (512x512)

Camera-man (256x256)

Clown (512x512)

Lena (512x512)

Mobile (720x576)

Claire (352x288)

Vectra (352x288)

Samenvatting

Gedurende de laatste jaren is er een enorme toename te zien geweest in de uitwisseling en opslag van informatie. Tegelijkertijd is visuele informatie in de vorm van digitale beelden en video een steeds belangrijkere plaats gaan innemen. Juist deze vorm van informatie brengt grote opslag- en transmissiekosten met zich mee. Daarom is het onderzoek naar snelle en betrouwbare beeldcompressietechnieken voor verschillende beeldkwaliteiten van groot belang. In de literatuur zijn meerdere methoden voorgesteld en onderzocht, waarvan er al enkele internationaal gestandaardiseerd zijn. Toch blijft er vraag naar nieuwe compressietechnieken, voor verbeterde standaarden en specifieke toepassingsgebieden. Dit proefschrift behandelt een van de meest recent ontwikkelde datacompressietechnieken met betrekking tot de toepassing voor digitale beeld- en videocodering. Deze techniek beschrijft de signalen als een gewogen som van speciale basisfuncties, de zogenaamde wavelets.

De ontwikkeling van de wavelet-theorie is mede voortgekomen uit het onvermogen van de Fourier-theorie om niet-stationaire signalen te kunnen representeren. Ondanks dat de wavelet-theorie in bepaalde opzichten een vervolg is op de Fourier-theorie wordt dit niet weerspiegeld in de notatie en de naamgeving in de meeste artikelen en boeken over wavelet-theorie. Een van de bijdragen van dit proefschrift is het gebruik van een consistente notatie en naamgeving in volledige analogie met de Fourier-theorie. Daarbij is de keus gemaakt voor de notatie zoals die gebruikt wordt in de signaalverwerkingsliteratuur [Oppe83] en niet voor de notatie uit de wiskundige literatuur.

De meeste boeken beschrijven de wavelet-theorie vanuit een wiskundig standpunt. Voor een goed begrip en een juiste implementatie voor beeld- en videocodering is daarentegen een grondige kennis van de wiskundige achtergronden niet noodzakelijk. In dit proefschrift worden die achtergronden dan ook zoveel mogelijk weggelaten.

De nadruk in dit proefschrift ligt op het onderdeel van de wavelet-theorie dat toegepast wordt in de beeldcodering: de discrete wavelet transformatie (DWT). Het codeerschema van de DWT blijkt sterk overeen te komen met het subband-codeerschema met twee kanalen dat sinds de jaren zeventig gebruikt wordt in de signaalverwerking. Het voornaamste verschil ligt in de eigenschappen van de gebruikte filters. Volgens de theorie van het subbandcoderen (SBC) moet een filter goede frequentieëigenschappen bezitten, zoals voldoende demping in de stopband en een korte overgang tussen doorlaatband en stopband. Een bekend voorbeeld van zulke filters wordt gevormd door de klasse van Johnston filters, waarvan we in sommige een verbetering aangebracht hebben. Anders gezegd, filters die ontworpen zijn volgens de SBC theorie hebben als doel zo goed mogelijk de ideale blokvorm in het frequentiedomein te benaderen. Een heel ander uitgangspunt bij het ontwerpen van filters wordt gebruikt bij de wavelet-theorie. Hierbij is het doel een filter een zekere mate van regulariteit te geven, wat een recent gedefinieerde filtereigenschap is. De bekendste wavelet filters zijn de Daubechies filters.

Regulariteit kan geïnterpreteerd worden als de mate van gladheid van de limietfunctie behorende bij het filter. De limietfunctie is de limiet van herhaalde filters in een boomstructuur van oneindige diepte. De regulariteit is sterk gecorreleerd aan het aantal afgeleiden van de frequentiekaracteristiek dat gelijk is aan nul op de helft van de bemonsterfrequentie. Een andere bijdrage van de wavelet-theorie aan de reeds bestaande SBC theorie is de connectie tussen digitale filters en de theorie van continue functieruimten en basisfuncties.

Een van de onderzoeksgroepen die de DWT toepasten in de beeldcodering stelde dat de regulariteitseigenschap belangrijk is, omdat die leidt tot minder storende kwantisatiefouten in de gereconstrueerde beelden [Anto92]. Dit is niet aangetoond in onze experimenten, waarin geen relatie naar voren kwam tussen regulariteit en codeerprestatie. Een andere veronderstelling werd wel bevestigd door onze experimenten. Het blijkt dat voor filterparen waar een van de filters een lage regulariteit heeft, dit filter het best in de analyse gebruikt kan worden. Alleen als beide filters een regulariteit hebben die boven 2 uitkomt, wordt de plaatsing van de filters in de filter-

banken bepaald door andere filtereigenschappen. Voor beeldcoderingstoepassingen lijkt een regulariteit van 2 voldoende.

Als we de codeerprestatie vergelijken van filters, zowel ontworpen met de SBC theorie als met de wavelet-theorie, dan vertonen de resultaten in het algemeen niet al te grote verschillen. Voor het vergelijken van de prestaties gebruiken we de zogenaamde subband coding gain, de signaal-ruis-verhouding (een objectieve maat) en de visuele beeldkwaliteit (een subjectieve maat). De experimenten zijn uitgevoerd met zes beelden en twee beeldsequenties (video's). De biorthogonale wavelet filters blijken dan het best te presteren. Deze filters hebben een lineaire fase. Vervolgens komen de lineaire fase SBC filters, en de niet-lineaire fase wavelet filters.

Het is mogelijk om de eendimensionale wavelet-theorie uit te breiden naar meerdere dimensies, waarvan we hier alleen het tweedimensionale (2-D) geval beschouwen. De algemene 2-D DWT heeft een verscheidenheid aan varianten, waarvan we de twee eenvoudigste nader onderzoeken. De een is de gebruikelijke gescheiden implementatie van de DWT (de dyadische 2-D DWT) die ook in de bovenstaande experimenten gebruikt is. De andere is de zogenaamde quincunx 2-D DWT die, net als de eendimensionale DWT, het signaal opdeelt in twee frequentiebanden, een met lage en een met hoge frequenties. De verwachting is dat deze quincunx 2-D DWT beter in staat is om sommige soorten beelden als luchtfoto's te comprimeren, omdat geen enkele richting bevoordeeld wordt boven andere richtingen. Het nadeel van de quincunx 2-DWT is de grotere complexiteit vergeleken met de dyadische 2-D DWT.

Bij de experimenten vergelijken we de codeerprestaties van de beide varianten van de 2-D DWT. De filters van de quincunx 2-D DWT zijn gerelateerd aan die van de dyadische 2-D DWT door middel van de McClellan transformatie. Het blijkt dat de quincunx 2-D DWT een lagere beeldkwaliteit oplevert, ook voor die soorten beelden waarvoor deze geschikt zou moeten zijn. Verder onderzoek in de richting van quincunx 2-D DWT zou zich voornamelijk moeten richten op het ontwerp van betere 2-D filters.

Naast de beeldcompressie methoden die gebaseerd zijn op transformatie, zoals de DWT, zijn in de literatuur ook verscheidene segmentatie-gebaseerde (region-based) methoden gepubliceerd (bijvoorbeeld in [Kunt85]). Het basisprincipe achter segmentatie-gebaseerd coderen is het opdelen van het beeld in verschillende, niet-overlappende segmenten. De randen van deze gebieden en de inhoud ervan (aangeduid met de term textuur) worden apart gecodeerd. In het ideale geval zullen de gebieden corresponderen met objecten in het beeld.

De meest gebruikelijke methode om textuur te coderen is de representatie van de grijswaarden in het beeld met behulp van polynomen. De DWT was tot nu toe niet geschikt voor textuurcodering, aangezien een signaaldecompositie in d niveaus een segment veronderstelt dat is opgebouwd uit blokken van $2^d \times 2^d$ beeldpunten. In het algemeen zullen zulke segmenten niet passen bij de objecten in het te coderen beeld. In dit proefschrift wordt een nieuwe methode voor signaalextensie geïntroduceerd die het mogelijk maakt om een wavelet-decompositie uit te voeren zonder extra transformatiecoëfficiënten te introduceren. Wanneer deze methode toegepast wordt op tweedimensionale signalen dan is het mogelijk om een DWT uit te voeren op segmenten van willekeurige vorm. Deze variant van de DWT wordt de segment-gebaseerde DWT genoemd (region-based discrete wavelet transform, RBDWT). De RBDWT kan eveneens geïnterpreteerd worden als een DWT waarbij de subbanden opgedeeld zijn in segmenten.

Hiermee is het mogelijk om een nieuw segmentatie-gebaseerd codeerschema te maken, waarin de textuur wordt gecodeerd met de RBDWT. Een voordeel van dit schema is dat het in staat is de beschikbare hoeveelheid bits te verdelen over zowel subbanden als segmenten. Een goede segmentatie bestaat over het algemeen uit relatief grote segmenten om de splitsing in subbanden optimaal te kunnen benutten.

Omdat de aandacht vooral uitgaat naar de implementatieaspecten van de RBDWT, gaan we niet in op de optimalisatie van het segmentatieproces. Aangezien het een belangrijk onderdeel is in het codeerschema bevelen we verder onderzoek in deze richting sterk aan. Met een geoptimaliseerde segmentatie kan de codeerprestatie van de RBDWT vergeleken worden met andere methoden voor textuurcodering.

Zoals veel segmentatie-gebaseerde codeermethoden is de RBDWT niet erg geschikt voor hiërarchisch coderen. De oorzaak hiervan ligt in het feit dat de bemonstering van de segmentatieinformatie een blokkerige fout in het gereconstrueerde beeld tot gevolg heeft bij lage resoluties. Bij het toepassen van de RBDWT voor de codering van video moet voornamelijk gedacht worden aan het coderen van die gebieden waarbij bewegingscompensatie geen goed resultaat geeft.

De codeerprestatie van het nieuwe codeerschema met de RBDWT is experimenteel vergeleken met de codeerprestatie van het standaard DWT codeerschema. De standaard DWT geeft weliswaar betere signaal-ruis-verhoudingen, maar de subjectieve beeldkwaliteit voor het RBDWT schema is zeker niet minder dan die van de standaard DWT. Dit komt voornamelijk doordat de randen beter worden gerepresenteerd en de zogenaamde ringing artefacten kleiner zijn. Daarom kunnen we concluderen dat de RBDWT een nieuwe codeermethode is die verder onderzoek verdient.

Acknowledgements

Writing a thesis is never the work of one person. Therefore, I would like to thank here those who have helped in one way or another to complete this work.

Firstly, I would like to thank Dick Boekee for introducing me to the field of image and video coding, and Jan van der Lubbe for his support in the first two years. Then, I would like to thank Bülent Sankur from Bogaziçi University, Istanbul, for his supervision during his two-year stay at the Information Theory Group and his continuing support and interest afterwards, and for being a member of the committee. I continued my work under the supervision of Jan Biemond and Jos Weber. I would like to thank both for their overall support during the last two years. As my mentor, Jos was always making time for me, and helped me a lot, especially in organizing the materials and keeping the overall aim and the time schedule in mind.

Further, I would like to thank all other members of the Information Theory Group. They were always helpful in solving various problems, and they created a friendly and pleasant environment in which to work. Some of them I want to mention separately. Firstly, Inald Lagendijk has read and given comments on some parts of my work. The software environment created and maintained by Ricardo Belfor and Frank Bosveld was very helpful. I thank Frank also for the discussions on several coding topics and for enabling me to use his quantization software. Further, I had several discussions with René van der Vleuten and he drew my attention to an important reference. Last but not least, I would like to thank my two roommates Marcel Reinders and Peter van Beek, not only for their help and numerous discussions about

my work, but also for the many, many discussions during lunch time, on scientific, political, religious, and social subjects. I enjoyed it very much.

Mrs. J. B. Zaat-Jones corrected the draft versions of the English text. Further, I would like to thank the members of Professor M. Barlaud's group for their hospitality during my one-week visit to the University of Nice-Sophia Antipolis.

Also outside the university I needed and received the support of many people. I thank my friends, for whom I could not make as much time as I would have liked to. Finally, I thank my family, for too many things to write down.

Curriculum Vitae

Herman Johannes (Herjan) Barnard was born in Almkerk, the Netherlands, on 1st January 1967. In 1985 he obtained his VWO diploma from the Christelijk Lyceum Dr. W. A. Visser 't Hooft, in Leiden. In September of the same year he started his course at the Department of Technical Mathematics of the Delft University of Technology, the Netherlands. In January 1990 he received his M. Sc. (cum laude) after finishing a Master's thesis project in the Applied Mathematics Group under the supervision of prof. dr. G. J. Olsder.

From January 1990 to May 1994 he worked as a Ph.D. student (A.I.O.) at the Information Group of the Department of Electrical Engineering of the Delft University of Technology. The research described in this thesis was carried out in this period.