

UNIVERSITY OF ZAGREB
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING
SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Miroslav Vrankić

**TWO-DIMENSIONAL ADAPTIVE
PERFECT RECONSTRUCTION
FILTER BANKS
USING LIFTING SCHEME**

**DVODIMENZIONALNI ADAPTIVNI
FILTARSKI SLOGOVI
S POTPUNOM REKONSTRUKCIJOM
REALIZIRANI METODOM PODIZANJA**

MASTER THESIS
MAGISTARSKI RAD

Zagreb, 2003

The master thesis has been realized at the Department of Electronic Systems and Information Processing, Faculty of Electrical Engineering and Computing, University of Zagreb.

Magistarski rad je izrađen na Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Mentor: Dr.sc. Damir Seršić, assistant professor

Mentor: Doc.dr.sc. Damir Seršić

The master thesis consists of 127 pages

Magistarski rad ima 127 stranica

Master thesis evaluation committee:

1. Professor dr. sc. Hrvoje Babić, Faculty of Electrical Engineering and Computing, University of Zagreb - chair
2. Assistant professor dr. sc. Damir Seršić, Faculty of Electrical Engineering and Computing, University of Zagreb - mentor
3. Professor dr. sc. Božidar Vojnović, Ruđer Bošković Institute

Master thesis defence committee:

1. Professor dr. sc. Hrvoje Babić, Faculty of Electrical Engineering and Computing, University of Zagreb - chair
2. Assistant professor dr. sc. Damir Seršić, Faculty of Electrical Engineering and Computing, University of Zagreb - mentor
3. Professor dr. sc. Božidar Vojnović, Ruđer Bošković Institute

Master thesis was defended on July 7, 2003.

Povjerenstvo za ocjenu rada:

1. Prof. dr. sc. Hrvoje Babić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu - predsjednik
2. Doc. dr. sc. Damir Seršić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu - mentor
3. Professor dr. sc. Božidar Vojnović, Institut Ruđer Bošković

Povjerenstvo za obranu rada:

1. Prof. dr. sc. Hrvoje Babić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu - predsjednik
2. Doc. dr. sc. Damir Seršić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu - mentor
3. Professor dr. sc. Božidar Vojnović, Institut Ruđer Bošković

Rad je obranjen 7. srpnja 2003. godine.

Contents

1	Introduction	1
1.1	Motivation	1
1.1.1	Seing the Forest and the Leaves at the Same Time	1
1.1.2	Second Generation Wavelets and the Lifting Scheme	2
1.2	Overview of the Thesis	3
2	Wavelet Filter Banks and Multiresolution Analysis	5
2.1	Fundamentals of Filter Banks	5
2.1.1	Downsampling	6
2.1.2	Upsampling	9
2.1.3	The Two Noble Identities	10
2.1.4	Perfect Reconstruction	12
2.1.5	Polyphase Representation	17
2.2	Wavelet Filter Banks	22
2.2.1	Multiresolution Analysis	22
2.2.2	Wavelet Series and Iterated Filter Banks	25
2.3	The Lifting Scheme	29
2.3.1	Dual Lifting Step	29
2.3.2	Primal Lifting Step	31
2.3.3	Alternating Dual and Primal Lifting Step	31
3	Two-Dimensional Nonseparable Wavelet Filter Banks Using Lifting Scheme	33
3.1	Multidimensional Signals and Notation	33
3.2	Nonseparable Versus Separable	34
3.3	Two-Dimensional Sampling	35
3.3.1	Lattices and Dilation Matrix	35
3.3.2	Downsampling	37
3.3.3	Upsampling	39
3.3.4	Simplest Nonseparable Choice: Quincunx Sampling	40

3.4	Quincunx Interpolating Filter Banks	42
3.4.1	Interpolating Filters	42
3.4.2	Neville Filters	42
3.4.3	Construction of Quincunx Neville Filters	42
3.4.4	Predict and Update	44
3.4.5	Vanishing Moments	46
3.4.6	Interchange of Dual and Primal Functions	48
3.4.7	Iterated Filter Banks	50
4	Filter Banks With Variable Parameters	53
4.1	Bad Way To Change Filter Parameters	53
4.1.1	Predict Step	54
4.1.2	Update Step	58
4.2	Good Way To Change Filter Parameters	61
4.2.1	Predict Step	61
4.2.2	Update Step	71
4.3	Importance of Lower Prediction Sections	76
5	Proposed Adaptive Structure	79
5.1	Adaptive Filter Bank Structure	79
5.2	Adaptation Criteria	80
5.2.1	Adapted Filter Parameters and Perfect Reconstruction	82
5.3	Least Squares Adaptation Methods	83
5.3.1	1-D LSW	86
5.3.2	1-D RLS	87
5.3.3	2-D LSW	88
5.3.4	Robust 2-D LSW	88
5.3.5	BLUE	89
5.4	Results	94
5.4.1	Lossy Image Reconstruction	103
6	Conclusion	111
6.1	Future Research	112
	Bibliography	113
	Curriculum Vitae	117
	Abstract	121

Chapter 1

Introduction

1.1 Motivation

The wavelet theory is widely used in the fields of signal and image processing. Actually, the list of areas and applications where wavelets are established as a useful tool is quite long and exceeding by far the signal and image processing applications. The wavelet theory emerged as a result of concurrent research in different fields such as the multiresolution analysis and filter banks.

It can be said that the first step towards wavelets was made by French scientist Joseph Fourier almost two hundred years ago. He proved that any periodic function can be expressed as a linear combination of sines and cosines of various frequencies. The well known Fourier transform [Stein 75] was extended to non-periodic functions but it has one serious disadvantage. It does not contain information about time properties of the analyzed signal. It only gives information about the frequency characteristics of the analyzed signal, but it does not say anything about its existence in time.

One step further are the linear transformations that give the information both on time and frequency properties of the analyzed signal. The two transformations with such properties are the Short Time Fourier Transform and its discrete version, the Gabor Transform. The segmentation of the time-frequency plane obtained by these transforms is linear. There are equal equidistant steps in both the frequency and time.

1.1.1 Seeing the Forest and the Leaves at the Same Time

The main idea behind the wavelet theory is to represent a signal with a linear combination of a number of elementary functions which are localized in time. Furthermore, those functions called wavelets are all of a "constant shape" [Grossmann 84]. It means that all the wavelets are constructed only from one function called *mother wavelet* as

it's scaled and shifted copies. Such a signal expansion is very efficient in representing signals that can be found in the real world, signals with transients and changes localized in time [Mayer 93]. One such "complicated" signal can be represented with few coefficients corresponding to wavelets from different scales and positions. The wavelet decomposition shows very good results in denoising, compression and other areas of the signal and image processing.

The multiresolution analysis [Mallat 89a] is the fundamental concept hidden behind wavelets. The signal is analyzed in different resolutions using wavelets of different scales. It is like looking at the same object from different distances at the same time. By looking at the object from a large distance, one gets the global picture. By looking at the same object by standing very close to it, one becomes able to perceive the tiny details. The wavelet transform provides different views of the signal at the same time, it gives the tiny details together with the very coarse average information. It is like being able to see the whole forest and tiny leaves on the trees at the same time!

The fundamental moment for further evolution of the wavelet theory and applications was when the two different aspects were linked together: wavelets and the filter bank theory [Mallat 89c]. It was shown that the wavelet transform could be implemented using an iterated filter bank of certain properties. After that, calculating the Discrete Wavelet Transform (DWT) became as easy as filtering a signal with a discrete-time filter bank. There were developed very efficient algorithms for calculating the DWT based on a wavelet filter bank implementation [Beylkin 91].

1.1.2 Second Generation Wavelets and the Lifting Scheme

There has been another filter bank construction called *lifting scheme* [Sweldens 95b, Sweldens 95a, Sweldens 96] presented relatively recently. Beside allowing for a numerically very efficient calculation of the wavelet transform, it automatically provides the perfect reconstruction property of the filter banks. Therefore, it has become an ideal tool for constructing second generation wavelets, wavelets that are not necessarily translates and dilates of one fixed function but instead they can be adapted to the properties of the analyzed signal.

In this thesis we give a construction of adaptive two-dimensional filter banks based on a lifting scheme [Vrankić 02]. The construction is based on a one-dimensional adaptive filter bank structure proposed by Seršić [Seršić 99, Seršić 00]. Seršić proposed a wavelet filter bank structure based on the lifting scheme that adapts to the properties of the analyzed signal. The obtained filter bank retains good properties of the wavelet decomposition while allowing for a number of filter parameters to be freely tuned to the properties of the analyzed signal. In the resulting wavelet decomposition, the

mother wavelet function is not constant. Rather, it changes in time in accordance with the properties of the analyzed signal. Such a construction gave superior results to the fixed wavelets.

Filter banks discussed in this thesis are a two-dimensional generalization of the filter bank structures previously reported by Seršić.

1.2 Overview of the Thesis

In chapter 2 we give an introduction to the wavelet filter banks and the multiresolution analysis. There are basic concepts explained for one-dimensional signals since the extension to the two-dimensional case is pretty straightforward.

In the first part of chapter 3 there are presented the basics of two-dimensional filter banks. The second part of the chapter presents a construction of the two-dimensional filter bank based on a lifting scheme with nonseparable quincunx polyphase decomposition of the analyzed image.

The filter bank analyzed in chapter 3 has been modified to become variable and therefore adaptable to the local image properties. Chapter 4 gives a construction of such a filter bank and analyzes the impact of changes in filter parameters upon the overall filter bank properties. Thus, the framework for later adaptations of the filter bank properties to the analyzed image is provided. The adaptation of filter parameters is analyzed in chapter 5. There are various adaptation methods discussed and tested on different synthetic and real-world images. Finally, results on the lossy image reconstruction are presented.

Chapter 2

Wavelet Filter Banks and Multiresolution Analysis

In this chapter basic concepts are explained necessary for the understanding of multi-rate systems and wavelet analysis. These concepts are discussed for one-dimensional signals since mathematical theory is simpler and still sufficient for further understanding of the corresponding two-dimensional (or even d -dimensional) analysis. In the next chapter these results will be generalized for two-dimensional signals, i.e. images.

In section 2.1 fundamentals of perfect reconstruction filter banks and multirate signal processing are presented. In section 2.2 we describe multiresolution analysis and outline the connection between wavelet transform and discrete filter banks. Finally, in section 2.3 we describe the lifting scheme, an efficient and very flexible method for building wavelet filter banks. It is the basis for two-dimensional adaptive filter banks proposed in subsequent chapters of the thesis.

2.1 Fundamentals of Filter Banks

Basic operations being performed in the multirate signal processing systems are decimation and interpolation. Decimation operators are called *decimators* while operators crucial for interpolation are called *expanders*. They cause the sampling rate alteration,

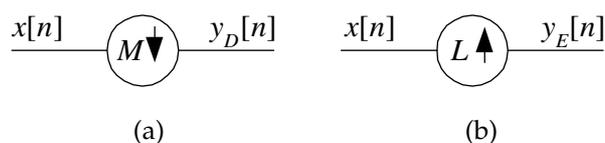


Figure 2.1: (a) Decimator. (b) Expander.

thus making the corresponding systems time-variant. A decimator as a downsampling

operator reduces the sampling rate (thins out samples in time) while an expander as an upsampling operator increases the sampling rate (increases the number of samples in time by adding zeros).

2.1.1 Downsampling

The M -fold decimator forwards every M -th sample of the input sequence $x[n]$ to its output as $y_D[n]$. All the other input samples are being discarded.

$$y_D[n] = x[Mn], \quad \text{for every } n \in \mathcal{Z}. \quad (2.1)$$

Decimation for $M = 2$ is shown in figure 2.2. As it can be seen, there is two times

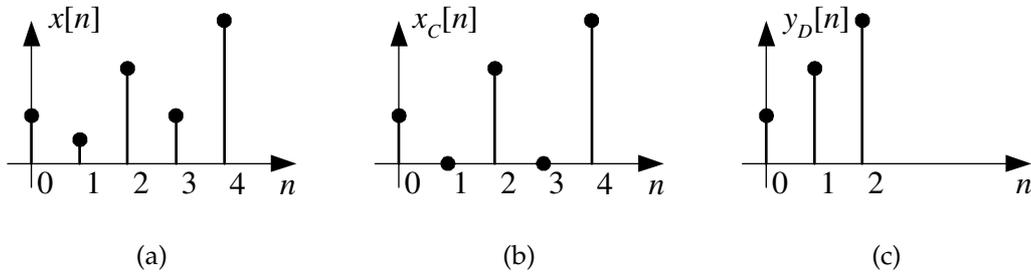


Figure 2.2: Downsampling for $M = 2$. (a) Input signal. (b) Input signal multiplied by a comb sequence. (c) Downsampled signal.

smaller amount of samples at the output of the decimator than at its input. The physical time scale being ignored and a digital signal being perceived as a mere sequence of numbers, signal at the output of a decimator looks like a contracted copy of the signal at its input. This contraction of the signal in a discrete time domain results in its expansion in the frequency domain. The relationship between signal $Y_D(e^{j\omega})$ at the decimator's output and signal $X(e^{j\omega})$ at the decimator's input for an arbitrary decimation constant M is

$$Y_D(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j\frac{\omega-2\pi k}{M}}), \quad (2.2)$$

or expressed it the z-domain

$$Y_D(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(z^{\frac{1}{M}} W_M^k), \quad (2.3)$$

where $W_M = e^{-j\frac{2\pi}{M}}$. To prove these equations we define the sequence $x_C[n] = c_M[n]x[n]$, where

$$c_M[n] = \begin{cases} 1 & \text{if } n \text{ is a multiple of } M, \\ 0 & \text{otherwise.} \end{cases} \quad (2.4)$$

is a *comb* sequence. In this way, the $x_C[n]$ contains samples of $x[n]$ only for those n values which are multiples of M (see figure 2.2). Any other samples equal zero. Let us express the $Y_D(e^{j\omega})$ in terms of $x_C[n]$. Now,

$$\begin{aligned} Y_D(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} y_D[n]e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x[Mn]e^{-j\omega n} \\ &= \sum_{n=-\infty}^{\infty} x_C[Mn]e^{-j\omega n} = \sum_{k=-\infty}^{\infty} x_C[k]e^{-j\omega \frac{k}{M}} \end{aligned} \quad (2.5)$$

which results in

$$Y_D(e^{j\omega}) = X_C(e^{j\frac{\omega}{M}}). \quad (2.6)$$

Knowing that

$$c_M[n] = \frac{1}{M} \sum_{k=0}^{M-1} e^{j\frac{2\pi}{M}kn} = \frac{1}{M} \sum_{k=0}^{M-1} W_M^{-kn} \quad (2.7)$$

the $X_C(e^{j\omega})$ can be expressed in terms of $X(e^{j\omega})$ as

$$\begin{aligned} X_C(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} x_C[n]e^{-j\omega n} = \sum_{n=-\infty}^{\infty} c_M[n]x[n]e^{-j\omega n} \\ &= \sum_{n=-\infty}^{\infty} \frac{1}{M} \sum_{k=0}^{M-1} W_M^{-kn} x[n]e^{-j\omega n} = \frac{1}{M} \sum_{k=0}^{M-1} \sum_{n=-\infty}^{\infty} x[n](e^{j\omega} W_M^k)^{-n}. \end{aligned} \quad (2.8)$$

Since $\sum_{n=-\infty}^{\infty} x[n](e^{j\omega} W_M^k)^{-n} = X(e^{j\omega} W_M^k)$

$$X_C(e^{j\omega}) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j\omega} W_M^k). \quad (2.9)$$

By using the equation 2.6 now,

$$Y_D(e^{j\omega}) = X_C(e^{j\frac{\omega}{M}}) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j\frac{\omega}{M}} W_M^k) = \frac{1}{M} \sum_{k=0}^{M-1} X(e^{j\frac{\omega}{M}} e^{-j\frac{2\pi}{M}k}), \quad (2.10)$$

which results the same as the equation 2.2. Equation 2.2 shows that the frequency spectrum of a decimated signal $Y_D(e^{j\omega})$ is obtained by adding M copies of the input signal's spectrum $X(e^{j\omega})$. Prior to the addition, each copy of $X(e^{j\omega})$ is M -times extended in frequency and shifted to the right for $\Delta\omega = 2\pi$ more than the preceding copy. Finally, the resulting summation is divided by M . For $M = 2$

$$Y_D(e^{j\omega}) = \frac{1}{2}[X(e^{j\frac{\omega}{2}}) + X(e^{j\frac{\omega-2\pi}{2}})] = \frac{1}{2}[X(e^{j\frac{\omega}{2}}) + X(-e^{j\frac{\omega}{2}})], \quad (2.11)$$

$$Y_D(z) = \frac{1}{2}[X(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}})]. \quad (2.12)$$

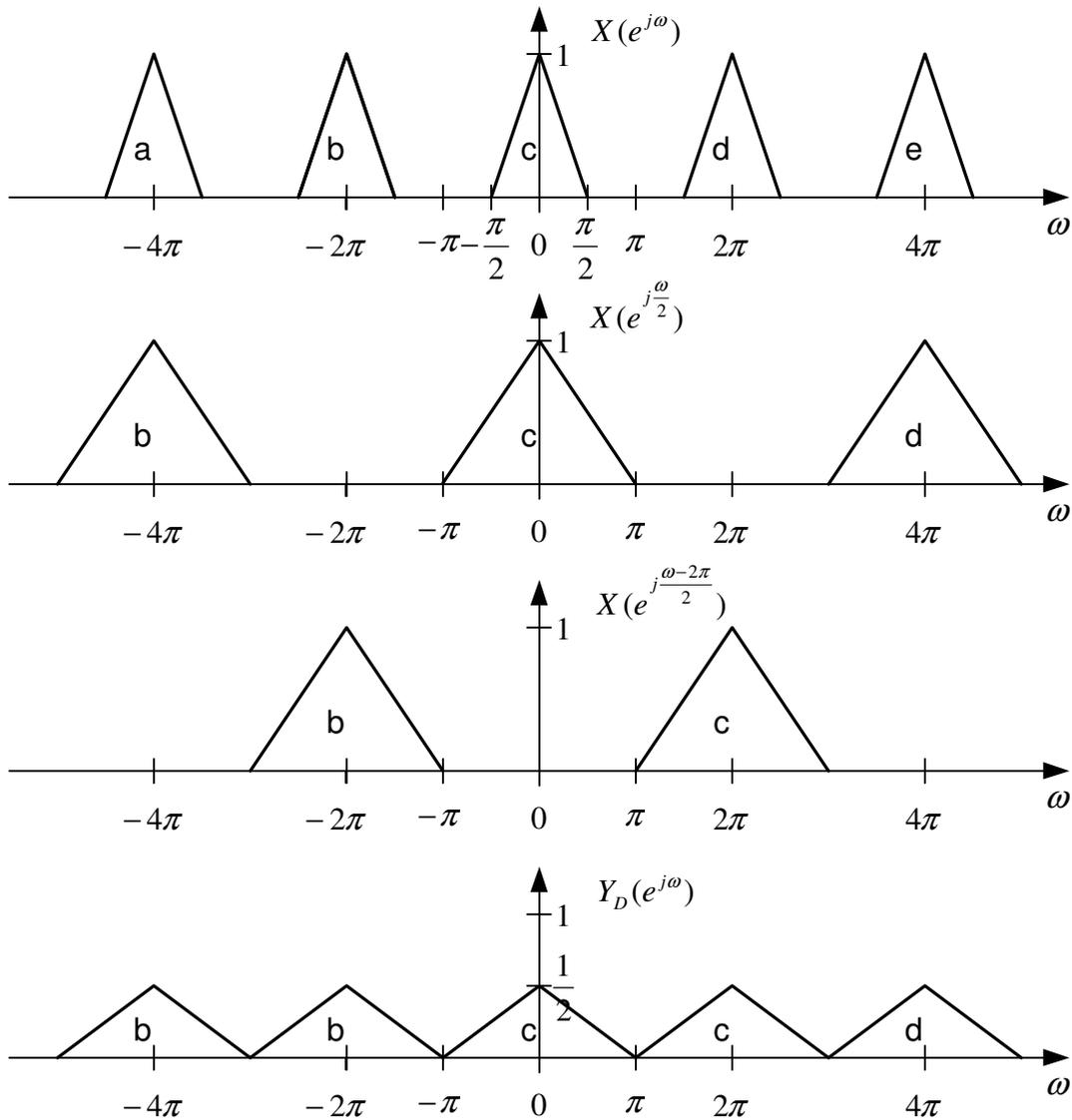


Figure 2.3: Decimation effect in the frequency domain for $M = 2$.

Figure 2.3 shows the effect of spectrum stretching after a 2-fold decimation. It can be seen that the final spectrum of the decimated signal consists of two copies of the input signal's spectrum: $X(e^{j\frac{\omega}{2}})$ and $X(e^{j\frac{\omega-2\pi}{2}})$, the second one being shifted in frequency by $\Delta\omega = 2\pi$. The spectrum bandwidth of the input signal was restricted to $\langle -\frac{\pi}{2}, \frac{\pi}{2} \rangle$ so no overlapping of $X(e^{j\frac{\omega}{2}})$ and $X(e^{j\frac{\omega-2\pi}{2}})$ occurred. If the input signal had not been bandlimited, the overlapping would have been inevitable. This overlapping effect is called aliasing. Once the aliasing happens, the reconstruction of the original input signal from its decimated version becomes impossible. To avoid aliasing caused by decimation by an arbitrary constant M , the spectrum bandwidth of the input signal must be restricted to $|\omega| < \frac{\pi}{M}$.

2.1.2 Upsampling

Contrary to the decimator, the expander increases the number of samples in the output signal $y_E[n]$ by adding $L - 1$ zeros after every sample of the input signal $x[n]$.

$$y_E[n] = \begin{cases} x[\frac{n}{L}] & \text{provided that } n \text{ is multiple of } L, \\ 0 & \text{otherwise.} \end{cases} \quad (2.13)$$

Evidently, this operation does not cause any loss of information. The information con-

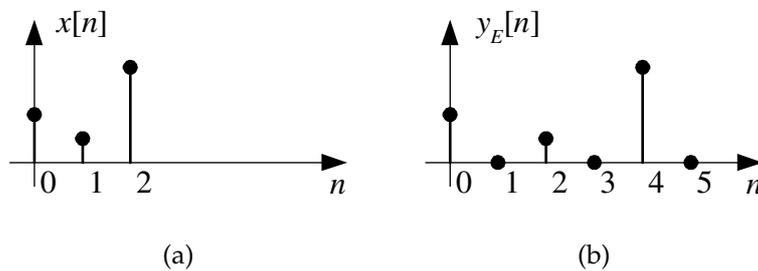


Figure 2.4: Upsampling for $L = 2$. (a) Input signal. (b) Upsampled signal.

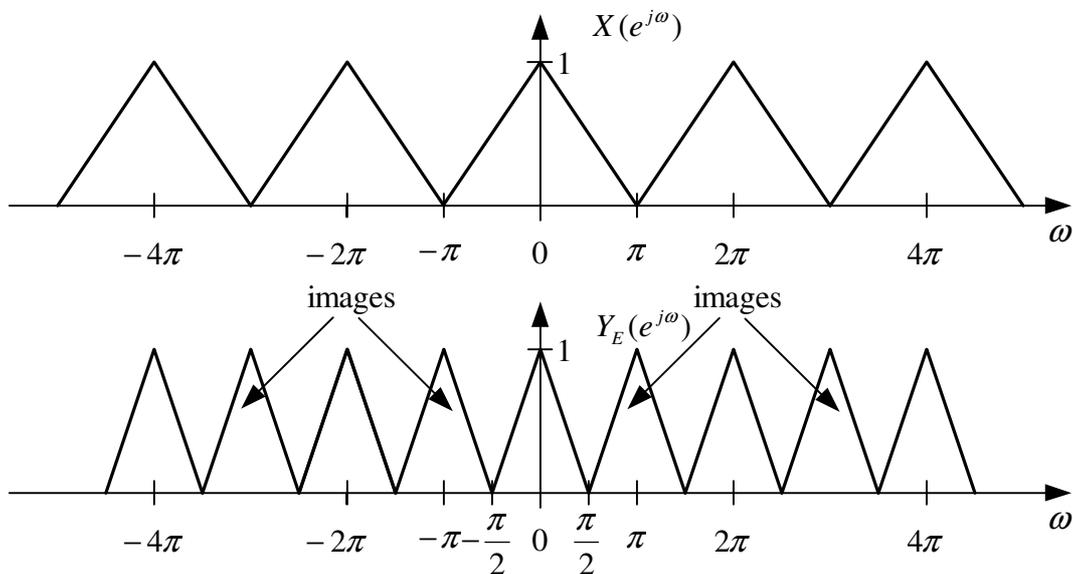


Figure 2.5: Upsampling effect in the frequency domain for $L = 2$.

tained in the input signal will be preserved regardless of the input signal's spectrum or the upsampling constant L . The upsampling operation of an expander with $L = 2$ is shown in figure 2.4. By using the relation 2.13

$$Y_E(e^{j\omega}) = \sum_{n=-\infty}^{\infty} y_E[n]e^{-j\omega n} = \sum_{k=-\infty}^{\infty} y_E[kL]e^{-j\omega kL} = \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega kL} \quad (2.14)$$

we show that

$$Y_E(e^{j\omega}) = X(e^{j\omega L}), \quad (2.15)$$

or in the z-domain

$$Y_E(z) = X(z^L). \quad (2.16)$$

Equation 2.15 shows that an L -fold upsampling operation causes the input signal's spectrum to shrink L times. It looks like some additional images of the original spectrum appeared. Figure 2.5 shows this effect for $L = 2$. There is one additional image in $Y_E(e^{j\omega})$ between 0 and 2π . Generally, for an L -fold interpolation, $L - 1$ images would appear between 0 and 2π .

2.1.3 The Two Noble Identities

The input signal being applied to a typical filter bank (see figure 2.8) is first being filtered and then decimated. Obviously, this approach is computationally inefficient. It doesn't seem reasonable to calculate samples that will be eventually discarded. It would be much better to perform decimation in the first place and then apply filtering on the reduced number of samples. In general, it is not possible.

First Noble Identity

Interchanging of decimation and filtering is possible only in one special case: provided that a filter's impulse response $h[n]$ has $M - 1$ zero coefficients after every nonzero coefficient. In such case the M -fold decimation can be done first and then followed by filtering with $H_{nz}(z)$. This filter's impulse response consists of nonzero coefficients of the original filter's impulse response $h[n]$ (see figure 2.6(b)). Expressed in the z-domain $H(z) = H_{nz}(z^M)$, where index nz stands for *nonzero*. The interchange of decimation and filtering (and swapping of filters), popularly called the *First Noble Identity* is shown in figure 2.6(a). It is easy to prove this identity. In a typical filter bank, the signal is first being filtered with $H(z) = H_{nz}(z^M)$. The filter's output $H_{nz}(z^M)X(z)$ is then being decimated (equation 2.3), resulting in

$$Y(z) = \frac{1}{M} \sum_{k=0}^{M-1} H_{nz}((z^{\frac{1}{M}} W_M^k)^M) X(z^{\frac{1}{M}} W_M^k) = H_{nz}(z) \frac{1}{M} \sum_{k=0}^{M-1} X(z^{\frac{1}{M}} W_M^k). \quad (2.17)$$

In the second case the input signal is being decimated first. The decimated signal

$$X_D(z) = \frac{1}{M} \sum_{k=0}^{M-1} X(z^{\frac{1}{M}} W_M^k) \quad (2.18)$$

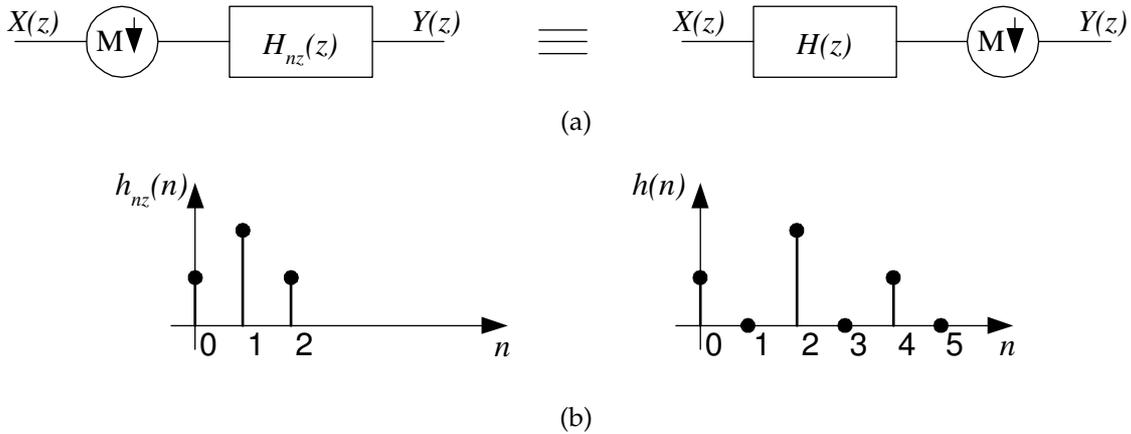


Figure 2.6: (a) First Noble Identity: placing a decimator ahead of the filter. (b) Example of the new filter's impulse response $h_{nz}[n]$ and the original filter's impulse response $h[n]$ for $M = 2$.

is then being filtered with $H_{nz}(z)$ giving the same result as equation 2.17.

This interchange of decimation and filtering is the very essence of the polyphase representation that is discussed in section 2.1.5. The idea is to break a filter into a set of M parallel filters (see figure 2.13) that comply with *Noble Identity's* requirements, i.e. their impulse responses have $M - 1$ zeros after every nonzero coefficient.

Second Noble Identity

It is very useful to place an expander after the filter on the reconstruction side of the filter bank. That is possible only if the filter $G(z)$ has $L - 1$ zeros after every nonzero coefficient in its impulse response $g[n]$. In that case the interchange of upsampling and filtering is done as shown in figure 2.7, where $G(z) = G_{nz}(z^L)$. This is the so called



Figure 2.7: Second noble identity: placing expander after the filter.

Second Noble Identity. To prove, first consider the standard form of a filter bank where the input signal is being upsampled first. The upsampled signal $X(z^L)$ is then being filtered with $G(z) = G_{nz}(z^L)$ which gives

$$Y(z) = G_{nz}(z^L)X(z^L). \quad (2.19)$$

In the polyphase case the output of the filter $G_{nz}(z)X(z)$ is being upsampled giving the same Y as in equation 2.19.

2.1.4 Perfect Reconstruction

A two-channel filter bank as shown in figure 2.8 consists of two parts: the analysis and the synthesis bank. Filters H_0 and H_1 are the analysis filters, while filters G_0 and G_1 are the synthesis filters.

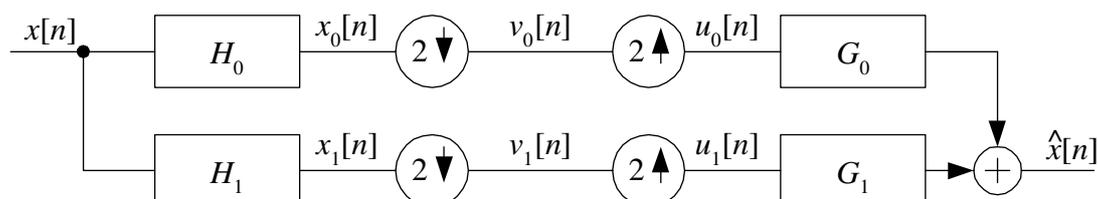


Figure 2.8: Two-channel filter bank.

If the reconstructed sequence at the output of the filter bank $\hat{x}[n]$ is a delayed copy of the signal at the filter bank's input $x[n]$, i.e.

$$\hat{x}[n] = x[n - n_0], \quad (2.20)$$

then it is said that the filter bank has the perfect reconstruction property. In a typical two-channel filter bank, H_0 is a low-pass filter and H_1 is a high-pass filter. Thus, the

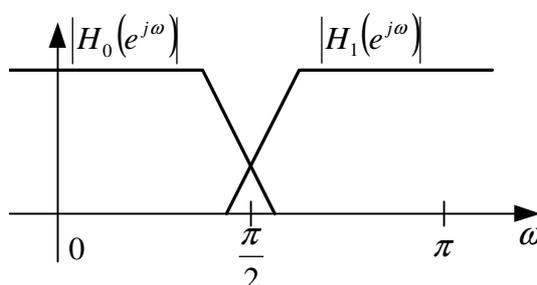


Figure 2.9: The analysis bank filters' frequency responses.

input signal is being split in two bands, a low-frequency and a high-frequency band. Frequency responses of the two analysis filters H_0 and H_1 are shown in 2.9. These filters' frequency responses overlap because of their final stopband attenuation and transition bandwidth length. As a consequence, signals $x_0[n]$ and $x_1[n]$ aren't properly bandlimited and after a decimation an aliasing in $v_0[n]$ and $v_1[n]$ will be inevitable. Nevertheless, a perfect reconstruction is still possible with suitable selection of G_0 and G_1 filters.¹ The idea is to choose appropriate synthesis filters so that the aliases from

¹Of course, it is possible to make non overlapping frequency responses of the analysis filters (and therefore avoid aliasing), e.g. by using filters with very short transition bandwidth, but this increases the cost of filter implementation. Therefore, the use of simpler filters with overlapping frequency responses and application of alias cancellation afterwards is preferable.

the two channels cancel each other.

Expressed in the z-domain, signals at the output of analysis filters are given by

$$X_0(z) = H_0(z)X(z), \quad (2.21a)$$

$$X_1(z) = H_1(z)X(z). \quad (2.21b)$$

They are shown in figures 2.10(b) and 2.11(b) respectively. These signals are then being decimated by a factor of two (equation 2.12) resulting in

$$V_0(z) = \frac{1}{2}[X_0(z^{\frac{1}{2}}) + X_0(-z^{\frac{1}{2}})], \quad (2.22a)$$

$$V_1(z) = \frac{1}{2}[X_1(z^{\frac{1}{2}}) + X_1(-z^{\frac{1}{2}})]. \quad (2.22b)$$

The second term (the one with $-z^{\frac{1}{2}}$) in both equations causes aliasing. In the frequency domain this alias-term is a copy of the two times extended spectrum of X_0 or X_1 that has been shifted by 2π . The outputs of the decimator for both channels are shown in figures 2.10(c) and 2.11(c) with alias-terms marked gray.

A two-fold downsampling is followed by a two-fold upsampling, with one zero being inserted after every sample of $v_0[n]$ and $v_1[n]$. By using equation 2.16, the upsampled signals are expressed as

$$U_0(z) = V_0(z^2) = \frac{1}{2}[X_0(z) + X_0(-z)], \quad (2.23a)$$

$$U_1(z) = V_1(z^2) = \frac{1}{2}[X_1(z) + X_1(-z)]. \quad (2.23b)$$

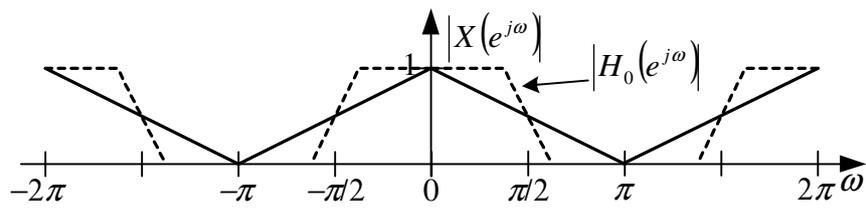
The effects of upsampling are shown in figures 2.10(d) and 2.11(d). It can be seen that in the frequency domain the upsampling following a downsampling results in two copies of the original signal (X_0 or X_1). The second copy is shifted by π and it represents the alias-term. In the low-pass channel the alias-term $0.5X_0(-z)$ is positioned in the high-pass frequency region. Therefore, to cancel it, a low-pass synthesis filter G_0 should be used. In the high-pass channel, the alias-term $0.5X_1(-z)$ dominates in the low-pass frequency region. To cancel it, G_1 should be a high-pass filter.

The reconstructed signal is a sum of the synthesis filters' outputs

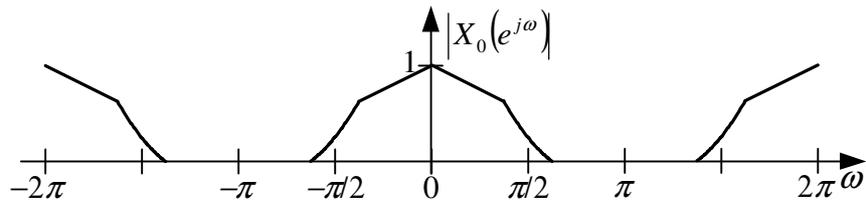
$$\hat{X}(z) = G_0(z)U_0(z) + G_1(z)U_1(z) \quad (2.24)$$

or, by using equations 2.23 and 2.21

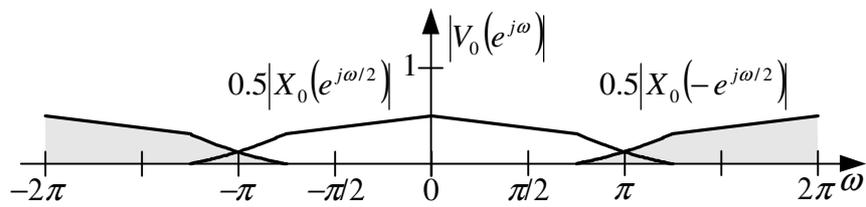
$$\begin{aligned} \hat{X}(z) &= G_0(z)\frac{1}{2}[H_0(z)X(z) + H_0(-z)X(-z)] \\ &+ G_1(z)\frac{1}{2}[H_1(z)X(z) + H_1(-z)X(-z)] \end{aligned} \quad (2.25)$$



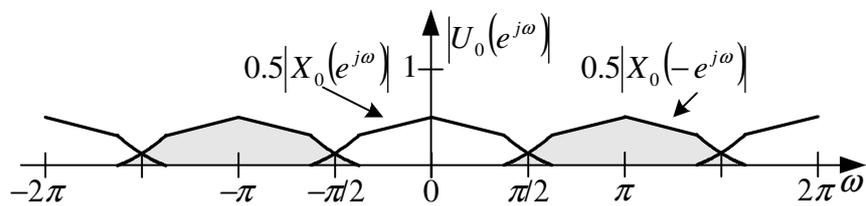
(a) Input signal and the low-pass filter's amplitude response (dashed).



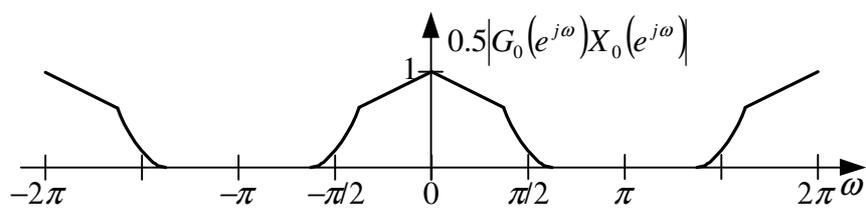
(b) Output of the low-pass filter.



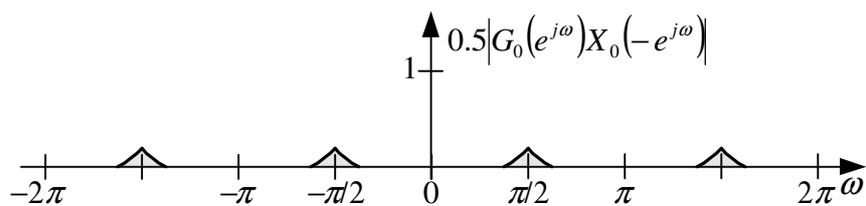
(c) Output of the two-fold decimator.



(d) Output of the two-fold expander.

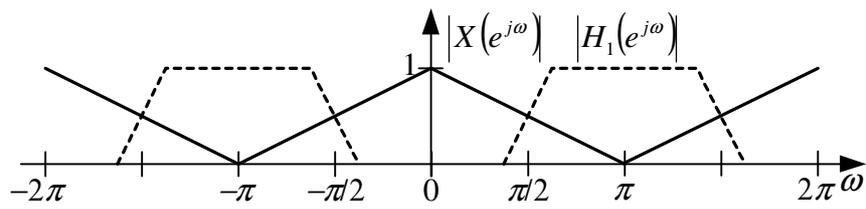


(e) Unaliased component filtered with a low-pass filter G_0 .

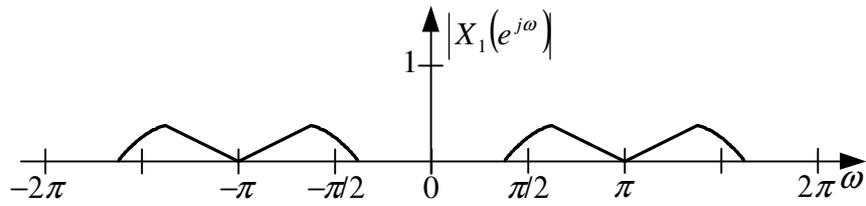


(f) Aliased component filtered with a low-pass filter G_0 .

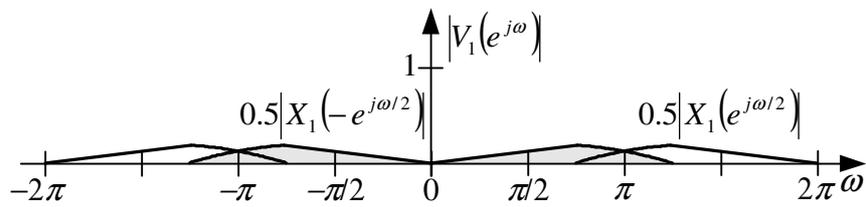
Figure 2.10: Signals in the low-pass channel.



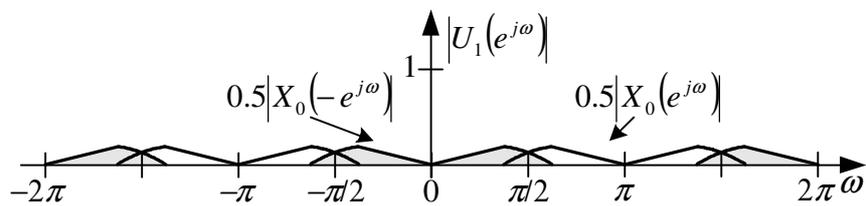
(a) Input signal and the high-pass filter's amplitude response (dashed).



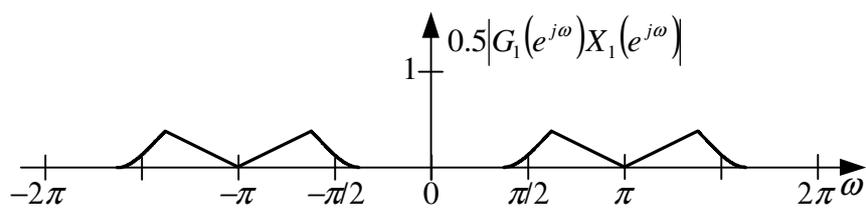
(b) Output of the high-pass filter.



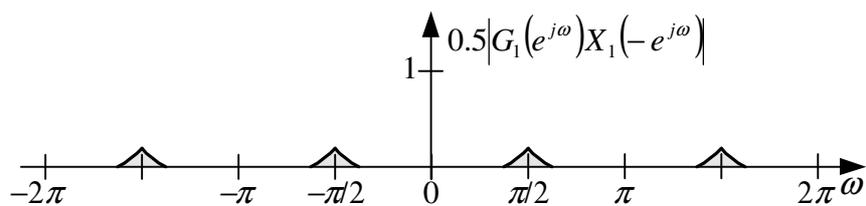
(c) Output of the two-fold decimator.



(d) Output of the two-fold expander.



(e) Unaliased component filtered with a high-pass filter G_1 .



(f) Aliased component filtered with a high-pass filter G_1 .

Figure 2.11: Signals in the high-pass channel.

and than reorganizing

$$\begin{aligned}\hat{X}(z) &= \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)]X(z) \\ &\quad + \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)]X(-z), \\ \hat{X}(z) &= T(z)X(z) + A(z)X(-z).\end{aligned}\tag{2.26}$$

To obtain the perfect reconstruction property of the given filter bank, i.e. to make $\hat{X}(z) = z^{-n_0}X(z)$, filters should be chosen in such a way that

$$A(z) = \frac{1}{2}[H_0(-z)G_0(z) + H_1(-z)G_1(z)] = 0\tag{2.27a}$$

$$T(z) = \frac{1}{2}[H_0(z)G_0(z) + H_1(z)G_1(z)] = z^{-n_0}.\tag{2.27b}$$

Equation 2.27a ensures alias cancellation. It states that the aliased part of the output from G_0 (figure 2.10(f)) will have the same absolute value of the Fourier transform but different sign when compared to the output from G_1 (figure 2.11(f)).

Once the aliasing is removed from the output signal, there are still linear shift-invariant distortions (generally both amplitude and phase distortions) that should be taken care of. Equation 2.27b ensures that there are no amplitude and phase distortions, making $T(e^{j\omega})$, i.e. the transfer function for an unaliased signal a pure delay. It means that the addition of the unaliased signal components from low-pass and high-pass channels' outputs (figures 2.10(f) and 2.11(f)) will result in the same absolute value as $|X(e^{j\omega})|$ and a phase of $X(e^{j\omega})$ that is linearly incremented for $-n_0\omega$.

Modulation Matrices

Equation 2.26 can be written more compactly as

$$\hat{X}(z) = \frac{1}{2} \begin{bmatrix} G_0(z) & G_1(z) \end{bmatrix} \underbrace{\begin{bmatrix} H_0(z) & H_0(-z) \\ H_1(z) & H_1(-z) \end{bmatrix}}_{\mathbf{H}_m(z)} \begin{bmatrix} X(z) \\ X(-z) \end{bmatrix}\tag{2.28}$$

where matrix $\mathbf{H}_m(z)$ is called the *analysis modulation matrix* because it contains modulated versions of the analysis filters. Perfect reconstruction conditions from equations 2.27a and 2.27b can now be stated in terms of modulation matrix $\mathbf{H}_m(z)$:

$$\begin{bmatrix} G_0(z) & G_1(z) \end{bmatrix} \underbrace{\begin{bmatrix} H_0(z) & H_0(-z) \\ H_1(z) & H_1(-z) \end{bmatrix}}_{\mathbf{H}_m(z)} = \begin{bmatrix} 2z^{-n_0} & 0 \end{bmatrix}\tag{2.29}$$

Equation 2.29 can be rewritten as

$$\underbrace{\begin{bmatrix} G_0(z) & G_1(z) \\ G_0(-z) & G_1(-z) \end{bmatrix}}_{\mathbf{G}_m(z)} \underbrace{\begin{bmatrix} H_0(z) & H_0(-z) \\ H_1(z) & H_1(-z) \end{bmatrix}}_{\mathbf{H}_m(z)} = \begin{bmatrix} 2z^{-n_0} & 0 \\ 0 & 2(-z)^{-n_0} \end{bmatrix} \quad (2.30)$$

where matrix $\mathbf{G}_m(z)$ is called the *synthesis modulation matrix*. In case that delay $n_0 = 0$, PR condition can be written as

$$\mathbf{G}_m(z)\mathbf{H}_m(z) = 2\mathbf{I}. \quad (2.31)$$

It is important to note that modulation matrices are redundant because every filter coefficient is included twice.

2.1.5 Polyphase Representation

In order to obtain more efficient filter bank structures and more compact representation of a given filter bank, the polyphase representation is used.

The basic idea of polyphase is to split a filter into a set of M parallel filters. Figure 2.13 shows how to split a two-fold decimation filter into a more efficient structure by using polyphase decomposition. It is the so called Type 1 polyphase. The original filter $H(z)$ is being split into two filters (see figure 2.12):

$$H(z) = H_e(z^2) + z^{-1}H_o(z^2). \quad (2.32)$$

Filter $H_e(z^2)$ contains the even-numbered impulse response coefficients of $H(z)$ in-

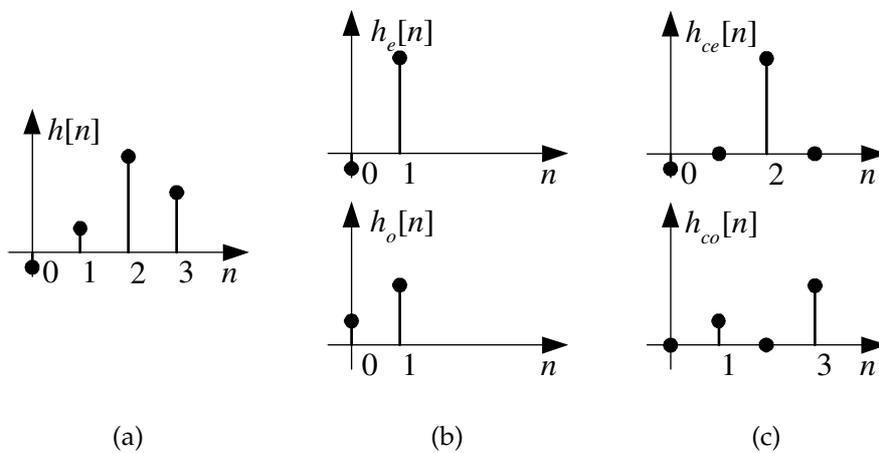


Figure 2.12: Original filter's impulse response (a) is split into "even" and "odd" filters (b). Subfigure (c) shows that $H(z) = H_e(z^2) + z^{-1}H_o(z^2)$.

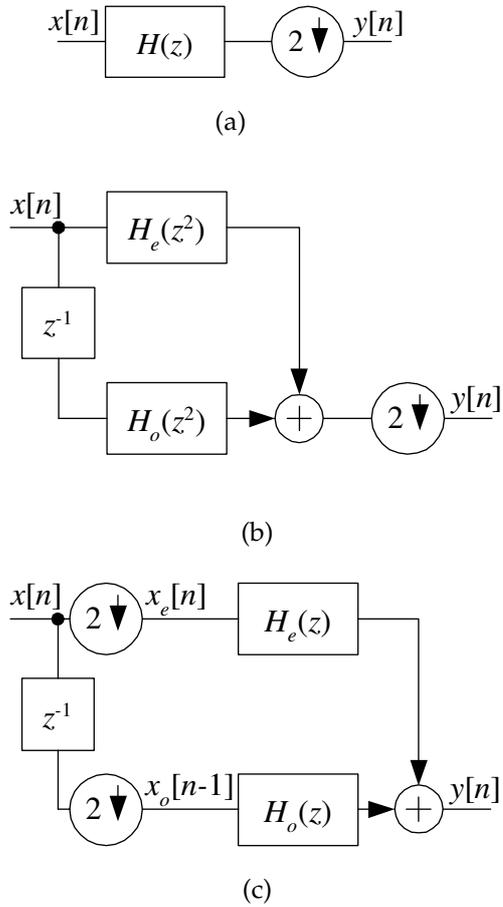


Figure 2.13: Decimation filter (a) can be implemented more efficiently by using Type 1 polyphase (b) and then applying the First Noble Identity (c).

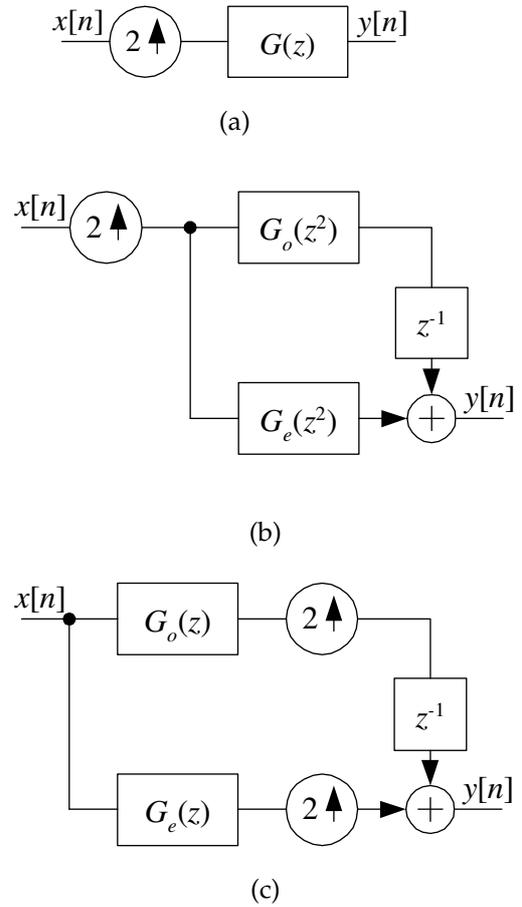


Figure 2.14: Interpolation filter (a) can be implemented more efficiently by using Type 2 polyphase (b) and then applying the Second Noble Identity (c).

terpolated with zeros, one zero following after every coefficient. $H_o(z^2)$ contains the odd-numbered impulse response coefficients of $H(z)$ interpolated with zeros. Such zero-padded filters allow us to interchange decimation and filtering by using the First Noble Identity (figure 2.6(a)). In this way, decimation will be done first and then followed by filtering with $H_e(z)$ and $H_o(z)$. These filters' impulse responses are

$$h_e[n] = h[2n], \quad (2.33a)$$

$$h_o[n] = h[2n + 1]. \quad (2.33b)$$

This final structure is shown in figure 2.13(c). It can be seen that the input signal is split in two signals or phases. The first phase $x_e[n]$ contains even samples of $x[n]$, while the second phase $x_o[n - 1]$ contains odd samples of $x[n]$ delayed by one time unit. The first phase is filtered with "even" filter $H_e(z)$ and the second one with "odd" filter $H_o(z)$.

The final summation gives

$$Y(z) = H_e(z)X_e(z) + H_o(z)z^{-1}X_o(z). \quad (2.34)$$

We can see that filtering is done in parallel by using two filters that are two times shorter than the original filter. With downsampling being done first, no redundant calculations are to be performed and the input rate for the filters is halved. By using the same computational power, we are allowed to calculate the output two times faster than by the direct implementation.

Similar benefits can be obtained for the interpolation filter. An efficient interpolation structure shown in figure 2.14 is realized by using Type 2 polyphase (now the delay is put in the first branch) and the Second Noble Identity (figure 2.7).

Polyphase Representation of Filter Banks

Let us show now how these concepts can be applied to filter banks. Firstly, let us consider the analysis filter bank shown in figure 2.15(a). Both analysis filters can be split into their polyphase equivalent structures in the way shown in figure 2.13(c). The outputs from both channels can be expressed now as

$$V_0(z) = H_{0e}(z)X_e(z) + H_{0o}(z)z^{-1}X_o(z), \quad (2.35a)$$

$$V_1(z) = H_{1e}(z)X_e(z) + H_{1o}(z)z^{-1}X_o(z), \quad (2.35b)$$

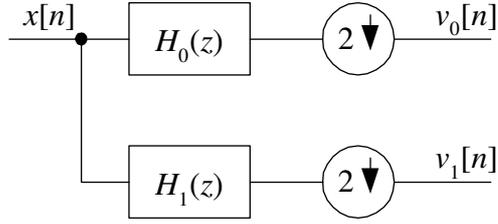
or by using a matrix notation

$$\begin{bmatrix} V_0(z) \\ V_1(z) \end{bmatrix} = \underbrace{\begin{bmatrix} H_{0e}(z) & H_{0o}(z) \\ H_{1e}(z) & H_{1o}(z) \end{bmatrix}}_{\mathbf{H}_p(z)} \begin{bmatrix} X_e(z) \\ z^{-1}X_o(z) \end{bmatrix}, \quad (2.36)$$

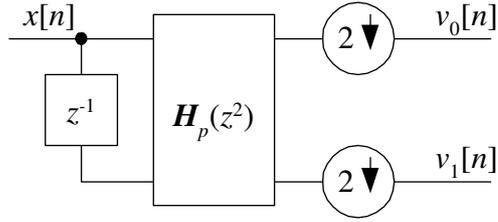
where $\mathbf{H}_p(z)$ is the polyphase matrix. The first row of the matrix represents polyphase components of the filter $H_0(z)$, while the second row represents polyphase components of the filter $H_1(z)$. This compact representation is used in a filter bank structure shown in figure 2.15(c). In general, the M -channel filter bank structure will be represented with a $M \times M$ polyphase component matrix where $H_{ij}(z)$ is the j -th polyphase component of the i -th filter.

The synthesis filter bank shown in figure 2.16(a) can also be realized with a more efficient structure. The synthesis filters are going to be split by using a Type 2 polyphase representation. Now the filter bank's output can be expressed in terms of the filters' polyphase components:

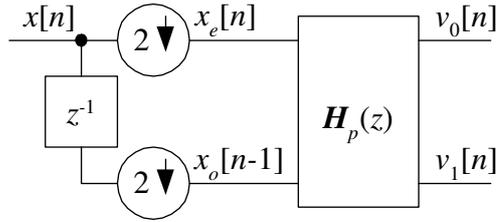
$$\begin{aligned} \hat{X}(z) &= G_0(z)V_0(z^2) + G_1(z)V_1(z^2) \\ &= (z^{-1}G_{0o}(z^2) + G_{0e}(z^2))V_0(z^2) + (z^{-1}G_{1o}(z^2) + G_{1e}(z^2))V_1(z^2). \end{aligned} \quad (2.37)$$



(a)

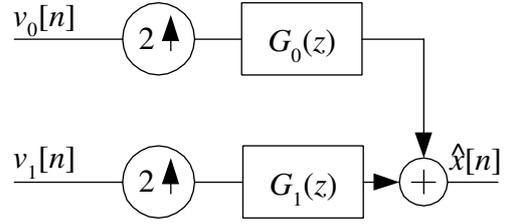


(b)

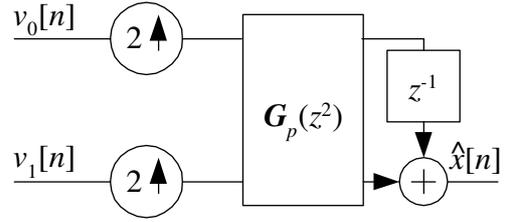


(c)

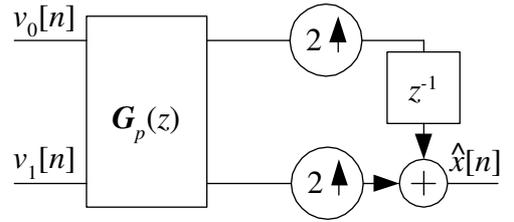
Figure 2.15: From direct form to Type 1 polyphase representation of the analysis filter bank.



(a)



(b)



(c)

Figure 2.16: From direct form to Type 2 polyphase representation of the synthesis filter bank.

This equation can be written in a matrix form as

$$\hat{X}(z) = \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \underbrace{\begin{bmatrix} G_{0o}(z^2) & G_{1o}(z^2) \\ G_{0e}(z^2) & G_{1e}(z^2) \end{bmatrix}}_{\mathbf{G}_p(z^2)} \begin{bmatrix} V_0(z^2) \\ V_1(z^2) \end{bmatrix}, \quad (2.38)$$

where

$$\mathbf{G}_p(z) = \begin{bmatrix} G_{0o}(z) & G_{1o}(z) \\ G_{0e}(z) & G_{1e}(z) \end{bmatrix} \quad (2.39)$$

is the synthesis polyphase component matrix. Now the polyphase components for each filter are placed in appropriate columns of this matrix (not in rows as for $\mathbf{H}_p(z)$). Since the delay is put in the first channel, polyphase components in the first row (the ones corresponding to the first channel) are the odd ones, and polyphase components in the second row are the even ones. The filter bank structure using $\mathbf{G}_p(z)$ is shown

in figure 2.16(c). \hat{X} can now be connected to polyphase components of X in terms of polyphase matrices:

$$\hat{X}(z) = \begin{bmatrix} z^{-1} & 1 \end{bmatrix} \mathbf{G}_p(z^2) \mathbf{H}_p(z^2) \begin{bmatrix} X_e(z^2) \\ z^{-2} X_o(z^2) \end{bmatrix}, \quad (2.40)$$

and the PR condition can be stated as

$$\mathbf{G}_p(z) \mathbf{H}_p(z) = \mathbf{I}. \quad (2.41)$$

If the equation 2.41 stands, the reconstructed signal

$$\hat{X}(z) = z^{-1} X_e(z^2) + z^{-2} X_o(z^2) = z^{-1} X(z) \quad (2.42)$$

is a delayed version of the input signal $X(z)$.

Relation Between Polyphase and Modulation Matrices

For the first channel analysis filter H_0 , the following equations stand:

$$H_{0e}(z^2) = \frac{1}{2}(H_0(z) + H_0(-z)), \quad (2.43a)$$

$$z^{-1} H_{0o}(z^2) = \frac{1}{2}(H_0(z) - H_0(-z)). \quad (2.43b)$$

Same relations can be written for all the other filters. More compact representation of these equations for the analysis filter bank can be written in a matrix form as

$$\underbrace{\begin{bmatrix} H_{0e}(z^2) & H_{0o}(z^2) \\ H_{1e}(z^2) & H_{1o}(z^2) \end{bmatrix}}_{\mathbf{H}_p(z^2)} \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix} = \frac{1}{2} \underbrace{\begin{bmatrix} H_0(z) & H_0(-z) \\ H_1(z) & H_1(-z) \end{bmatrix}}_{\mathbf{H}_m(z)} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad (2.44)$$

This is the relation between modulation and polyphase matrices of the analysis bank.

The relation between modulation and polyphase matrices for the synthesis bank is

$$\begin{bmatrix} z^{-1} & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} G_{0o}(z^2) & G_{1o}(z^2) \\ G_{0e}(z^2) & G_{1e}(z^2) \end{bmatrix}}_{\mathbf{G}_p(z^2)} = \frac{1}{2} \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} G_0(z) & G_1(z) \\ G_0(-z) & G_1(-z) \end{bmatrix}}_{\mathbf{G}_m(z)}. \quad (2.45)$$

2.2 Wavelet Filter Banks

The wavelet analysis described in the following sections can be very efficiently implemented by using a filter bank of certain properties [Mallat 89c]. In the following sections we give an introduction to the wavelet theory and show the connection between the wavelet theory and filter banks.

2.2.1 Multiresolution Analysis

Multiresolution analysis (MRA) is a powerful concept that is very helpful in analyzing wavelet filter banks. Mathematical theory of multiresolution analysis will help us analyze the decomposition of a signal into a sequence of coarser approximation and detail (difference) signals that are mutually orthogonal. This orthogonality ensures that in these signals there is no redundancy of representation of the original signal. Every level of decomposition results in one approximation and one detail signal. The obtained approximation signal is further decomposed in a coarser approximation and a difference signal. The reconstruction of the original signal is done by successively adding finer detail signals to the coarsest approximation signal, and every reconstruction level improves resolution of the signal by a factor of two.

Multiresolution analysis was introduced by [Mallat 89b] and [Meyer 90]. The MRA of $L^2(\mathbf{R})$ (space of square-integrable one-dimensional functions) is based on a sequence of nested closed subspaces

$$\dots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \dots \quad (2.46)$$

with properties of completeness, scale invariance, shift invariance and the existence of a basis. The completeness property states that

$$\overline{\bigcup_{j \in \mathbf{Z}} V_j} = L^2(\mathbf{R}) \quad \text{and} \quad \bigcap_{j \in \mathbf{Z}} V_j = \{0\}. \quad (2.47)$$

The scale invariance ensures that

$$f(t) \in V_j \iff f(2t) \in V_{j-1}. \quad (2.48)$$

In other words, if a function $f(t)$ belongs to the space V_j then its two times shorter equivalent $f(2t)$ belongs to the space V_{j-1} . We say that the subspace in level j is coarser, and the subspace in level $j - 1$ is finer.

The shift invariance property states that if some function $f(t)$ is an element of a subspace V_0 , then its translates for an integer value will also be elements of V_0 :

$$f(t) \in V_0 \iff f(t - n) \in V_0, \text{ for all } n \in \mathbf{Z}. \quad (2.49)$$

Finally, the V_0 has an orthonormal basis

$$\{\varphi(t - n) \mid n \in \mathbf{Z}\}. \quad (2.50)$$

Function $\varphi(t)$ whose translated versions form the basis in the central space V_0 is called the *scaling function*. The orthonormality states that

$$\int_{-\infty}^{\infty} \varphi^*(t) \varphi(t - n) dt = \delta[n], \quad (2.51)$$

or expressed in the Fourier domain

$$\sum_{k=-\infty}^{\infty} |\Phi(\omega + 2k\pi)|^2 = 1. \quad (2.52)$$

From relations 2.48, 2.49 and 2.50 it follows that a set of functions $\{\sqrt{2^j} \varphi(2^j t - n) \mid n \in \mathbf{Z}\}$, i.e. functions that are 2^j times shorter equivalents of the original scaling function $\varphi(t)$ normalized by $\sqrt{2^j}$, form the basis of V_{-j} .

Based on the above mentioned properties of the multiresolution analysis, it can be written

$$\varphi(t) = \sum_{n=-\infty}^{\infty} g_0[n] \sqrt{2} \varphi(2t - n) = \sqrt{2} \sum_{n=-\infty}^{\infty} g_0[n] \varphi(2t - n). \quad (2.53)$$

This equation is known as the *dilation equation*, *refinement equation* or a *two-scale equation*. It states that a scale function (which belongs to V_0) can be expressed as a linear combination of its two times shorter equivalents that form the basis in V_{-1} . Such relation must be possible because V_0 is embedded in V_{-1} , and thence the $\varphi(t)$ belongs not only to V_0 but also to V_{-1} (relation 2.46). In the Fourier domain the dilation equation can be expressed as

$$\Phi(\omega) = \frac{1}{\sqrt{2}} G_0(e^{j\frac{\omega}{2}}) \Phi\left(\frac{\omega}{2}\right), \quad (2.54)$$

where $G_0(e^{j\omega})$ is a discrete-time Fourier transform of $g_0[n]$. By perceiving $g_0[n]$ as an impulse response of a digital filter, a connection between the multiresolution analysis and discrete filter banks has been established.

The Wavelet Representation

Let us now introduce another space W_j . It is called the *detail* space at level j and it is an orthogonal complement of V_j . It represents differences of information between approximation spaces V_j and V_{j-1} . In other words, by adding the detail signal belonging to space W_j to the approximation signal belonging to space V_j we obtain a two times

finer approximation signal that belongs to the space V_{j-1} . For those subspaces we can state that

$$V_j \oplus W_j = V_{j-1}, \quad (2.55)$$

which can be expanded in

$$V_j \oplus W_j \dots \oplus W_3 \oplus W_2 \oplus W_1 = V_0. \quad (2.56)$$

Since $\overline{\bigcup_{j \in \mathbf{Z}} V_j} = L^2(\mathbf{R})$ and $\bigcap_{j \in \mathbf{Z}} V_j = \{0\}$ we can state that

$$L^2(\mathbf{R}) = \bigoplus_{j \in \mathbf{Z}} W_j. \quad (2.57)$$

Function $\psi(t)$ is called the *wavelet function* if a set of functions $\{\psi(t - n) \mid n \in \mathbf{Z}\}$ is a basis of W_0 . Also, family of functions $\{\sqrt{2^j} \psi(2^j t - n) \mid n \in \mathbf{Z}\}$ is a basis of W_{-j} . Furthermore, since the relation 2.57 holds, the family of wavelet functions for all scales and shifts $\{\sqrt{2^j} \psi(2^j t - n) \mid (j, n) \in \mathbf{Z}^2\}$ is an orthonormal basis of $L^2(\mathbf{R})$.

It is worth noting that scaling functions are only orthogonal across each scale separately. For example, all the functions from the set $\{\varphi(t - n) \mid n \in \mathbf{Z}\}$ belonging to the space V_0 are mutually orthogonal but they are not orthogonal to their two times shorter equivalents from V_{-1} . Scaling functions cannot be orthogonal across scales because $\varphi(t)$ belongs to V_0 as well as to V_{-1} and all the other finer subspaces. This important property of orthogonality across scales is introduced by using wavelet subspaces. Wavelet subspaces are mutually orthogonal and therefore all the wavelet functions derived from the original wavelet by shifting (belonging to the same scale) and scaling in time (belonging to a different scale) are mutually orthogonal.

Orthogonality of W_0 and V_0 can be expressed in terms of wavelet and scaling functions as

$$\int_{-\infty}^{\infty} \varphi(t - n)^* \psi(t) dt = 0, \quad (2.58)$$

or expressed in the Fourier domain

$$\sum_{k=-\infty}^{\infty} \Phi^*(\omega + 2k\pi) \Psi(\omega + 2k\pi) = 1. \quad (2.59)$$

Since the wavelet $\psi(t)$ from W_0 belongs also to an approximation subspace V_{-1} it can be expressed as a linear combination of scaling functions that form a basis in V_{-1} :

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} g_1[n] \varphi(2t - n). \quad (2.60)$$

This equation is called the *wavelet equation*. In the Fourier domain it states that

$$\Psi(\omega) = \frac{1}{\sqrt{2}}G_1(e^{j\frac{\omega}{2}})\Phi\left(\frac{\omega}{2}\right). \quad (2.61)$$

By combining equations 2.59, 2.54 and 2.61, $G_1(e^{j\omega})$ can be expressed in terms of $G_0(e^{j\omega})$. To satisfy the orthogonality of W_0 and V_0 , we choose

$$G_1(e^{j\omega}) = -e^{-j\omega}G_0^*(e^{j(\omega+\pi)}). \quad (2.62)$$

Now we can substitute the relation 2.62 in 2.61 to express $\Psi(\omega)$ in terms of $G_0(e^{j\omega})$:

$$\Psi(\omega) = -\frac{1}{\sqrt{2}}e^{-j\frac{\omega}{2}}G_0^*(e^{j\frac{\omega+2\pi}{2}})\Phi\left(\frac{\omega}{2}\right). \quad (2.63)$$

In the time domain the relation 2.62 becomes

$$g_1[n] = -1^n g_0[-n + 1], \quad (2.64)$$

which gives

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} (-1)^n g_0[-n + 1]\varphi(2t - n). \quad (2.65)$$

2.2.2 Wavelet Series and Iterated Filter Banks

Normally, to find projections of an analyzed continuous-time signal $f(t)$ on a set of approximation subspaces $\{V_0, V_1, V_2, \dots\}$ and detail subspaces $\{W_0, W_1, W_2, \dots\}$ one should calculate a number of inner products involving continuous-time integration. Luckily, by using properties of the multiresolution analysis, this won't be necessary. Complete calculation can be done by using a discrete-time algorithm based on the appropriate multiresolution (filter bank) structure. This algorithm is called the *Mallat's algorithm* since it was first introduced by [Mallat 89c].

The first step is to find the projection of $f(t)$ on V_0 :

$$\hat{f}(t) = \sum_{n=-\infty}^{\infty} a^{(0)}[n]\varphi(t - n), \quad \text{where} \quad a^{(0)}[n] = \langle \varphi(t - n), f(t) \rangle. \quad (2.66)$$

Coefficients of the projection onto V_0 , $a^{(0)}[n]$ are called the *discrete approximation* of $f(t)$ at the resolution 2^0 . This inner product can be interpreted as filtering with a low-pass filter $\varphi(t)$ and then sampling the obtained signal at integer time values. Since the integral of $\varphi(t)$ is equal to 1, for a sufficiently fine resolution of V_0 comparing to resolution of $f(t)$, $a^{(0)}[n]$ can be approximated with a sampled version of the input

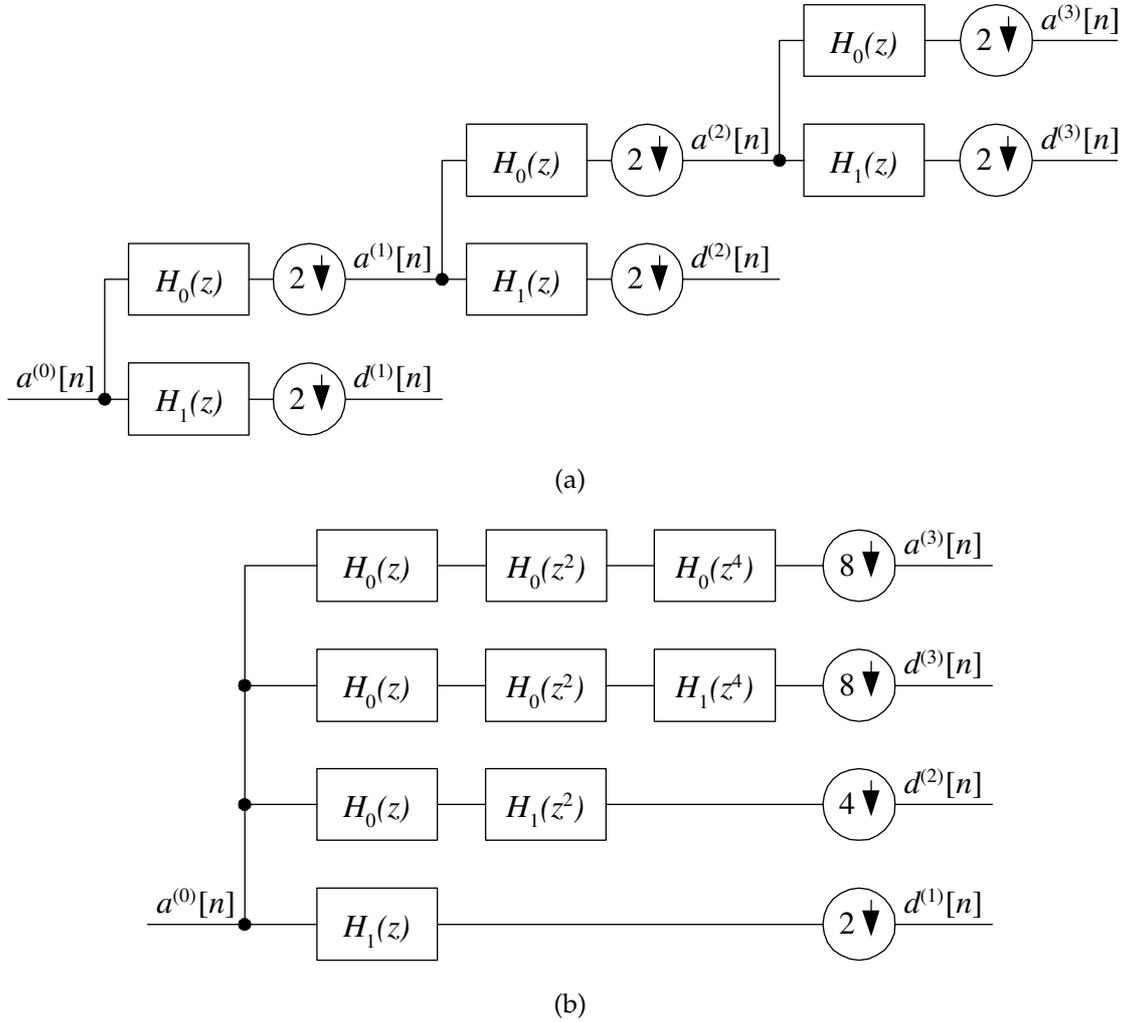


Figure 2.17: (a) Filter bank iterated on a low-pass filter branch implementing MRA. (b) Equivalent structure obtained by using the *First Noble Identity*.

signal $f[n]$. By combining the dilation equation 2.53 with equation 2.66 and by taking $h_0[n] = g_0[-n]$, we obtain:

$$a^{(j)}[n] = \sum_{k=-\infty}^{\infty} h_0[2n - k]a^{(j-1)}[k]. \quad (2.67)$$

It states that discrete approximation coefficients $a^{(j)}[n]$ are obtained by filtering $a^{(j-1)}[n]$ with $h_0[n]$ followed by downsampling by two. This is the beauty of the Mallat's algorithm: higher level (smaller resolution) coefficients are obtained recursively from the previous level coefficients.

Similarly, by using the wavelet equation 2.60 and equation 2.66, we can find the recur-

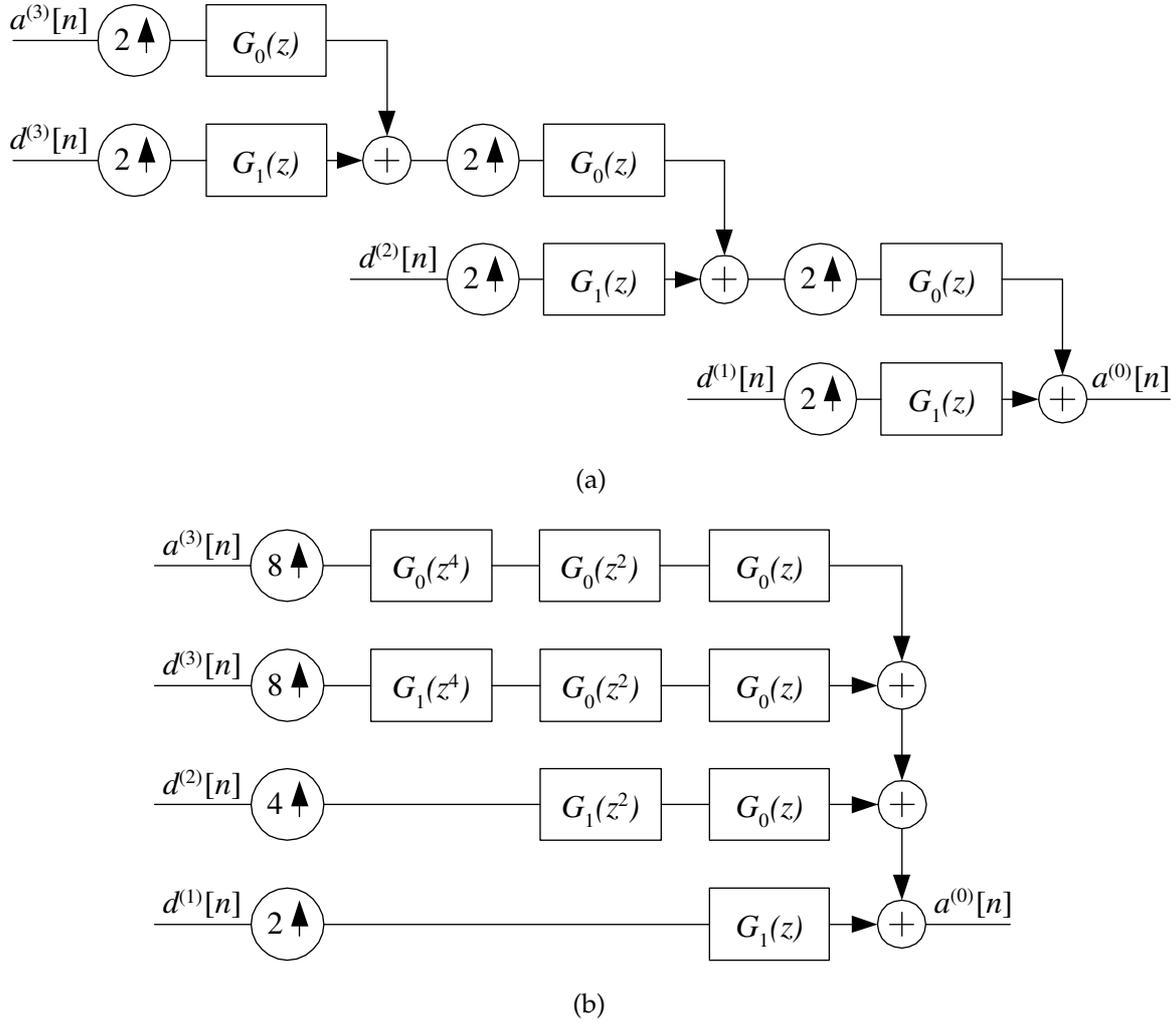


Figure 2.18: (a) Multiresolution reconstruction filter bank structure. (b) Equivalent structure obtained by using the *Second Noble Identity*.

sive formula for coefficients of the projection onto the detail space W_j :

$$d^{(j)}[n] = \sum_{k=-\infty}^{\infty} h_1[2n - k]a^{(j-1)}[k]. \quad (2.68)$$

Thus, the discrete detail signal $d^{(j)}[n]$ is obtained by filtering $d^{(j-1)}[n]$ with $h_1[n]$ and then downsampling by two. The equivalent structure that implements this algorithm is shown in figure 2.17(a). It is the basis for the fast discrete wavelet transform.

Where the property of orthogonality holds, reconstruction is possible and the connection among decomposition and reconstruction filters is: $g_0[n] = h_0[-n]$, $g_1[n] = h_1[-n]$. Including $h_1[n] = (-1)^n h_0[1 - n]$, these are the well known CQF (conjugate quadrature filters) relations described by Smith and Barnwell in [Smith 86] and [Mintzer 85].

Reconstruction is given with the following equation:

$$a^{(j-1)}[n] = \sum_{k=-\infty}^{\infty} (g_0[n-2k]a^{(j)}[k] + g_1[n-2k]d^{(j)}[k]). \quad (2.69)$$

The corresponding reconstruction filter bank structure is shown in figure 2.18(a). If $\{\varphi(t-n) \mid n \in \mathbf{Z}\}$ is the Riesz basis in V_0 , then the set of reconstruction functions is not the same as the set of decomposition functions. Beside $\varphi(t)$ and $\psi(t)$, there are also dual scaling function $\tilde{\varphi}(t)$ and dual wavelet $\tilde{\psi}(t)$ and filters $h_0[n]$, $h_1[n]$, $g_0[n]$ and $g_1[n]$ make a biorthogonal filter bank structure.

Derivation of Wavelet and Scale Functions Using Iterated Filter Bank

Let us analyze the reconstruction part of the (orthogonal) filter bank given in figure 2.18(a). Figure 2.18(b) shows the equivalent structure obtained by interchanging up-sampling and filtering by means of the *Second Noble Identity*. The equivalent low-pass filter obtained after N iterations of the basic filter bank structure is

$$G_0^{(N)}(z) = \prod_{i=0}^{N-1} G_0(z^{2^i}), \quad (2.70)$$

and the equivalent high-pass filter is

$$G_1^{(N)}(z) = G_1(z^{2^{N-1}}) \prod_{i=0}^{N-2} G_0(z^{2^i}). \quad (2.71)$$

Continuous-time functions are associated to these discrete filters' impulse responses in the following way:

$$\varphi^{(N)}(t) = 2^{\frac{N}{2}} g_0^{(N)}[n], \quad \frac{n}{2^N} \leq t < \frac{n+1}{2^N}, \quad (2.72)$$

$$\psi^{(N)}(t) = 2^{\frac{N}{2}} g_1^{(N)}[n], \quad \frac{n}{2^N} \leq t < \frac{n+1}{2^N}. \quad (2.73)$$

Since with every iteration the total impulse response becomes longer, normalization of the time scale was necessary to retain a compact support of the resulting functions. Also, multiplication by $2^{\frac{N}{2}}$ ensures preservation of the energy. With appropriate regularity of $g_0[n]$ and $g_1[n]$, where $N \rightarrow \infty$, these functions converge to the scale function and wavelet:

$$\varphi(t) = \lim_{N \rightarrow \infty} \varphi^{(N)}(t), \quad (2.74)$$

$$\psi(t) = \lim_{N \rightarrow \infty} \psi^{(N)}(t). \quad (2.75)$$

Necessary (but not sufficient) conditions for the convergence are:

$$\frac{1}{\sqrt{2}}G_0(1) = 1, \quad \text{and} \quad G_0(-1) = 0. \quad (2.76)$$

2.3 The Lifting Scheme

The lifting scheme [Sweldens 95b, Sweldens 95a, Sweldens 96] enables an easy and efficient construction of wavelets and wavelet transforms. A very important feature of the lifting scheme is that every filter bank based on lifting automatically satisfies perfect reconstruction properties. The lifting scheme starts with a set of well known filters, whereafter lifting steps are used in an attempt to improve (lift) the properties of a corresponding wavelet decomposition. There are two types of lifting steps: the *primal lifting step* (figure 2.21(a)) and the *dual lifting step* (figure 2.19(a)). A number of such lifting steps (dual and primal ones being interchanged) can be used in order to obtain desired properties of a wavelet transform.

2.3.1 Dual Lifting Step

Let us consider the filter bank from figure 2.8 which satisfies the perfect reconstruction property. Filters $H_0(z)$ and $H_1(z)$ are complementary since the determinant of the corresponding polyphase matrix $\mathbf{H}_p^{new}(z)$ equals 1. In this case we can obtain another filter $H_1^{new}(z)$ complementary to $H_0(z)$, by using the *dual lifting step* (figure 2.19(a)):

$$H_1^{new}(z) = H_1(z) - H_0(z)P(z^2) \quad (2.77)$$

With a good choice of P , properties of the new high-pass filter can be improved. It is said that the high-pass channel is being lifted (improved) with a help of the low-pass channel. Polyphase components of the new high-pass filter are:

$$H_{1e}^{new}(z) = H_{1e}(z) - H_{0e}(z)P(z), \quad (2.78a)$$

$$H_{1o}^{new}(z) = H_{1o}(z) - H_{0o}(z)P(z). \quad (2.78b)$$

New polyphase matrix obtained by using the dual lifting step is

$$\mathbf{H}_p^{new}(z) = \begin{bmatrix} H_{0e}(z) & H_{0o}(z) \\ H_{1e}(z) - H_{0e}(z)P(z) & H_{1o}(z) - H_{0o}(z)P(z) \end{bmatrix}, \quad (2.79)$$

and can be derived from the old polyphase matrix as

$$\mathbf{H}_p^{new}(z) = \begin{bmatrix} 1 & 0 \\ -P(z) & 1 \end{bmatrix} \mathbf{H}_p(z). \quad (2.80)$$

It is evident that this operation does not change the determinant of the polyphase matrix, which still equals 1.

To preserve the perfect reconstruction property of a new filter bank, an inverse dual

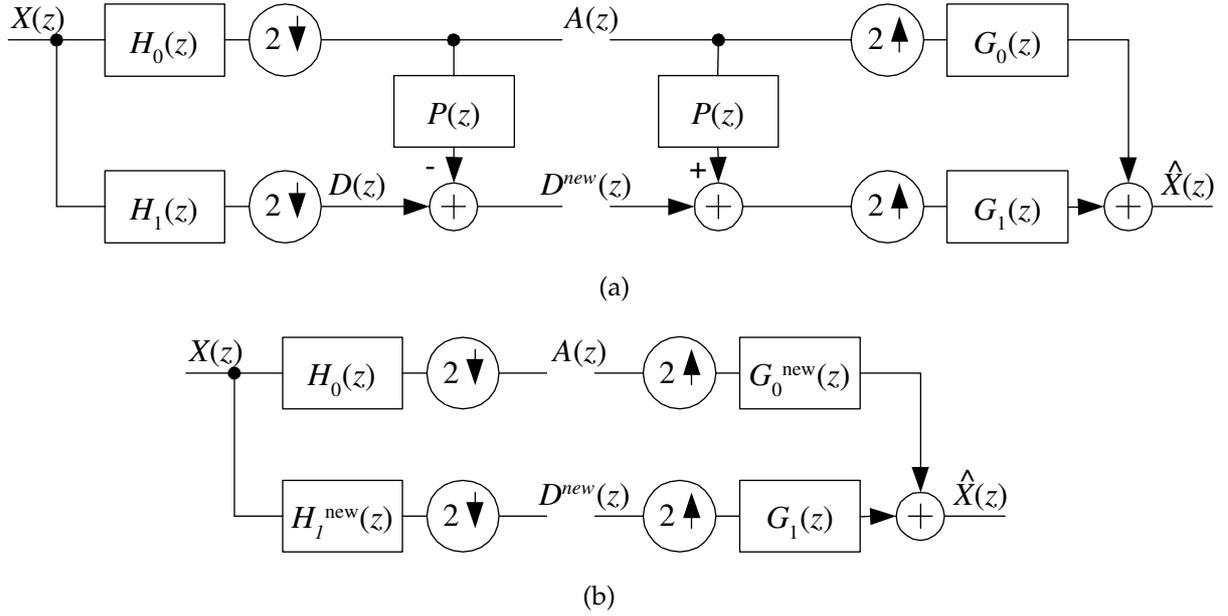


Figure 2.19: (a) Usage of a dual lifting step in improving properties of the high-pass channel. (b) The resulting equivalent structure.

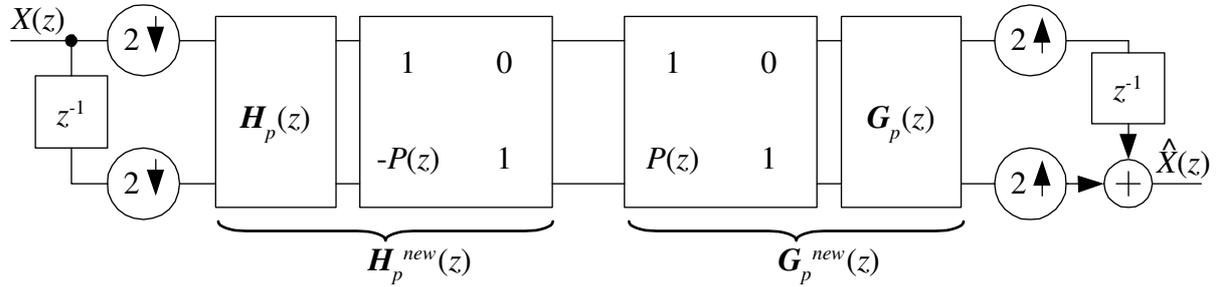


Figure 2.20: Polyphase representation of a dual lifting step.

lifting step will be performed on the synthesis side. The inverse is trivial to obtain since it is the same as the original dual lifting step with the only difference being the sign. The result is a new polyphase matrix

$$\mathbf{G}_p^{new}(z) = \mathbf{G}_p(z) \begin{bmatrix} 1 & 0 \\ P(z) & 1 \end{bmatrix}, \quad (2.81)$$

where

$$\mathbf{G}_p^{new}(z) = \begin{bmatrix} G_{0o}(z) + G_{1o}(z)P(z) & G_{1o}(z) \\ G_{0e}(z) + G_{1e}(z)P(z) & G_{1e}(z) \end{bmatrix}. \quad (2.82)$$

2.3.2 Primal Lifting Step

Similar procedure can be done in order to lift the properties of the low-pass channel. It is the so called *primal lifting step* or *update step*. The new low-pass analysis filter is the

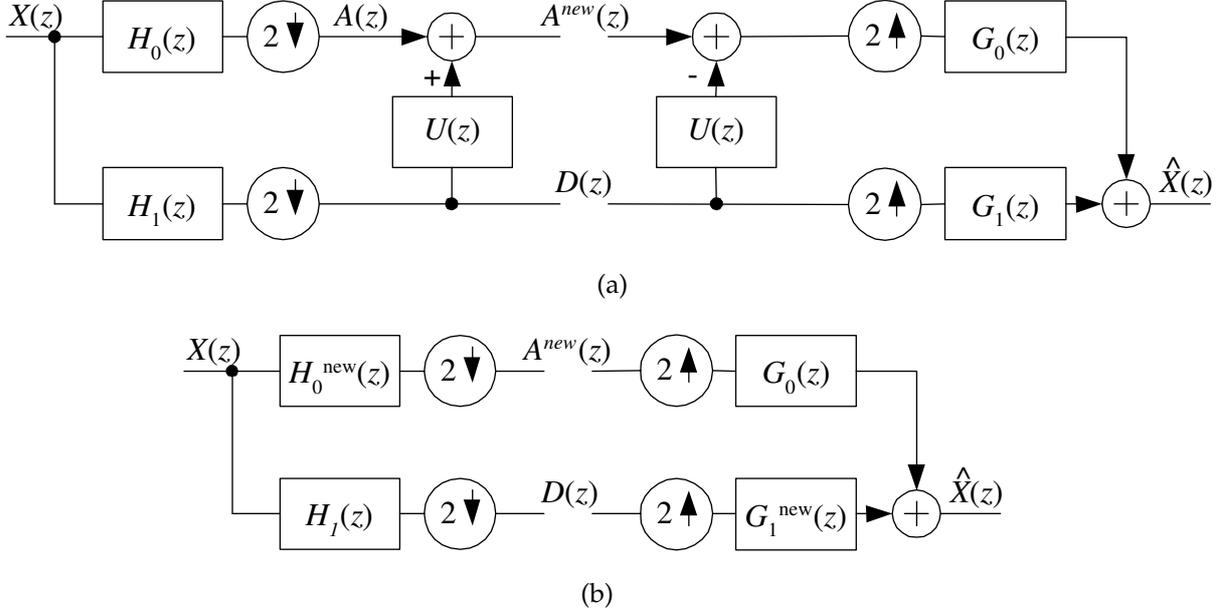


Figure 2.21: (a) Usage of a lifting step in improving the properties of the low-pass channel. (b) The resulting equivalent structure.

following:

$$H_0^{new}(z) = H_0(z) + H_1(z)U(z^2), \quad (2.83)$$

New polyphase matrices for the analysis and reconstruction sides can be expressed as

$$\mathbf{H}_p^{new}(z) = \begin{bmatrix} 1 & U(z) \\ 0 & 1 \end{bmatrix} \mathbf{H}_p(z), \quad (2.84a)$$

$$\mathbf{G}_p^{new}(z) = \mathbf{G}_p(z) \begin{bmatrix} 1 & -U(z) \\ 0 & 1 \end{bmatrix}, \quad (2.84b)$$

where

$$G_1^{new}(z) = G_1(z) - G_0(z)U(z^2). \quad (2.85)$$

2.3.3 Alternating Dual and Primal Lifting Step

A good choice of filters to begin lifting is a *Lazy wavelet*, i.e. a plain polyphase decomposition with polyphase matrices $\mathbf{H}_p(z) = \mathbf{I}$ and $\mathbf{G}_p(z) = \mathbf{I}$. A structure obtained in this way by using only two lifting steps is shown in figure 2.22. Appropriate polyphase

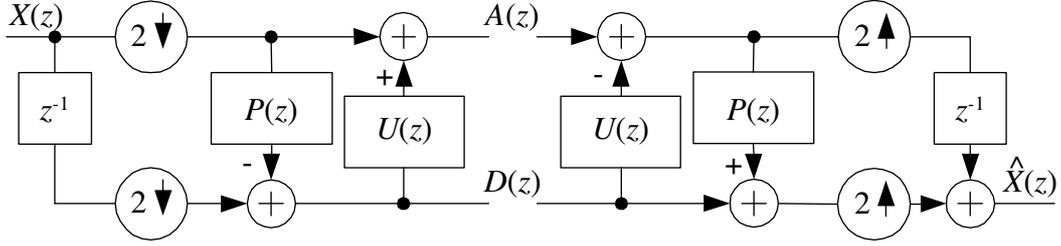


Figure 2.22: Filter bank based on lifting. Dual and primal lifting steps are used to obtain the filter bank structure with desired properties.

matrices are obtained as:

$$\mathbf{H}_p(z) = \begin{bmatrix} 1 & U(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -P(z) & 1 \end{bmatrix}, \quad (2.86a)$$

$$\mathbf{G}_p(z) = \begin{bmatrix} 1 & 0 \\ P(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -U(z) \\ 0 & 1 \end{bmatrix}. \quad (2.86b)$$

Predict and Update

The dual lifting step can be observed as a predictor, i.e. a *prediction step*. It predicts values of samples in the high-pass channel based on the samples from the low-pass channel. In this perspective, the resulting coefficients in the high-pass channel are actually a prediction error. If prediction is good, the respective coefficients will be small, with values that are more concentrated around zero. To retain some properties of the original signal (e.g. mean value), the primal lifting step is used. It can be called the *update step* since the samples from the low-pass channel are updated with the prediction-error coefficients from the high-pass channel.

The Generalization

Daubechies and Sweldens in [Daubechies 96] show that any polyphase matrix representing a wavelet transform with FIR filters can be factored into a finite number of lifting steps starting with the Lazy wavelet. The factorization can be done by using Euclidean algorithm. By using the proposed factorization, the analysis and synthesis polyphase matrices can be expressed as:

$$\mathbf{H}_p(z) = \begin{bmatrix} K_1 & 0 \\ 0 & K_2 \end{bmatrix} \left\{ \prod_{i=1}^m \begin{bmatrix} 1 & U_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -P_i(z) & 1 \end{bmatrix} \right\}, \quad (2.87a)$$

$$\mathbf{G}_p(z) = \begin{bmatrix} 1/K_1 & 0 \\ 0 & 1/K_2 \end{bmatrix} \left\{ \prod_{i=1}^m \begin{bmatrix} 1 & 0 \\ P_i(z) & 1 \end{bmatrix} \begin{bmatrix} 1 & -U_i(z) \\ 0 & 1 \end{bmatrix} \right\}. \quad (2.87b)$$

Chapter 3

Two-Dimensional Nonseparable Wavelet Filter Banks Using Lifting Scheme

In this chapter we present a 2-D filter bank that is the basis for the adaptive structure proposed in this thesis and explained in detail in chapter 5. It is a nonseparable filter bank based on a lifting scheme. Its implementation was presented by Kovačević and Sweldens in [Kovačević 00]. They have shown how to build discrete compactly supported biorthogonal wavelets and PR filter banks for any lattice, in any dimension, and with any number of primal and dual vanishing moments. Two lifting steps are used: the predict and the update, forming an interpolating filter bank structure. Prior to explaining the filter bank construction, an introduction to the basics of the two-dimensional sampling will be given.

3.1 Multidimensional Signals and Notation

Although this chapter and the rest of the thesis deal with two-dimensional signals, most of the equations given in the following sections hold not only for a 2-D case, but for a general multidimensional case as well. With a view to expressing equations that are independent of the number of dimensions of the analyzed signal, we use vectors instead of scalars for indexing purposes. These vectors are denoted with boldface lowercase letters (while boldface uppercase letters are used to denote matrices). Here are some vectors that will be frequently used in denoting the "time" index, the frequency

index and the z -domain index respectively:

$$\mathbf{n} = \begin{bmatrix} n_1 & n_2 & \dots & n_d \end{bmatrix}^T, \quad (3.1a)$$

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_1 & \omega_2 & \dots & \omega_d \end{bmatrix}^T, \quad (3.1b)$$

$$\mathbf{z} = \begin{bmatrix} z_1 & z_2 & \dots & z_d \end{bmatrix}^T. \quad (3.1c)$$

Now, the discrete d -dimensional signal $x[n_1, n_2, \dots, n_d]$ can be indexed as $x[\mathbf{n}]$. Also,

$$\mathbf{z}^{\mathbf{k}} = z_1^{k_1} z_2^{k_2} \dots z_d^{k_d}. \quad (3.2)$$

Raising a d -dimensional vector \mathbf{z} to a matrix \mathbf{M} with d rows and d columns yields

$$\mathbf{z}^{\mathbf{M}} = \begin{bmatrix} \mathbf{z}^{\mathbf{m}_1} & \mathbf{z}^{\mathbf{m}_2} & \dots & \mathbf{z}^{\mathbf{m}_d} \end{bmatrix}^T, \quad (3.3)$$

where \mathbf{m}_i denotes i -th column of the matrix \mathbf{M} . For the sake of an example, let us consider one two-dimensional case:

$$\mathbf{z} \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} & \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 3 \\ 4 \end{bmatrix} \end{bmatrix}^T = \begin{bmatrix} z_1^1 z_2^2 \\ z_1^3 z_2^4 \end{bmatrix}. \quad (3.4)$$

3.2 Nonseparable Versus Separable

A straightforward way of implementing two-dimensional wavelet decomposition is using a pyramidal algorithm described in [Mallat 89c]. It uses the well known one-dimensional wavelet decomposition applied separately along rows and columns of the analyzed image. The corresponding 2-D filters are separable, i.e. they can be expressed as products of 1-D filters. Definitely, this is the "easy" way to implement a 2-D wavelet transform. Still, because of its horizontal and vertical bias, this scheme is not the best choice in the image processing. Since the human visual system is most sensitive in horizontal and vertical directions, this bias will introduce perceptually greater distortions after applying nonlinear operations on wavelet coefficients (such as thresholding and quantization). It seems logical that in dealing with images a true 2-D processing should be used. True 2-D transforms are using nonseparable sampling and nonseparable filters. Although computationally more demanding, they bring some additional benefits. Nonseparable sampling can be chosen to treat all directions in the picture similarly. In this way, the wavelet decomposition will be better tuned to the human visual system. Also, there is more freedom in the design of nonseparable filters than in the design of separable filters.

3.3 Two-Dimensional Sampling

The first task of an m -channel critically sampled filter bank is to split the input image in m equally large distinct sets of pixels. This is a more sophisticated task than that in the 1-D case and there are many ways to subsample an image in a number of subsets. One example of splitting an image in two subsets is shown in figure 3.1. The original lattice is split in two sublattices in a checkerboard-like way. Gray circles represent samples on the first sublattice and white circles represent samples on the second sublattice. This type of lattice is called *quincunx* lattice.

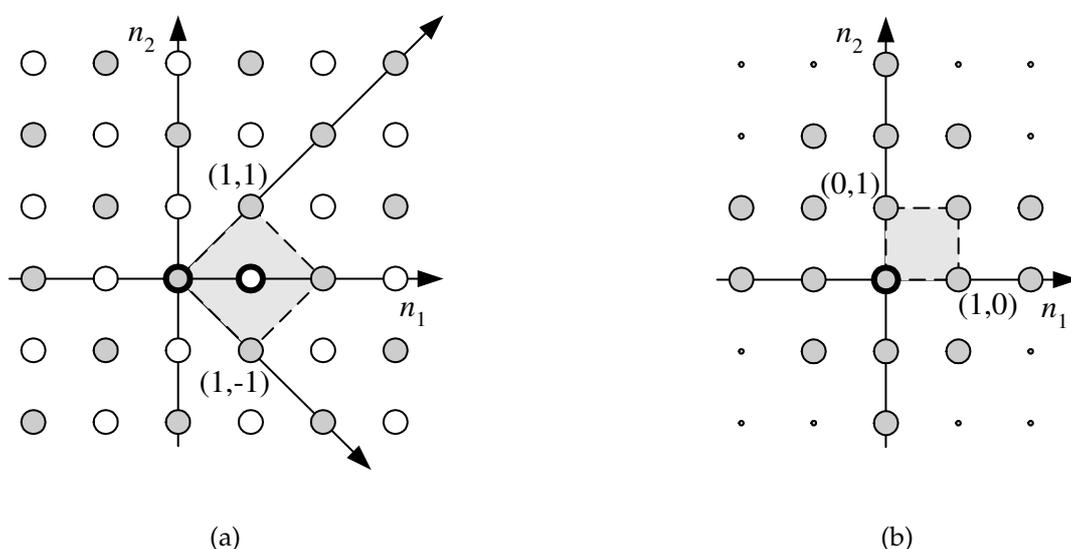


Figure 3.1: Quincunx subsampling. The fundamental parallelepiped is marked gray. Samples belonging to the *FPD* are marked with bold circles. (a) Samples of the original image. Gray circles represent samples belonging to the first coset. White circles represent the second coset. (b) First coset in the subsampled domain. Samples are renumbered in such a way that the overall effect is their 45° counterclockwise rotation.

3.3.1 Lattices and Dilation Matrix

There are other types of lattices [Dubois 85] that enable splitting of images in many different ways. Figure 3.2(a) shows one separable lattice, and figure 3.2(b) shows a hexagonal lattice. Every lattice is defined with a corresponding *dilation matrix* \mathbf{D} . In general, for a d -dimensional sampling this matrix will have d rows and d columns. Columns of matrix \mathbf{D} are vectors that form a basis of a given lattice. For a 2-D case the lattice is defined as a set of all vectors obtained as a linear combination of the two basis

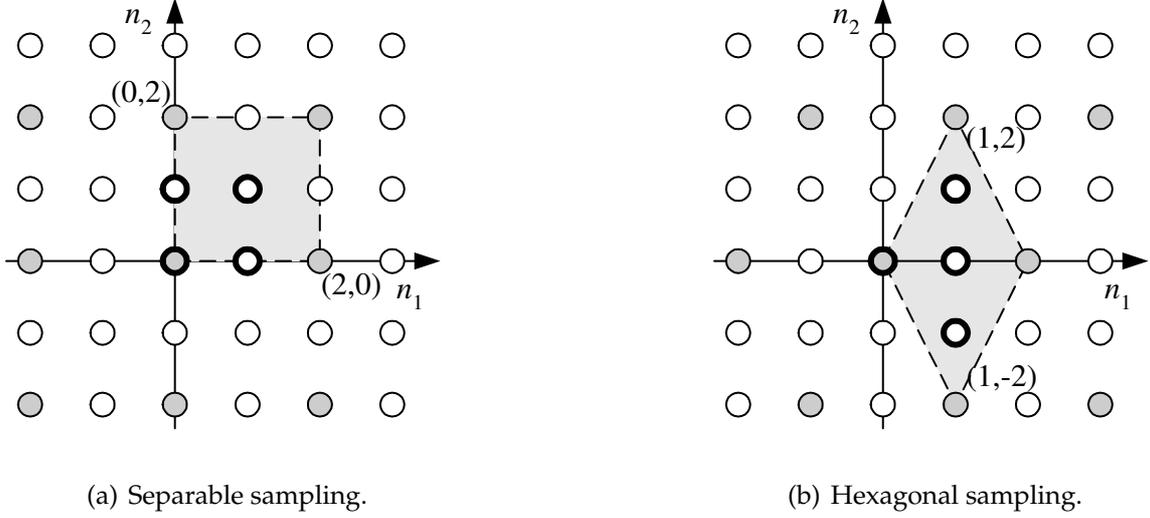


Figure 3.2: Two different sampling schemes: separable (a) and nonseparable (b). The fundamental parallelepiped is marked gray. Samples belonging to the *FPD* are marked with bold circles.

vectors:

$$LAT(\mathbf{D}) = \mathbf{D}\mathbf{n}, \quad \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \in \mathbb{Z}^2. \quad (3.5)$$

2-D dilation matrices corresponding to quincunx, separable and hexagonal lattices in figures 3.1(a), 3.2(a) and 3.2(b) are:

$$\mathbf{D}_q = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{D}_s = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \quad \mathbf{D}_h = \begin{bmatrix} 1 & 1 \\ -2 & 2 \end{bmatrix}. \quad (3.6)$$

It should be noted that the separable lattice is represented with a diagonal matrix. It is worth pointing out that there can be found a number of dilation matrices forming the same lattice. For example, all of the following dilation matrices

$$\mathbf{D}_q = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{D}_{q2} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}, \quad \mathbf{D}_{q3} = \begin{bmatrix} 1 & 2 \\ -1 & 0 \end{bmatrix}, \quad (3.7)$$

form the same quincunx lattice.

Determinant of the Dilation Matrix

A unit cell of a lattice defined by vectors of the matrix \mathbf{D} is called *fundamental parallelepiped* (*FPD*). The area of the fundamental parallelepiped can be calculated by using a dilation matrix as $|\det \mathbf{D}|$. Since only one point (point $(0, 0)$) of the lattice $LAT(\mathbf{D})$ belongs to $FPD(\mathbf{D})$, it follows that the sampling density, i.e. the number of samples per

unit area is $1/|\det \mathbf{D}|$. Similarly, the number of points $\mathbf{n} \in \mathcal{Z}^2$ belonging to the fundamental parallelepiped (*FPD*) can be expressed as reciprocal of the sampling density:

$$M = |\det \mathbf{D}|. \quad (3.8)$$

By shifting the lattice origin onto each member of $FPD(\mathbf{D})$, a corresponding coset of image samples is obtained. The total number of cosets equals M . Considering the critically sampled filter banks, every coset will represent one polyphase component and the total number of filter bank channels will equal M . It means that the quincunx sampling scheme will require a 2-channel filter bank since $|\det \mathbf{D}_q| = 2$, the separable sampling from figure 3.2(a) will require 4 channels since $|\det \mathbf{D}_s| = 4$, and the hexagonal sampling from figure 3.2(b) will also require 4 channels since $|\det \mathbf{D}_h| = 4$.

3.3.2 Downsampling

By using the vector notation, downsampling can be expressed as:

$$y_D[\mathbf{n}] = x[\mathbf{Dn}], \quad \text{for every } \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \in \mathcal{Z}^2. \quad (3.9)$$

This equation is very similar to 1-D decimation expressed in equation 2.1. For a d -dimensional case the relation 3.9 also holds with the only difference concerning the size of vector \mathbf{n} and matrix \mathbf{D} . The downsampling operation stated in the equation 3.9 can be perceived as a sequence of the following three steps:

Lattice points are defined with a choice of a dilation matrix \mathbf{D} (see equation 3.5).

Keep only image samples that lie on a given lattice, i.e. the first coset. All the other samples are being discarded. In the $FPD(\mathbf{D})$, it is only the sample at $(0, 0)$ that is retained. The decimation ratio is $M = |\det \mathbf{D}|$, i.e. one sample out of M is retained.

Renumber the retained samples to fill all the integer points $\mathbf{n} \in \mathcal{Z}^2$ of the output image. An example of renumbering of samples caused by quincunx downsampling is shown in figure 3.1(b).

In the frequency domain the following relationship stands:

$$Y_D(\boldsymbol{\omega}) = \frac{1}{|\det \mathbf{D}|} \sum_{\mathbf{k} \in \mathcal{N}(\mathbf{D}^T)} X(\mathbf{D}^{-T}(\boldsymbol{\omega} - 2\pi\mathbf{k})), \quad (3.10)$$

where $\mathcal{N}(\mathbf{D}^T)$ is a set of integer vectors in the $FPD(\mathbf{D}^T)$, $\mathbf{D}^{-T} = (\mathbf{D}^{-1})^T$ and $\boldsymbol{\omega} = [\omega_1 \ \omega_2]^T$. The effect is similar to the effect of one-dimensional decimation. A spectrum of the decimated signal is obtained by adding $M = |\det \mathbf{D}|$ copies of $X(\mathbf{D}^{-T}\boldsymbol{\omega})$, i.e. a stretched input signal's spectrum. Each stretched copy is shifted for the corresponding $2\pi\mathbf{k}$.

For a quincunx decimation with dilation matrix as defined in 3.6 there will be two copies of the input signal's spectrum. The fundamental parallelepiped $FPD(\mathbf{D}^T)$ contains two integer vectors, namely $(0, 0)$ and $(0, 1)$. Therefore, equation 3.10 can be written as

$$Y_D(\boldsymbol{\omega}) = \frac{1}{2} \left[X \left(\begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} \right) + X \left(\begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 - 2\pi \end{bmatrix} \right) \right]. \quad (3.11)$$

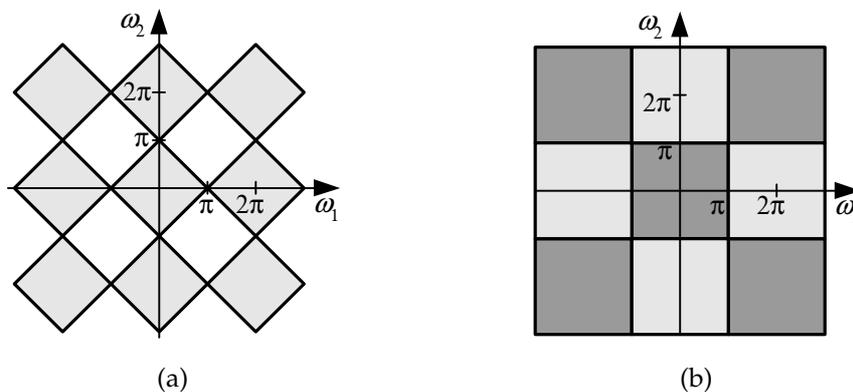


Figure 3.3: The effect of a quincunx decimation in the frequency domain. (a) Support of $X(\boldsymbol{\omega})$. (b) Support of $X(\mathbf{D}^{-T}\boldsymbol{\omega})$ is marked dark gray. Support of its shifted version is represented with the light gray areas. The original signal's spectrum was bandlimited to the Voronoi cell of the scaled reciprocal lattice $LAT(2\pi\mathbf{D}^{-T})$ (see figure 3.4(b)) so that no aliasing occurred.

Reciprocal Lattice and the Voronoi Cell

In equation 3.10, the \mathbf{D}^{-T} is responsible for stretching the original signal's spectrum. Lattice $LAT(\mathbf{D}^{-T})$ is called the *reciprocal lattice* to the original lattice $LAT(\mathbf{D})$. Furthermore, a *scaled reciprocal lattice* is $LAT(2\pi\mathbf{D}^{-T})$. Another type of unit cell is the *Voronoi cell*. It consists of points that are closer to the origin than to any other point of a given lattice. If the spectrum of a signal (image) is bandlimited to the Voronoi cell of a frequency domain (scaled reciprocal) lattice, then no overlapping after decimation will occur (see figure 3.3).

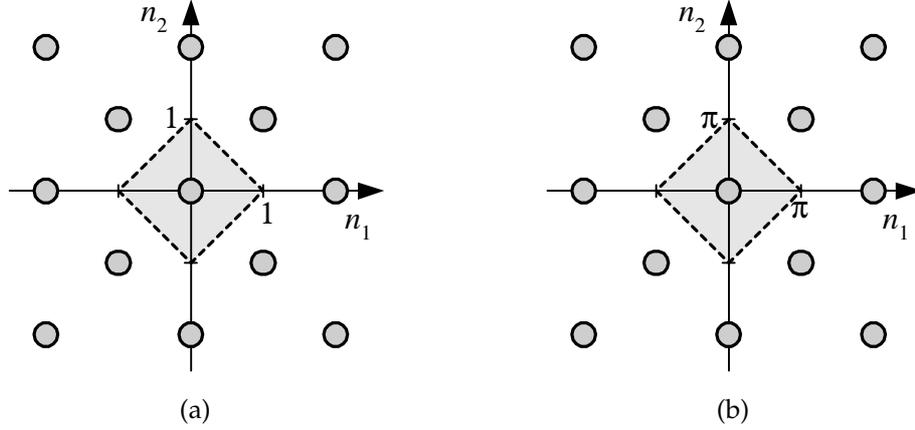


Figure 3.4: (a) Voronoi cell of the quincunx lattice $LAT(\mathbf{D}_q)$. (b) Voronoi cell of the quincunx scaled reciprocal lattice $LAT(2\pi\mathbf{D}_q^{-T})$.

3.3.3 Upsampling

The upsampling operation can be expressed by the following equation:

$$y_E[\mathbf{n}] = \begin{cases} x[\mathbf{D}^{-1}\mathbf{n}] & \text{if } \mathbf{n} \in LAT(\mathbf{D}), \\ 0 & \text{otherwise.} \end{cases} \quad (3.12)$$

Equation 3.12 states that samples from the input image are placed on the points of the output images that belong to the lattice $LAT(\mathbf{D})$. All the other points of the output image are filled with zeros. In the frequency domain the upsampling operation gives:

$$Y_E(\boldsymbol{\omega}) = X(\mathbf{D}^T\boldsymbol{\omega}), \quad (3.13)$$

or expressed in the z-domain:

$$Y_E(\mathbf{z}) = X(\mathbf{z}^{\mathbf{D}}). \quad (3.14)$$

Given a quincunx dilation matrix, equation 3.14 becomes

$$Y_E(z_1, z_2) = X\left(\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\right) = X\left(\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}\right) = X(z_1 z_2^{-1}, z_1 z_2) \quad (3.15)$$

The effect of quincunx upsampling is shown in figure 3.5. The obtained imaging effect is similar as in the 1-D case discussed in section 2.1.2. The original image spectrum has been shrunk so that $|\det \mathbf{D}| = 2$ copies of the basic spectrum fit into the frequency square from $(0, 0)$ to $(2\pi, 2\pi)$. In that area, there are $|\det \mathbf{D}| - 1$ images. In the quincunx case, there is only one image. Figure 3.6 show the effect of iterative upsampling of the diamond shaped low-pass filter using a quincunx sampling scheme. After every iterated upsampling step the basic spectrum rotates by 45° and shrinks. It should be noted that for every second step, the same shape is obtained.

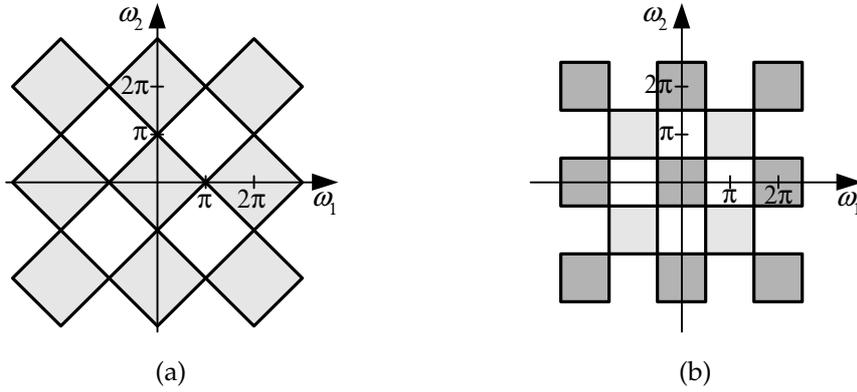


Figure 3.5: The frequency domain imaging effect caused by a quincunx upsampling. (a) Support of $X(\omega)$. (b) Support of $X(\mathbf{D}^T \omega)$. Lighter gray areas represent images.

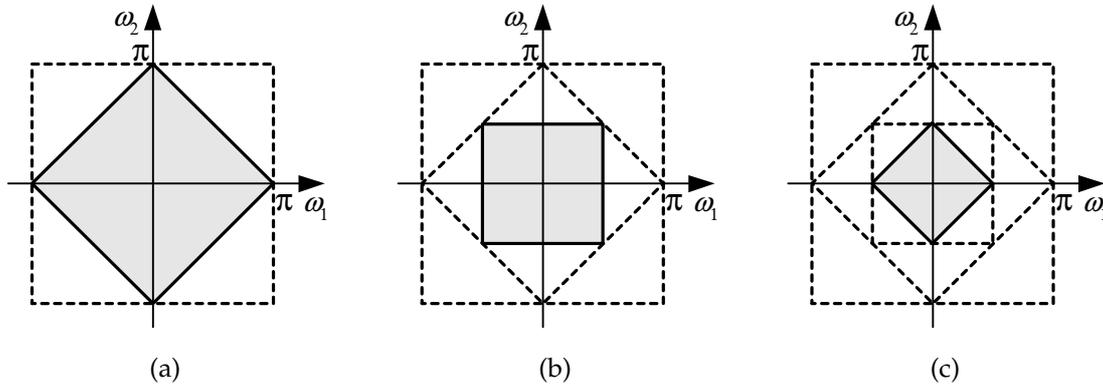


Figure 3.6: Frequency support of the iterated quincunx low-pass filter. (a) Support of the original quincunx low-pass filter. (b) Support of the filter obtained after first quincunx upsampling of the basic filter. (c) Support of the filter obtained after second quincunx upsampling of the basic filter.

3.3.4 Simplest Nonseparable Choice: Quincunx Sampling

Filter banks that are presented in the following sections are based on a quincunx sampling. Quincunx sampling is chosen as the simplest nonseparable sampling scheme. Since the determinant of the quincunx dilation matrix equals 2, the corresponding critically sampled filter bank will have two channels. Considering the frequency domain, the quincunx sampling shows better results than the separable sampling. The reason lies in the fact that the human visual system is more sensitive to higher frequencies in horizontal and vertical directions than to the frequencies in diagonal directions. The ideal quincunx low-pass filter based on a reciprocal Voronoi cell (see figure 3.4(b)) has a diagonal cutoff. It means that the horizontal and vertical frequencies will be preserved, while the diagonal ones will be limited bounded to one half of their original range, and

it fits the human eye pretty good.

For a quincunx downsampling relation 3.9 becomes

$$y_D[n_1, n_2] = x[n_2 + n_1, n_2 - n_1], \quad (3.16)$$

while the upsampling relation 3.12 can be expressed as

$$y_E[n_1, n_2] = \begin{cases} x[\frac{1}{2}(n_1 - n_2), \frac{1}{2}(n_1 + n_2)] & \text{if } \mathbf{n} \in LAT(\mathbf{D}), \\ 0 & \text{otherwise.} \end{cases} \quad (3.17)$$

The effect of a quincunx subsampling is shown in figure 3.1. The original image is split in two cosets as shown in figure 3.1(a). The first phase (shown in figure 3.1(b)) is obtained by shifting the first coset by 45° in the counterclockwise direction. This procedure can also be perceived as mapping of samples in the coordinate system defined by unit vectors (see rotated axes marked in figure 3.1(a)). A quincunx polyphase decomposition of one simple image is shown in figure 3.7. Notice the effect of the counterclockwise rotation¹ by 45° . It is interesting to note that the one pixel wide diagonal line is completely transferred in the first phase while it completely disappeared in the second phase.

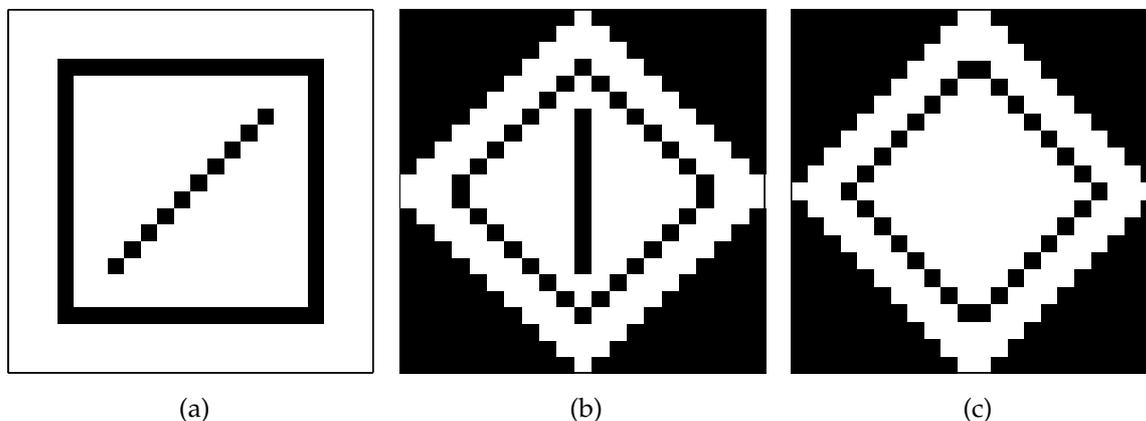


Figure 3.7: Quincunx polyphase decomposition: the original image (a) is split in two phases (b) and (c).

¹Practical note. In further sections results of wavelet image analysis will be shown by using the MATLAB program. Since MATLAB loads the image in a ij coordinate system, rather than a xy , the polyphase components will seem to be rotated in a clockwise direction.

3.4 Quincunx Interpolating Filter Banks

In this section we present a method to design filter banks based on a lifting scheme as proposed by Kovačević and Sweldens in [Kovačević 00]. This is a very powerful method allowing one to custom design wavelet filter banks and wavelets in any dimension, for any kind of sampling schemes and any number of dual and primal vanishing moments. Being based on lifting, it brings along additional benefits inherent in the lifting scheme (see section 2.3). For purposes of the thesis, just one special family of filter banks will be considered: the two-dimensional two-channel filter banks based on a quincunx sampling scheme.

3.4.1 Interpolating Filters

For a filter to be interpolating with a given sampling scheme its impulse response must be zero valued for all the samples of the 0th coset except in the origin. In the case of the sampling scheme that yields only two phases, the interpolating filter is expressed as:

$$H(\mathbf{z}) = 1 + \mathbf{z}^t H_o(\mathbf{z}), \quad (3.18)$$

where t is the shift between the two cosets.

3.4.2 Neville Filters

Filters built by Kovačević and Sweldens in [Kovačević 00] are based on a polynomial interpolation. They call them the Neville filters. To explain the purpose of a Neville filter let us consider a polynomial of order $N - 1$ sampled on some d -dimensional lattice. It will be denoted as $\pi(\mathcal{Z}^d)$. By applying a Neville filter P of order N or higher on the sampled polynomial, the result obtained will be samples of the very polynomial, yet sampled on a given lattice shifted by τ :

$$P\pi(\mathcal{Z}^d) = \pi(\mathcal{Z}^d + \tau) \quad (3.19)$$

where $\tau \in \mathcal{R}^d$.

3.4.3 Construction of Quincunx Neville Filters

Building of Neville filters in one dimension is simple and straightforward. It relies on a very well known Lagrangian interpolation. The efficient algorithm for calculating the interpolating value is called the Neville algorithm (thence the name Neville filters). In two or more dimensions the interpolation problem becomes more complicated as

not only the function of the number of given points the interpolation is based on, but also as the function of their mutual positions. The solution to this problem has been dealt with in the study by de Boor and Ron [de Boor 90, de Boor 92]. These authors have provided an algorithm that, based on a given set of points, gives a polynomial space for which the interpolation to arbitrary data given at these points is possible and unique.

Figure 3.8 shows a quincunx interpolation scheme. Samples of the original image are split in two cosets by using quincunx sampling. The first coset is represented with gray circles and the second coset is represented with white circles. As shown in figure 3.8(a), one sample from the second coset is being predicted by using interpolation to its surrounding samples from the first coset. Figure 3.8(b) represents the same interpolation now in the sampled domain. We interpolate at a number of points that are grouped in rings. For higher interpolation orders more rings are being included. Rings are chosen in the way shown in figure 3.8(b).

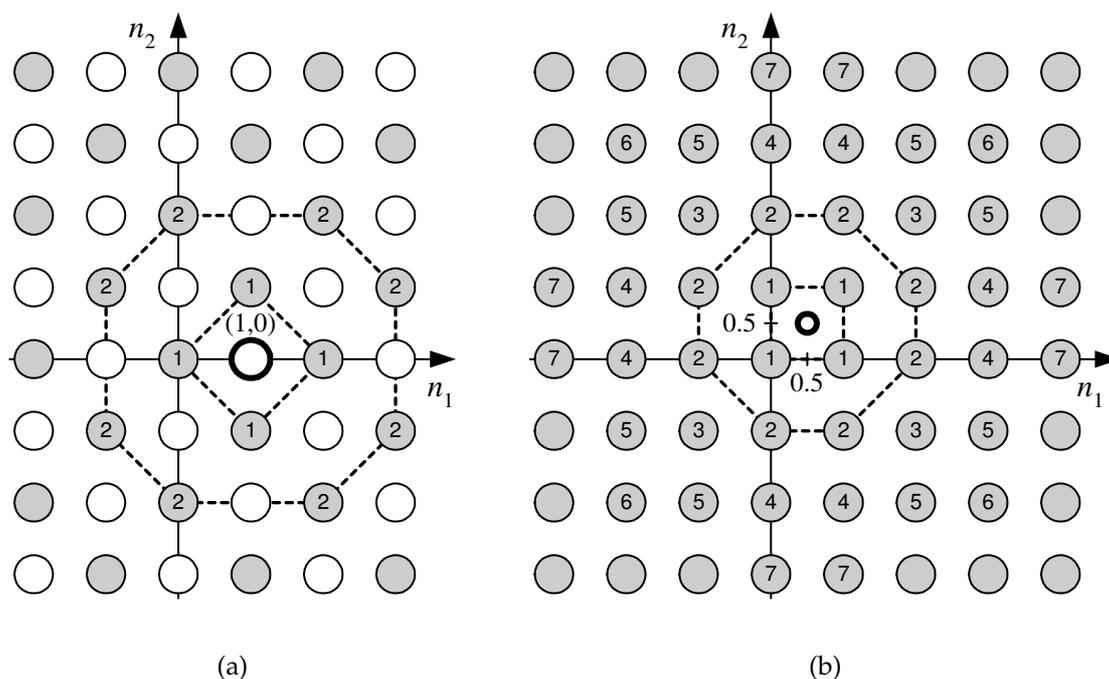


Figure 3.8: Quincunx interpolation. (a) The original rectangular lattice is split in two quincunx lattices. The bold white circle represents the sample from the second coset whose value is being predicted based on a number of neighboring samples from the first coset. Samples are marked with the numbers of rings they belong to. Samples of the first two rings are connected with a dashed line. (b) Interpolating rings in the sampled domain. The small white circle represents the interpolation point.

Order	ring 1	ring 2	ring 3	ring 4	ring 5	ring 6	ring 7	
2	1							$\times 2^{-2}$
4	10	-1						$\times 2^{-5}$
6	174	-27	2	3				$\times 2^{-9}$
8	23300	-4470	625	850	-78	9	-80	$\times 2^{-16}$

Table 3.1: Filter coefficients. Filter of a given order consists of one or more rings. Each ring includes a number of filter coefficients. The ring numbering scheme is as shown in figure 3.8.

In the sampled domain, the prediction of a sample from the second coset is obtained by filtering the first coset with a Neville filter of shift

$$\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t} = \begin{bmatrix} 0.5 & -0.5 \\ 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \quad (3.20)$$

where \mathbf{t} is a shift of the second coset with respect to the first coset. Coefficients of a Neville filter obtained by using the de Boor-Ron algorithm are given in table 3.1. For a higher interpolation order more interpolation rings are included and the corresponding Neville filter has more taps.

3.4.4 Predict and Update

In building a filter bank with desired properties, only two lifting steps will suffice: the predict and the update. The purpose of the predict step is to predict values of the second coset (even samples) based on the values from the first coset (odd samples). Wavelet (highpass) coefficients will be calculated as a prediction error, i.e. even samples minus the prediction of even samples (see figure 3.9). Neville filters described

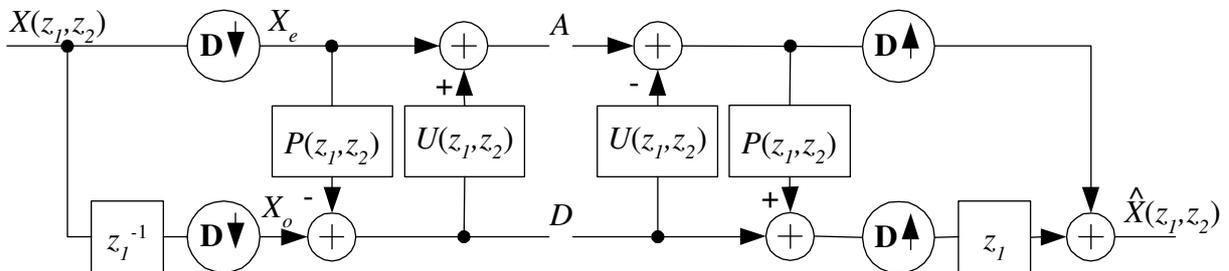


Figure 3.9: Quincunx interpolating filter bank. Downsampling and upsampling operators are defined by using a quincunx dilation matrix, $\mathbf{D} = \mathbf{D}_q$ from equation 3.6.

earlier will be used as the predict filters (section 3.4.5). In this way, if the input is a

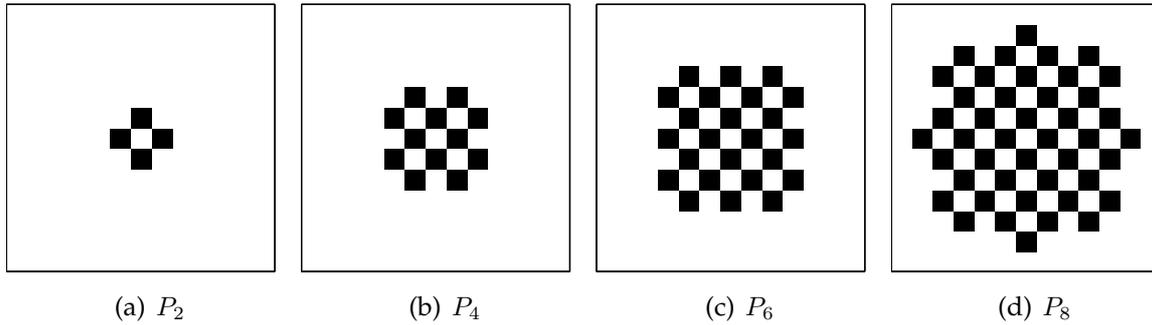


Figure 3.10: Support of the impulse response of the predict filters P of different orders in the domain of the original image. Black squares represent pixels that have nonzero values. Support of P_4 shown in (b) is defined by the first two rings as shown in figure 3.8(a).

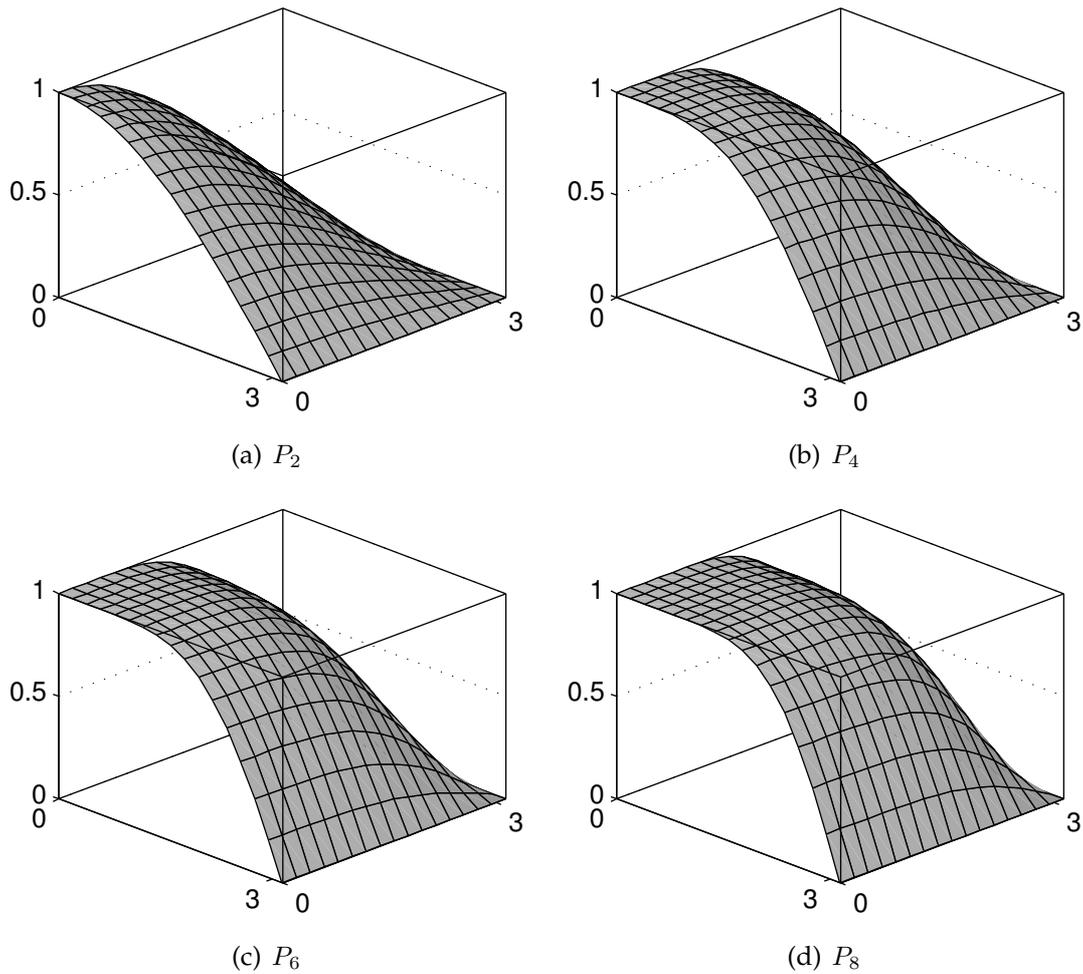


Figure 3.11: Magnitude frequency responses for the predict filters of different orders.

polynomial of an order smaller than the order of a predict filter P , prediction will be perfect and wavelet coefficients will all be set to zero. After a predict step, the update step is used to update even samples based on the previously calculated detail (wavelet) coefficients. The update filter U is designed to make the average value of the output of the low-pass channel the same as the average values of the input signal (image).

The analysis polyphase matrix of the filter bank shown in figure 3.9 is:

$$\mathbf{H}_p(\mathbf{z}) = \begin{bmatrix} H_{0e}(\mathbf{z}) & H_{0o}(\mathbf{z}) \\ H_{1e}(\mathbf{z}) & H_{1o}(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} 1 & U(\mathbf{z}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -P(\mathbf{z}) & 1 \end{bmatrix} = \begin{bmatrix} 1 - U(\mathbf{z})P(\mathbf{z}) & U(\mathbf{z}) \\ -P(\mathbf{z}) & 1 \end{bmatrix}, \quad (3.21)$$

which can be written as

$$H_0(\mathbf{z}) = 1 - P(\mathbf{z}^D)U(\mathbf{z}^D) + \mathbf{z}^{-t}U(\mathbf{z}^D), \quad (3.22a)$$

$$H_1(\mathbf{z}) = -P(\mathbf{z}^D) + \mathbf{z}^{-t}. \quad (3.22b)$$

The synthesis polyphase matrix follows directly from analysis polyphase matrix:

$$\mathbf{G}_p(\mathbf{z}) = \begin{bmatrix} G_{0e}(\mathbf{z}) & G_{0o}(\mathbf{z}) \\ G_{1e}(\mathbf{z}) & G_{1o}(\mathbf{z}) \end{bmatrix} = \mathbf{H}_p^{*-1}(\mathbf{z}) = \begin{bmatrix} 1 & P^*(\mathbf{z}) \\ -U^*(\mathbf{z}) & 1 - U^*(\mathbf{z})P^*(\mathbf{z}) \end{bmatrix}. \quad (3.23)$$

Therefore,

$$G_0(\mathbf{z}) = 1 + \mathbf{z}^t P^*(\mathbf{z}^D), \quad (3.24a)$$

$$G_1(\mathbf{z}) = -U^*(\mathbf{z}^D) + \mathbf{z}^t(1 - P^*(\mathbf{z}^D)U^*(\mathbf{z}^D)). \quad (3.24b)$$

3.4.5 Vanishing Moments

A filter bank has \tilde{N} dual vanishing moments (DM) if the analysis high-pass channel annihilates the polynomial sequences of order lower than \tilde{N} . Also, a filter bank is said to have N vanishing moments (PM) if the synthesis high-pass channel annihilates polynomial sequences of the order lower than N . The following equations stand:

$$(\downarrow \mathbf{D})H_1\pi = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}, \quad (3.25a)$$

$$G_1(\uparrow \mathbf{D})\pi = 0 \quad \text{for } \pi \in \Pi_N, \quad (3.25b)$$

where $\Pi_{\tilde{N}}$ and Π_N denote the spaces of all the polynomial sequences of the total degree less than \tilde{N} and N respectively. To permit the perfect reconstruction, the polynomial sequences annihilated in the high-pass channel have to be preserved in the low-pass channel. In terms of wavelet and scaling functions, if the dual (analysis) wavelet has \tilde{N} vanishing moments, then the primal (synthesis) scaling function has to be able to

reproduce polynomials up to the degree \tilde{N} . Also, if the primal wavelet has N vanishing moments, then the dual scaling function has to reproduce polynomials up to the degree N .

Firstly we will consider the condition for obtaining \tilde{N} dual vanishing moments using the lifting scheme. Downsampled output of the analysis high pass filter can be expressed in terms of the polyphase components, each polyphase component affecting one sublattice. Therefore, the DM condition becomes:

$$(\downarrow \mathbf{D})H_1\pi(\mathcal{Z}^2) = H_{1e}\pi(\mathbf{D}\mathcal{Z}^2) + H_{1o}\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \quad (3.26)$$

Combining equations 3.26 and 3.21 we get:

$$-P\pi(\mathbf{D}\mathcal{Z}^2) + \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}, \quad (3.27)$$

which yields

$$P\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \quad (3.28)$$

Equation 3.28 can be expressed in the downsampled domain as

$$P\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \quad (3.29)$$

Equation 3.29 reveals the DM condition. In order to have \tilde{N} dual vanishing moments, P has to be a Neville filter of the order \tilde{N} and shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$! In that case the predict filter will be able to give values of the polynomial sampled on the odd lattice (odd phase) based on the values of that polynomial sampled on the even lattice (even phase). Prediction will be perfect, and the resulting high-pass output, which is calculated as a difference of the odd phase and a prediction of the odd phase, will be equal to zero.

With the P filter satisfying the DM condition, U filter will be defined to satisfy the PM condition:

$$\begin{aligned} G_1(\uparrow \mathbf{D})\pi(\mathcal{Z}^2) &= G_{1e}\pi(\mathbf{D}\mathcal{Z}^2) + G_{1o}\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \\ &= -U^*\pi(\mathbf{D}\mathcal{Z}^2) + (1 - U^*P^*)\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \\ &= -U^*\pi(\mathbf{D}\mathcal{Z}^2) + \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) - U^*P^*\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_N. \end{aligned} \quad (3.30)$$

For $N \leq \tilde{N}$, equation 3.29 stands and 3.30 becomes

$$2U^*\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \quad \text{for } \pi \in \Pi_N. \quad (3.31)$$

Equation 3.31 states that in order to obtain N primal vanishing moments, with P being a Neville filter of the order \tilde{N} and shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$, $2U^*$ has to be a Neville filter of the

order N and shift τ . Consequently, $2U$ has to be a Neville filter of the order N and shift $-\tau$. The simplest choice is to make

$$U = U_N = \frac{1}{2}P_N^*, \quad (3.32)$$

where P_N is a Neville filter of the order N and shift τ while U_N per se is not a Neville filter.

For example, the filter bank with 4 dual vanishing moments and 2 primal vanishing moments will be constructed using the prediction filter P_4 and the update filter $U = 1/2P_2^*$ (see table 3.1):

$$P_4(z_1, z_2) = \frac{10}{32}(1 + z_1^{-1} + z_2^{-1} + z_1^{-1}z_2^{-1}) - \frac{1}{32}(z_1^{-2} + z_2^{-2} + z_1^{-2}z_2^{-1} + z_1^{-1}z_2^{-2} + z_1 + z_2 + z_1z_2^{-1} + z_1^{-1}z_2), \quad (3.33)$$

$$U_2(z_1, z_2) = \frac{1}{2}P_2^*(z_1, z_2) = \frac{1}{8}(1 + z_1 + z_2 + z_1z_2). \quad (3.34)$$

Magnitude frequency responses of the resulting analysis and synthesis filters are shown in figure 3.12. Since the analysis wavelet function has 4 vanishing moments and the synthesis wavelet function has 2 vanishing moments, the magnitude frequency response for the analysis high-pass filter is much more flat around the DC than for the synthesis low-pass filter. Analysis high-pass filter has 4th order zero and the synthesis high-pass filter has 2nd order zero at the DC frequency. The synthesis low-pass filter is interpolating (see equation 3.24), enabling the reconstruction of the polynomial sequences up to the 4th order. The duality between the primal (synthesis) low-pass filter and the dual (analysis) high-pass filter is obvious when comparing their magnitude frequency responses. Considering the frequency domain from 0 to π , the magnitude frequency response of the G_0 filter is a mirrored magnitude frequency response of the H_1 obtained by reversing the frequency scale (compare figures 3.12(b) and 3.12(c)). The same duality holds for the analysis low-pass filter H_0 and the synthesis high-pass filter G_1 .

3.4.6 Interchange of Dual and Primal Functions

In most image processing applications the constraint $N \leq \tilde{N}$ (number of primal vanishing moments must be less or equal to the number of dual vanishing moments) is not a problem. Actually, it is convenient to have smooth reconstruction functions (see figure 3.15) resulting from an interpolating synthesis low-pass filter. Also, higher number of dual vanishing moments gives better polynomial cancellation.

Yet, if one needs $N \geq \tilde{N}$, it can be obtained by a simple alteration of signs in the filter

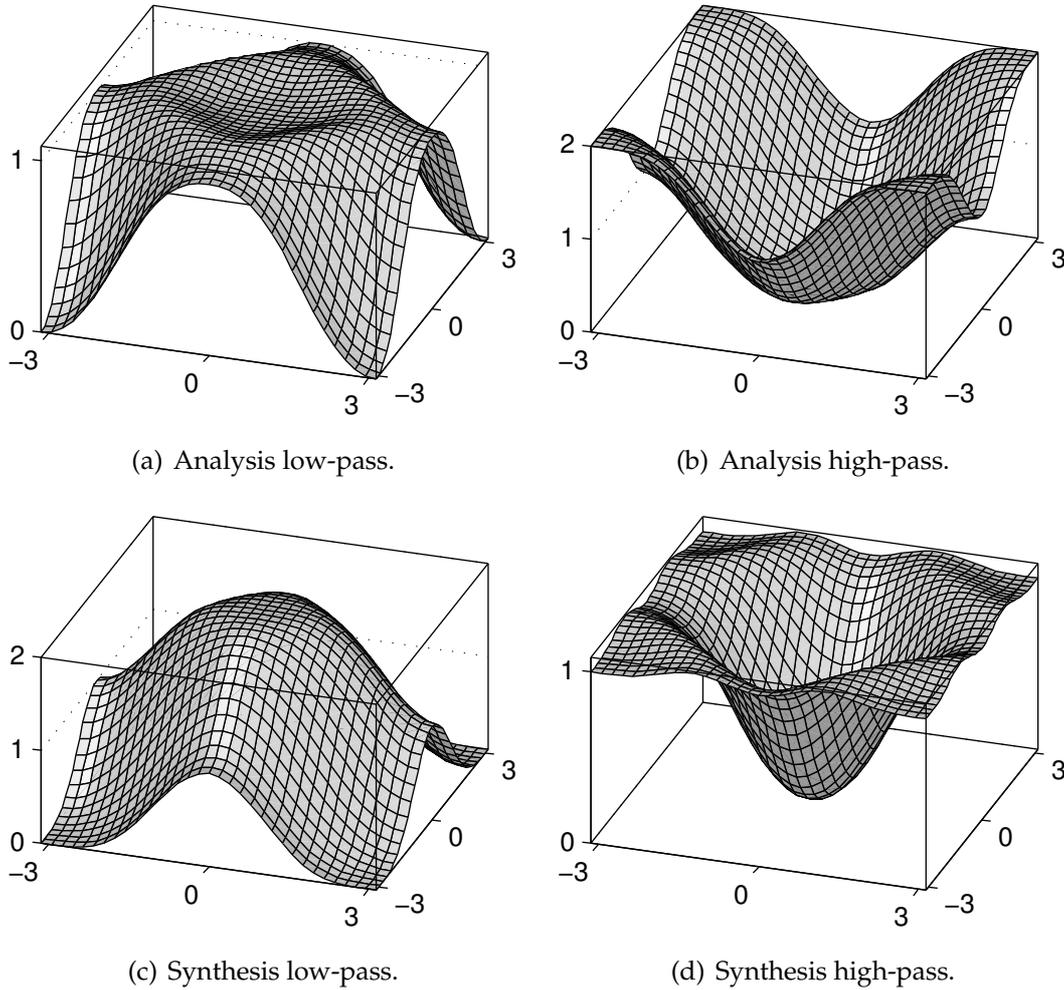


Figure 3.12: Magnitude frequency responses of the filters obtained by using prediction filter P_4 and update filter U_2 .

bank, as shown in figure 3.13. In such a filter bank roles of dual and primal functions are interchanged. The following equations stand:

$$H_0(\mathbf{z}) = P(\mathbf{z}^D) + \mathbf{z}^{-t}, \quad (3.35a)$$

$$H_1(\mathbf{z}) = 1 - P(\mathbf{z}^D)U(\mathbf{z}^D) - \mathbf{z}^{-t}U(\mathbf{z}^D), \quad (3.35b)$$

$$G_0(\mathbf{z}) = U^*(\mathbf{z}^D) + \mathbf{z}^t(1 - P^*(\mathbf{z}^D)U^*(\mathbf{z}^D)), \quad (3.36a)$$

$$G_1(\mathbf{z}) = 1 - \mathbf{z}^tP^*(\mathbf{z}^D). \quad (3.36b)$$

Now, the analysis low-pass filter H_0 is interpolating. Also, the resulting analysis wavelet and scale functions have higher regularity than the synthesis wavelet and scale functions.

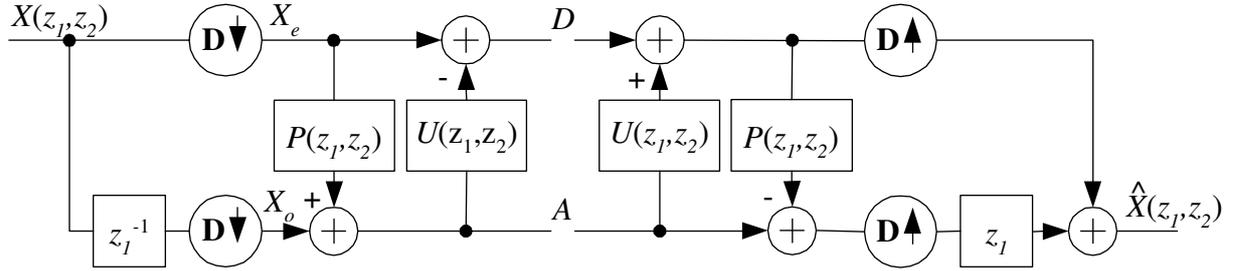


Figure 3.13: Reversed filter bank structure obtained by changing signs in the quincunx interpolating filter bank shown in 3.9. The first channel becomes high-pass, and the second one becomes low-pass.

3.4.7 Iterated Filter Banks

By iterating a synthesis filter bank on a low-pass channel the following low-pass and high-pass filters are obtained:

$$G_0^{(N)}(\mathbf{z}) = \prod_{i=0}^{N-1} G_0(\mathbf{z}^{\mathbf{D}^i}), \quad (3.37)$$

$$G_1^{(N)}(\mathbf{z}) = G_1(\mathbf{z}^{\mathbf{D}^{N-1}}) \prod_{i=0}^{N-2} G_0(\mathbf{z}^{\mathbf{D}^i}). \quad (3.38)$$

The similar relations hold for the analysis filter bank:

$$H_0^{(N)}(\mathbf{z}) = \prod_{i=0}^{N-1} H_0(\mathbf{z}^{\mathbf{D}^i}), \quad (3.39)$$

$$H_1^{(N)}(\mathbf{z}) = H_1(\mathbf{z}^{\mathbf{D}^{N-1}}) \prod_{i=0}^{N-2} H_0(\mathbf{z}^{\mathbf{D}^i}). \quad (3.40)$$

The results on iterating analysis filters from figure 3.12 are shown in figure 3.14. Compare the diamond-shaped frequency decomposition of the iterated quincunx scheme with figure 3.6.

Resulting impulse responses of the analysis and synthesis filters after seven iterations, corresponding to the limit scale and wavelet functions, are shown in figure 3.15. As expected, the synthesis functions are smooth, as a consequence of the synthesis low-pass filter being interpolating. Analysis functions are far less regular.

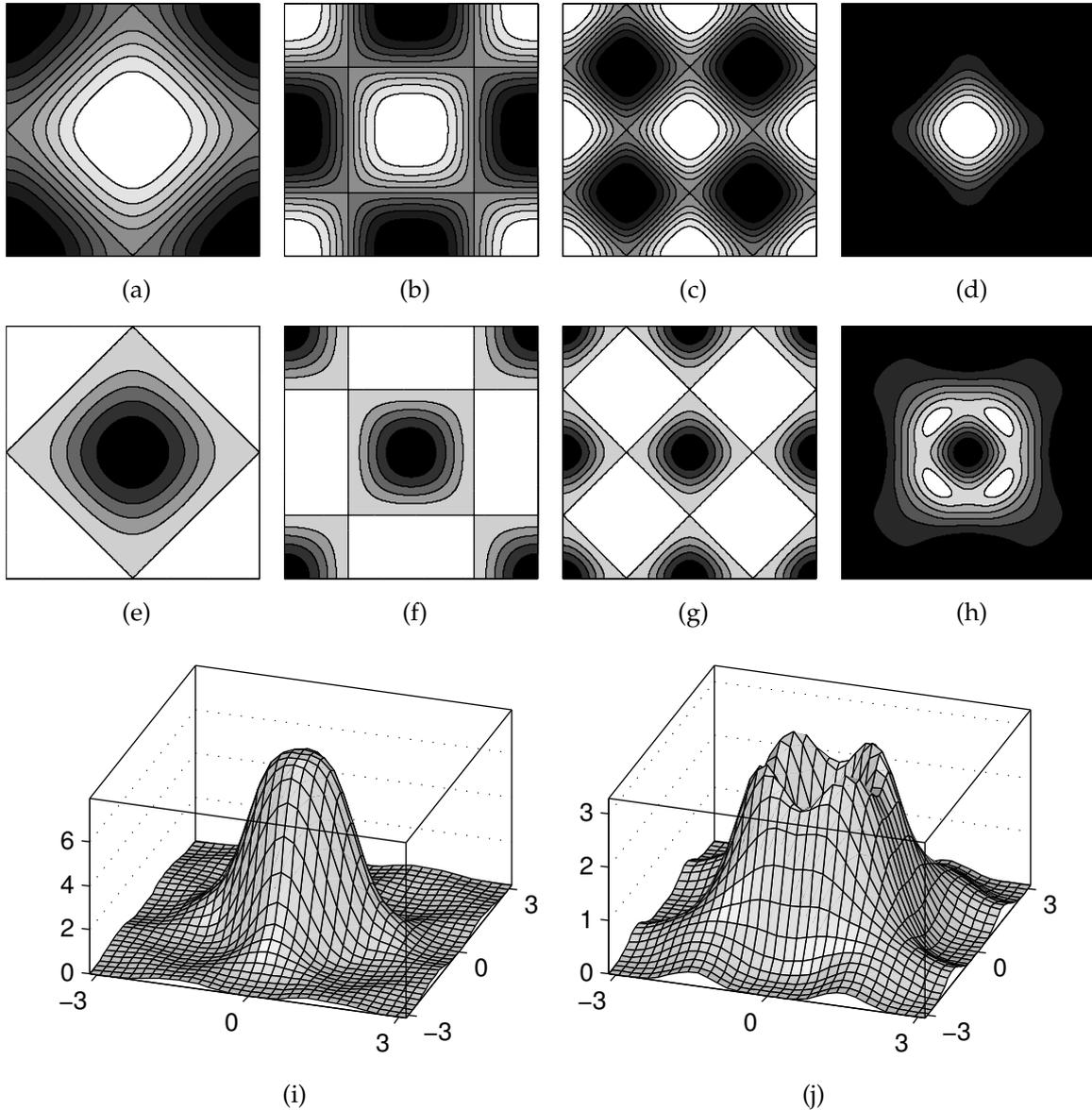


Figure 3.14: Support of the magnitude frequency responses of the synthesis filters. White areas represent smallest values and darker areas represent higher values. (a) Original synthesis low-pass filter from figure 3.12(c), upsampled with D_q (b) and D_q^2 (c). Product of these three filters is shown in (d) and (i). (e) Original synthesis high-pass filter from figure 3.12(d), upsampled with D_q (f) and D_q^2 (g). Corresponding high-pass filter shown in (h) and (j) is obtained as a product of low-pass filters from (a) and (b) and a last-level high-pass from (g).

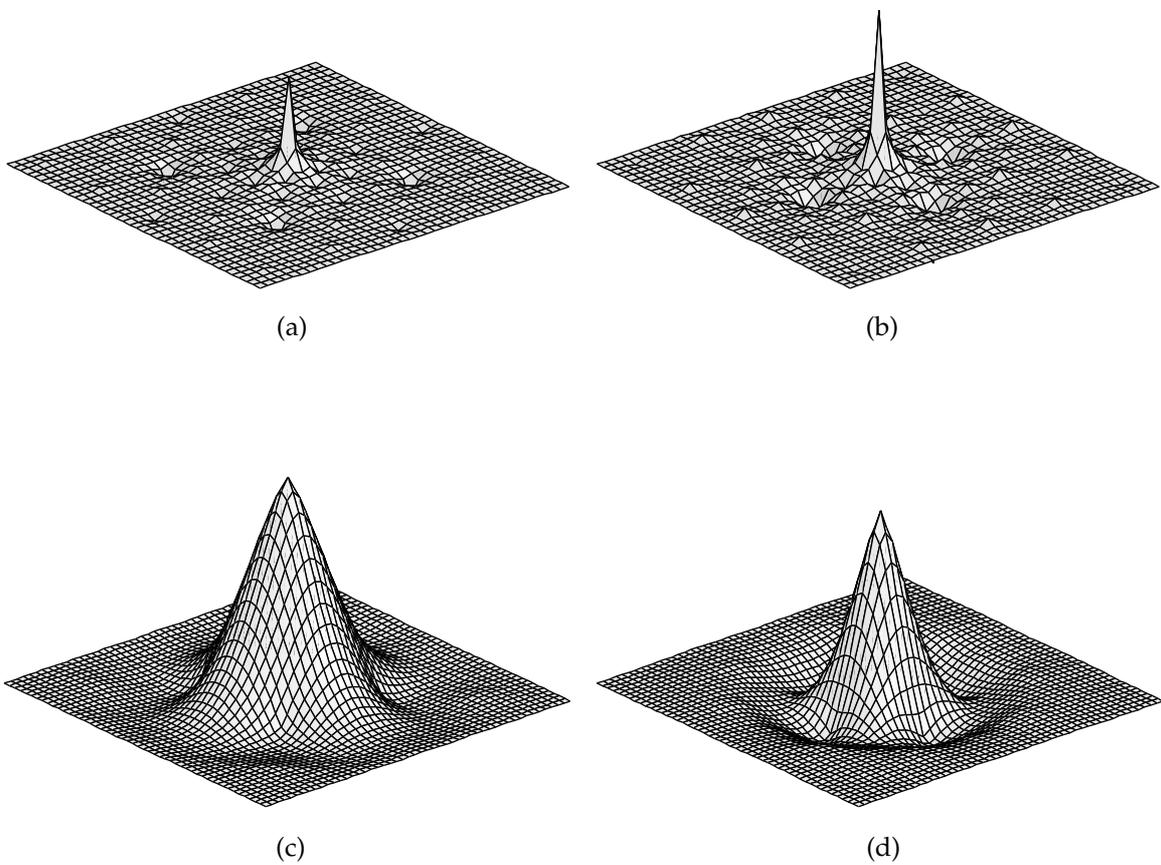


Figure 3.15: Impulse responses obtained after seven iterations on the low-pass channel. First row: dual (analysis) limit scale (a) and wavelet (b) functions. Second row: primal (synthesis) limit scale (c) and wavelet (d) functions.

Chapter 4

Filter Banks With Variable Parameters

The quincunx interpolating filter banks presented in section 3.4 have a number of good properties that can be exploited in various fields of image processing. Yet, for a given image, these filter banks are still not optimal. It is logical to expect that the modification of these filter banks that is tunable in accordance with specific image properties could outperform the fixed filter banks.

In this chapter we discuss the modification of a quincunx interpolating filter bank presented in section 3.4. The modified filter bank structure allows for a number of filter parameters to be tuned based on image properties, while still preserving good properties of the wavelet filter bank. The filter bank consists of two parts: the fixed part and the variable variable. The basic number of primal and dual vanishing moments is provided by the fixed part of the filter bank. The variable part of the filter bank can be independently adapted to the analyzed signal (image). The adaptation criteria and adaptation algorithms used are discussed in chapter 5.

4.1 Bad Way To Change Filter Parameters

Let us consider a variable analysis filter bank structure as shown in figure 4.1. There are the simplest predict and update filters used (see table 3.1):

$$P_2(z_1, z_2) = \frac{1}{4}(1 + z_1^{-1} + z_2^{-1} + z_1^{-1}z_2^{-1}), \quad (4.1a)$$

$$U_2(z_1, z_2) = \frac{1}{2}P_2^*(z_1, z_2) = \frac{1}{8}(1 + z_1 + z_2 + z_1z_2). \quad (4.1b)$$

The output of the prediction filter P_2 is multiplied by the weight parameter p_1 and output of the update U_2 filter is multiplied by the weight parameter u_1 . In this way, there are new predict and update filters obtained. By changing those weight parameters, one

can influence the properties of the corresponding analysis high-pass and low-pass filters. Of course, to satisfy the perfect reconstruction requirement, the same parameters should be used on the synthesis side making corresponding synthesis filters change in a fashion similar to the analysis filters.

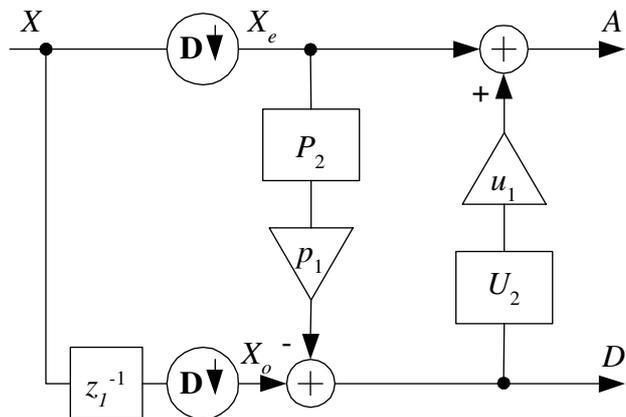


Figure 4.1: Analysis filter bank with one variable parameter in the predict stage and one variable parameter in the update stage.

4.1.1 Predict Step

Let us first change only the prediction parameter p_1 , while the $u_1 = 1$ (plain U_2 update filter), as shown in figure 4.2. Figure 4.4 shows the magnitude frequency responses of

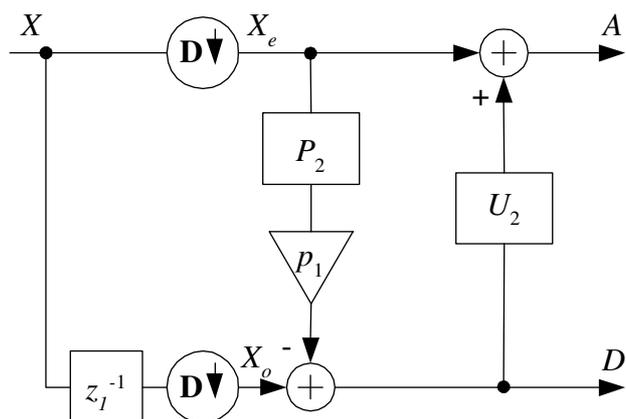


Figure 4.2: Analysis filter bank with one variable parameter in the predict stage.

the corresponding analysis high-pass filter for different values of the parameter p_1 . It is obvious that basic property of the high-pass filter, i.e. zero for the DC frequency, is lost for all values of p_1 except for $p_1 = 1$. Negative values of p_1 cause the filter's

magnitude frequency response to become a low-pass mirror image of the magnitude frequency response obtained with the same, yet positive value of p_1 . Figure 4.3 shows the magnitude frequency responses for all four corresponding filters (analysis and synthesis) obtained with $p_1 = 1.25$. Take note of the biorthogonality property: the analysis low-pass as a mirrored synthesis high-pass and the analysis high-pass as a mirrored synthesis low-pass. The impulse responses corresponding to the limit analysis and

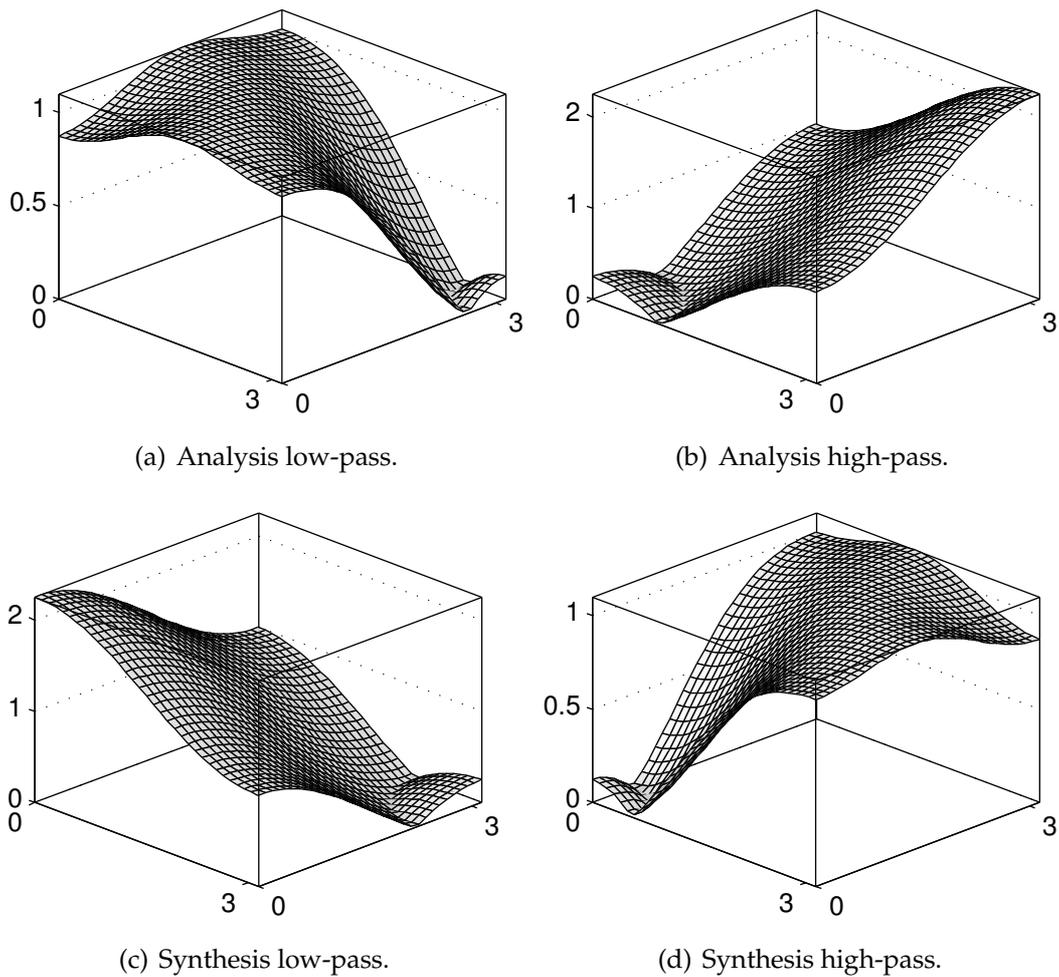


Figure 4.3: Magnitude frequency responses obtained by using a weighted prediction filter $1.25P_2$ and update filter U_2 .

synthesis wavelet functions (after seven iterations) obtained by using different p_1 parameters are shown in figure 4.5. It is evident that the convergence and regularity problems appear for $p_1 \neq 1$.

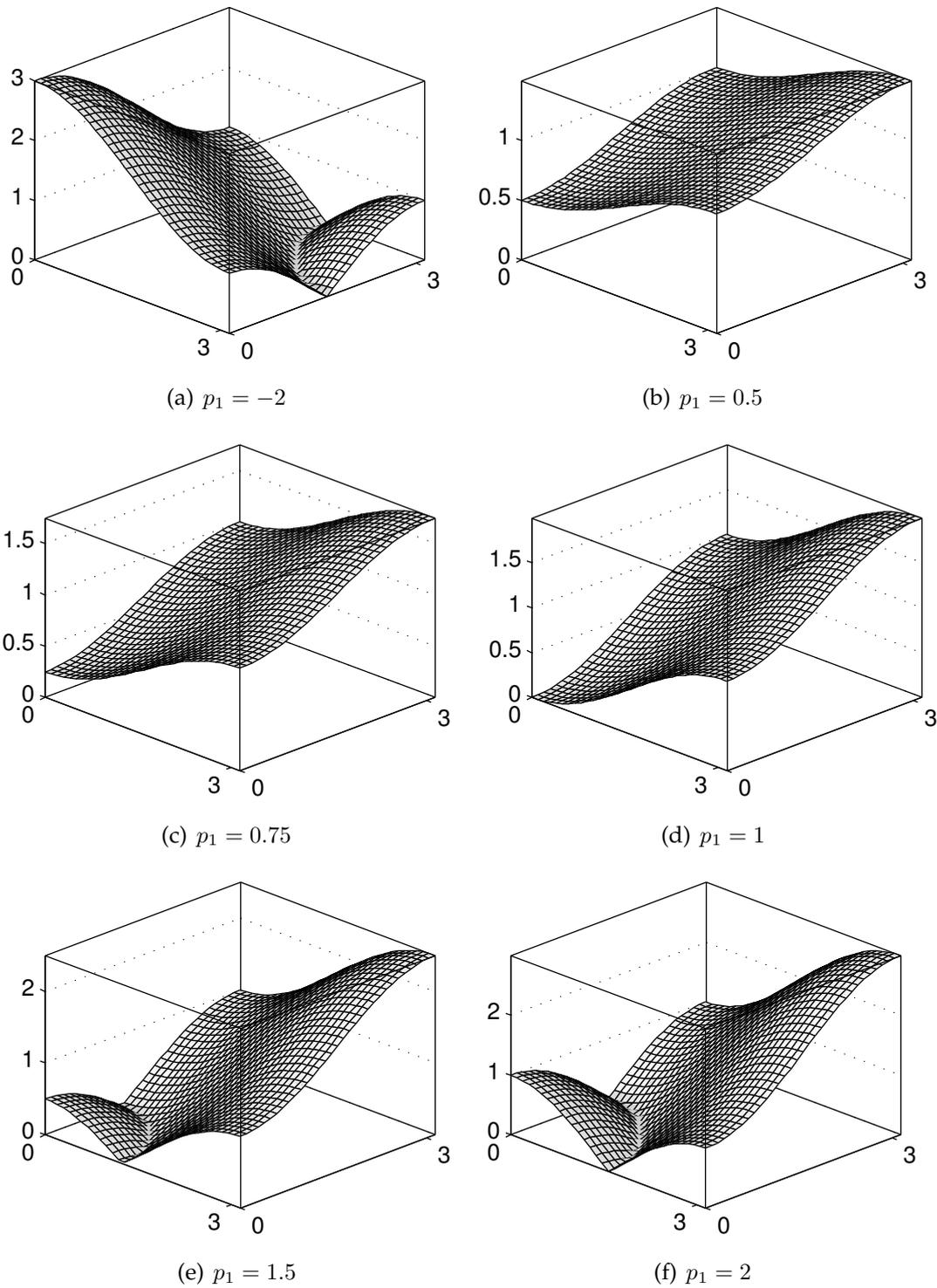
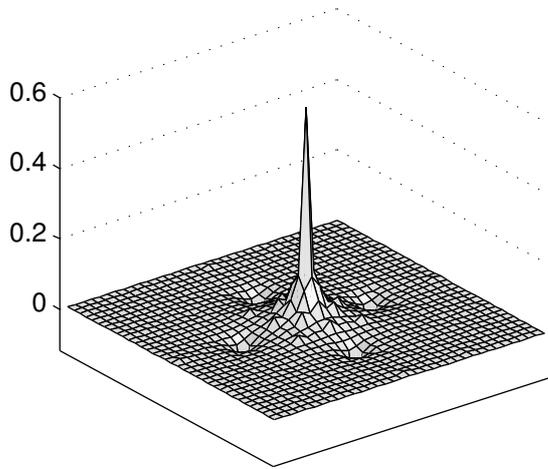
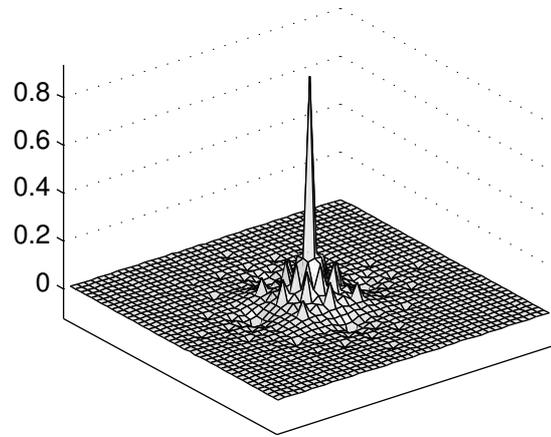


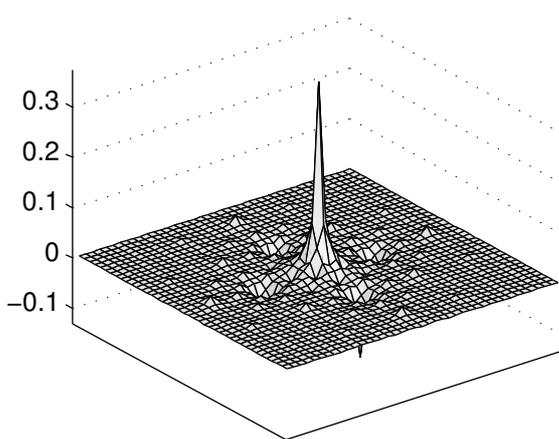
Figure 4.4: Magnitude frequency responses of the analysis high-pass filter obtained by using a weighted prediction filter p_1P_2 and update filter U_2 .



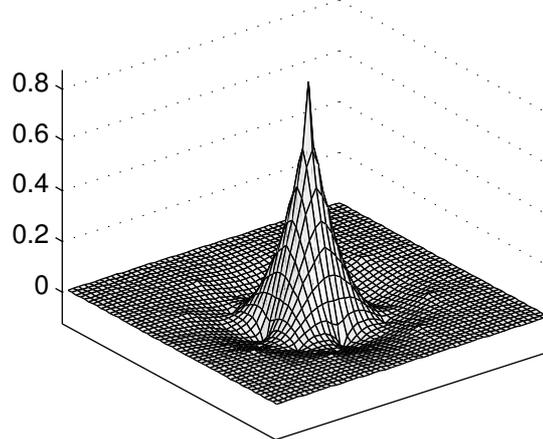
(a) $p_1 = 0.5$



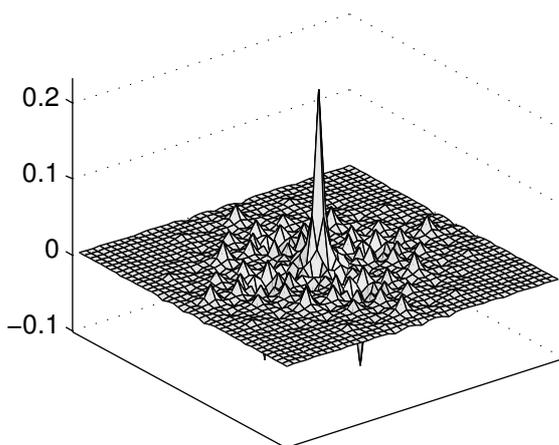
(b) $p_1 = 0.5$



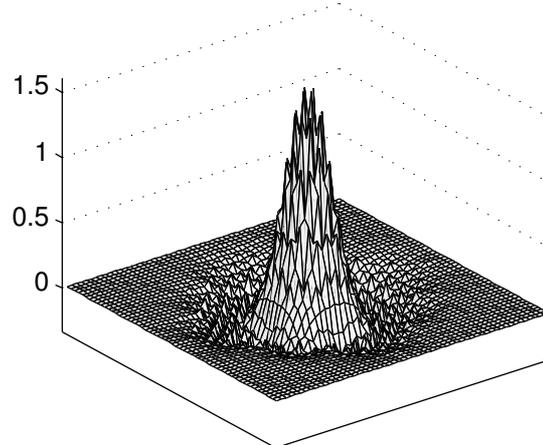
(c) $p_1 = 1$



(d) $p_1 = 1$



(e) $p_1 = 1.5$



(f) $p_1 = 1.5$

Figure 4.5: Analysis (left column) and synthesis (right column) limit wavelets obtained by using a weighted prediction filter $p_1 P_2$ and update filter U_2 .

4.1.2 Update Step

Now let us examine the effects caused by changing the properties of the update branch. The predict filter is fixed to P_2 ($p_1 = 1$) and the parameter u_1 is being changed (figure

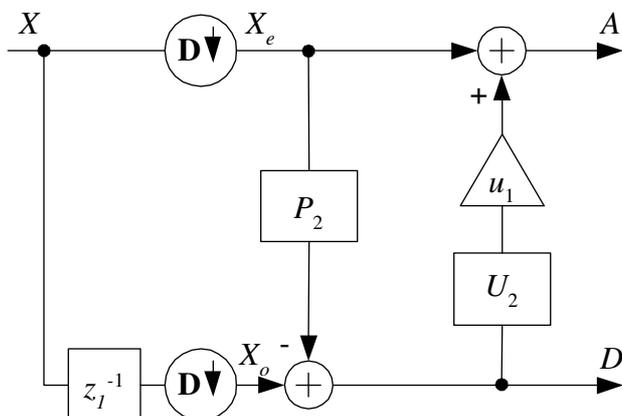


Figure 4.6: Analysis filter bank with one variable parameter in the update stage.

4.6). As it can be seen in figure 4.7, the low-pass filter loses zero at $(\omega_1, \omega_2) = (\pi, \pi)$ for $u_1 \neq 1$. For positive values of u_1 greater than 1, a "ditch" of zeros is introduced, turning the low-pass filter into a kind of band-stop filter. These variations of the update filter produce an inopportune effect on the properties of the iterated filter bank.

As shown in figure 4.8(c), the analysis wavelet obtained with fixed P_2 and U_2 has an irregular structure already. Its irregularity is being increased by further increasing the parameter u_1 . On the reconstruction side, the change in the update filter affects only the high-pass filter (see figure 2.21), while the synthesis low-pass filter remains the same. This makes the corresponding limit scale function the same and invariant with regard to changes in the u_1 parameter. Therefore, the regularity of the limit wavelet functions is kept (see figures 4.8(b), 4.8(d) and 4.8(f)).

However, for $u_1 \neq 1$ these are not true wavelet functions because $\int \int \psi^{(N)}(t_1, t_2) \neq 0$. In order to enable that, it is necessary that the synthesis high-pass filter has zero at $(\omega_1, \omega_2) = (0, 0)$ which is equivalent to the necessity of the zero of the low-pass analysis filter at the aliasing frequency $(\omega_1, \omega_2) = (\pi, \pi)$ [Kovačević 92].

It is obvious that the variable filter bank structure shown in figure 4.1 is not a good choice, since some properties of wavelet filter banks are degraded or even lost and therefore a decomposition of a signal using this kind of a filter bank iterated on a low-pass channel does not give a wavelet decomposition.

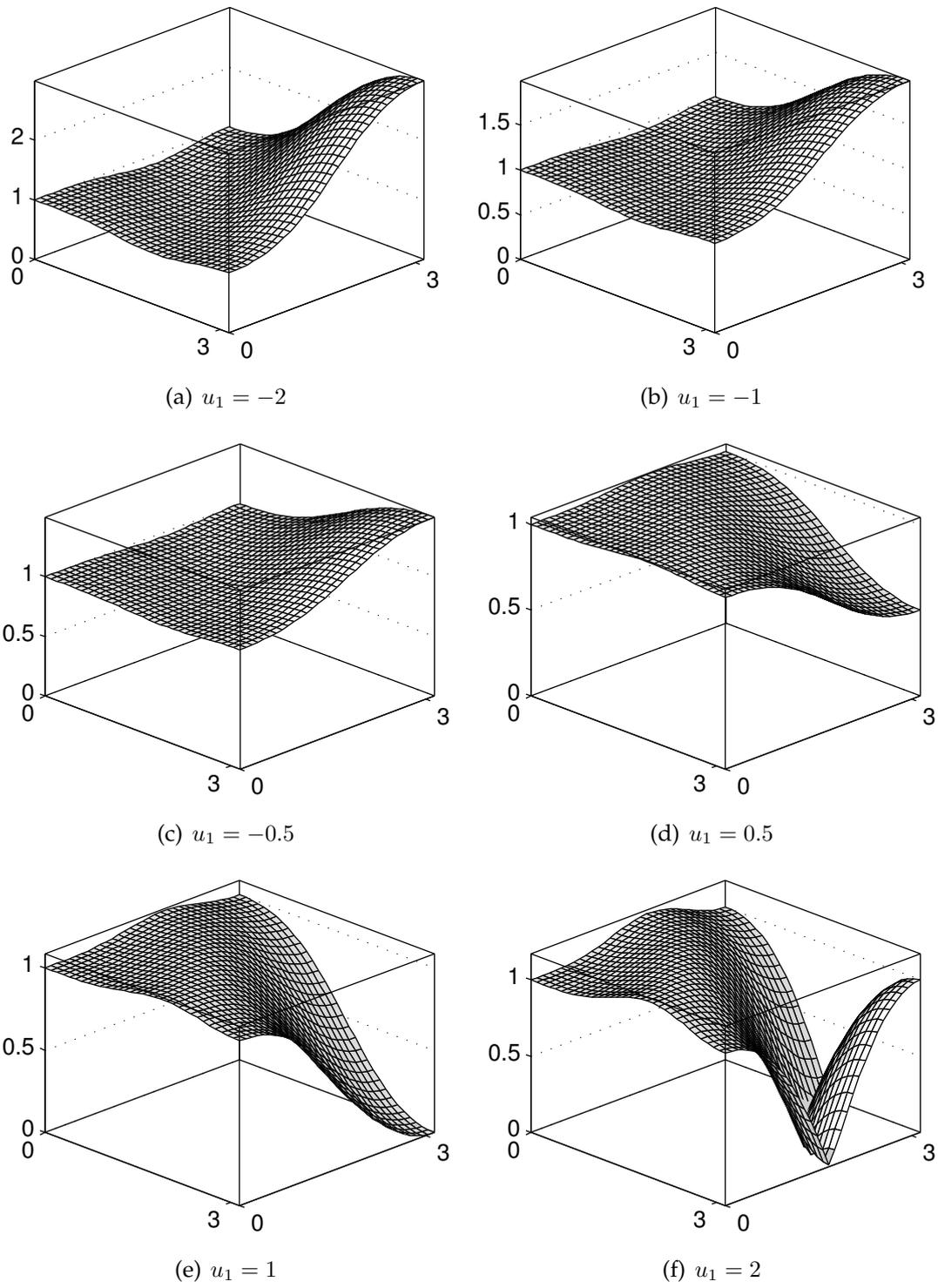
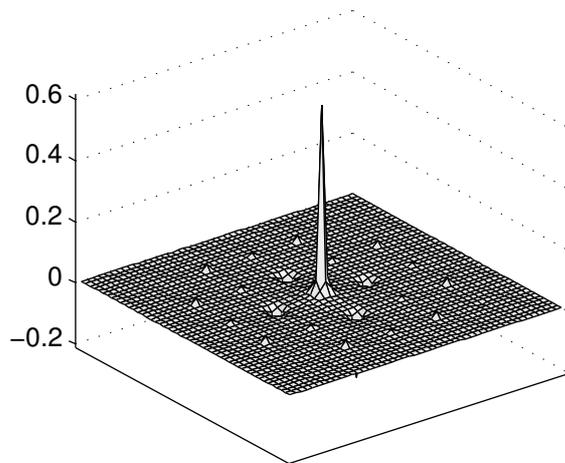
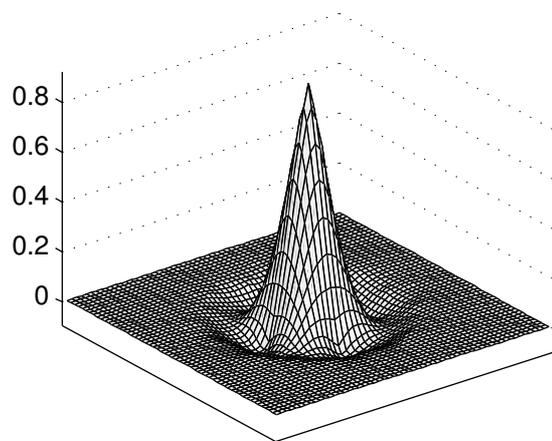


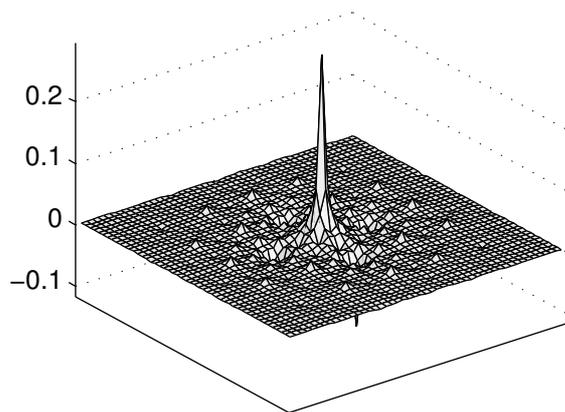
Figure 4.7: Magnitude frequency responses of the analysis low-pass filter H_0 obtained by using the prediction filter P_4 and an update filter $u_1 U_2$.



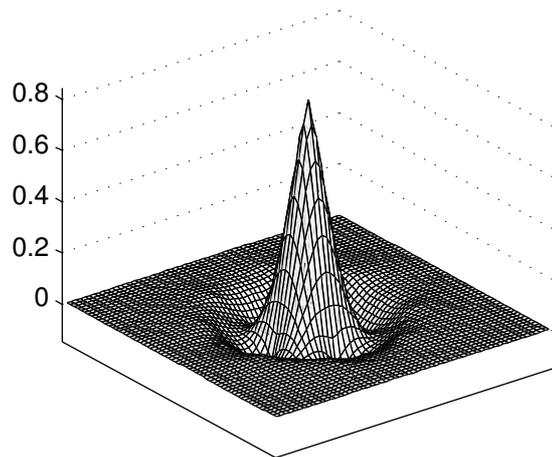
(a) $u_1 = 0.5$



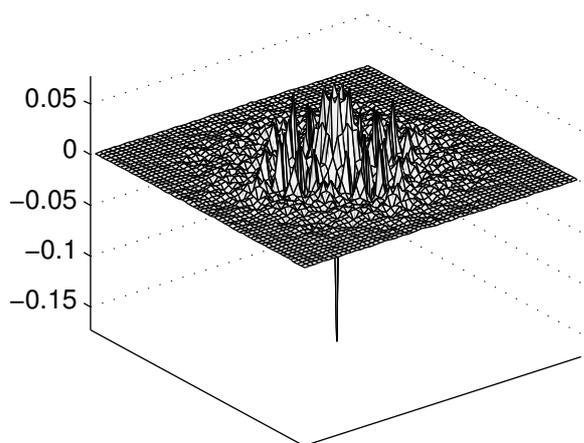
(b) $u_1 = 0.5$



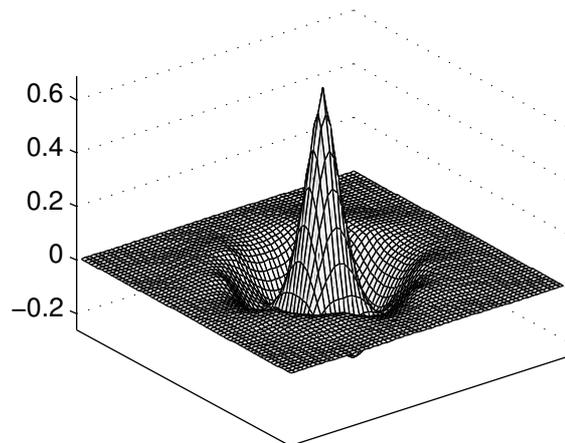
(c) $u_1 = 1$



(d) $u_1 = 1$



(e) $u_1 = 2$



(f) $u_1 = 2$

Figure 4.8: Analysis (left column) and synthesis (right column) wavelets obtained by using the prediction filter P_4 and a weighted update filter $u_1 U_2$.

4.2 Good Way To Change Filter Parameters

We will show now that it is possible to have a certain degree of freedom in changing filter bank parameters while still preserving good properties of the wavelet decomposition.

4.2.1 Predict Step

Instead of directly applying the predictor of a desired order, we create a structure that enables us to split the prediction filter in the fixed and variable part. The fixed part will provide a desired number of dual vanishing moments while the variable part will be used to fine-tune the desired properties of the wavelet decomposition in accordance with the analyzed image. As shown in figure 4.9, the prediction filter is constructed as a weighted sum of additive components:

$$P = p_1 P_2 + p_2 (P_4 - P_2) + p_3 (P_6 - P_4) + p_4 (P_8 - P_6). \quad (4.2)$$

If multiplying parameters $\{p_1, p_2, p_3, p_4\}$ are chosen from sets $\{1, 0, 0, 0\}$, $\{1, 1, 0, 0\}$,

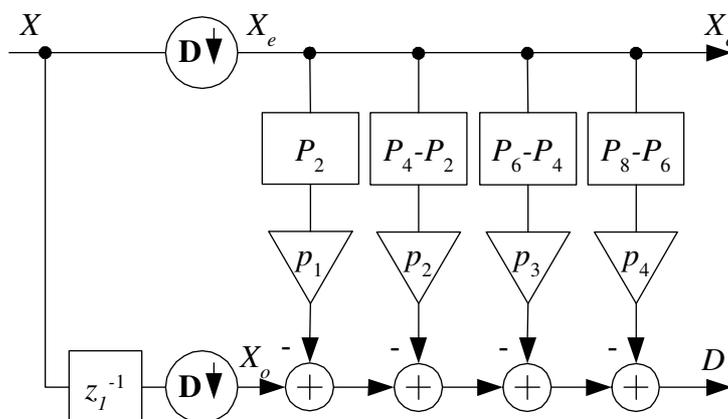


Figure 4.9: Dual lifting structure with four variable parameters: p_1, p_2, p_3 and p_4 .

$\{1, 1, 1, 0\}$ and $\{1, 1, 1, 1\}$, then the prediction filters P_2, P_4, P_6 and P_8 respectively are obtained. The prediction structure with desired properties is obtained as follows:

Fix the desired number of dual vanishing moments by setting first n prediction parameters to 1: $p_1 = \dots = p_n = 1$. In this way, $2n$ dual vanishing moments are guaranteed and cannot be lost, no matter what values the remaining free parameters may take.

Vary the remaining free parameters to additionally improve the properties of the filter bank.

In the case of fixing $p_2 = 1$, the two vanishing moments are guaranteed and the overall prediction filter can be expressed as:

$$P = P_2 + p_2(P_4 - P_2) + p_3(P_6 - P_4) + p_4(P_8 - P_6). \quad (4.3)$$

In the prediction structure, there are three parameters left which can be varied. We will show now that these parameters can be freely changed without affecting the number of dual vanishing moments. The dual vanishing moments condition has already been expressed in the relation 3.29 as

$$P\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \quad (4.4)$$

By combining equations 4.3 and 4.4 the DM condition can be expressed as

$$\begin{aligned} (P_2 + p_2(P_4 - P_2) + p_3(P_6 - P_4) + p_4(P_8 - P_6))\pi(\mathcal{Z}^2) &= \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}), \\ P_2\pi(\mathcal{Z}^2) + p_2(P_4 - P_2)\pi(\mathcal{Z}^2) + p_3(P_6 - P_4)\pi(\mathcal{Z}^2) + p_4(P_8 - P_6)\pi(\mathcal{Z}^2) &= \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}). \end{aligned} \quad (4.5)$$

Since P_2 is a Neville filter of order 2 and shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$, the relation 4.4 can be applied to it, yielding

$$\pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) + p_2(P_4 - P_2)\pi(\mathcal{Z}^2) + p_3(P_6 - P_4)\pi(\mathcal{Z}^2) + p_4(P_8 - P_6)\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}), \quad (4.6)$$

which further simplifies the DM condition to

$$p_2(P_4 - P_2)\pi(\mathcal{Z}^2) + p_3(P_6 - P_4)\pi(\mathcal{Z}^2) + p_4(P_8 - P_6)\pi(\mathcal{Z}^2) = 0 \quad \text{for } \pi \in \Pi_2. \quad (4.7)$$

To prove that the DM condition stands for the given filter bank structure, let us consider a filter section $p(P_{\tilde{M}} - P_{\tilde{N}})$, where p is a parameter that can take any positive or negative value. $P_{\tilde{M}}$ and $P_{\tilde{N}}$ are Neville filters of shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$ and order $\tilde{M} \geq 2$ and $\tilde{N} \geq 2$ respectively. Applying this filter section to a polynomial sequence $\pi \in \Pi_2$ and using the relation 4.4

$$p(P_{\tilde{M}} - P_{\tilde{N}})\pi(\mathcal{Z}^2) = p(\pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) - \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t})) = 0 \quad \text{for } \pi \in \Pi_2. \quad (4.8)$$

Therefore,

$$\underbrace{p_2(P_4 - P_2)\pi(\mathcal{Z}^2)}_{=0} + \underbrace{p_3(P_6 - P_4)\pi(\mathcal{Z}^2)}_{=0} + \underbrace{p_4(P_8 - P_6)\pi(\mathcal{Z}^2)}_{=0} = 0 \quad \text{for } \pi \in \Pi_2, \quad (4.9)$$

which concludes the proof that 2 dual vanishing moments are guaranteed no matter what values the remaining free parameters p_2, p_3 and p_4 may take!

In the case of a structure with two filter parameters being fixed to 1, i.e. $p_1 = p_2 = 1$ the overall prediction structure can be expressed as

$$P = P_4 + p_3(P_6 - P_4) + p_4(P_8 - P_6). \quad (4.10)$$

The similar procedure shown above can be used in order to prove that 4 dual vanishing moments are guaranteed no matter what values the remaining free parameters p_3 and p_4 may take. To simplify the generalization, let us define the predict filter $P_{\tilde{N}+}$ that is obtained by using a structure from figure 4.9 where the first $\tilde{N}/2$ prediction parameters are set to 1, i.e. $p_1 = \dots = p_{\tilde{N}/2} = 1$ and the remaining parameters are free to change. Therefore,

$$P_{2+} = P_2 + p_2(P_4 - P_2) + p_3(P_6 - P_4) + p_4(P_8 - P_6), \quad (4.11a)$$

$$P_{4+} = P_4 + p_3(P_6 - P_4) + p_4(P_8 - P_6), \quad (4.11b)$$

$$P_{6+} = P_6 + p_4(P_8 - P_6). \quad (4.11c)$$

The generalization follows:

If there is the filter $P_{\tilde{N}+}$ chosen for the predict step of the filter bank shown in the figure 3.9, then \tilde{N} dual vanishing moments are guaranteed no matter what values the remaining free parameters may take.

Changing the p_2 Parameter

Let us analyze the filter bank structure which uses just two prediction branches: first one is fixed ($p_1 = 1$) and the second branch parameter p_2 is being varied. Therefore, the complete prediction filter is obtained as $P = P_2 + p_2(P_4 - P_2)$. For the update step, a fixed filter U_2 is used.

Since $p_1 = 1$, we get a double zero at a DC frequency of the corresponding analysis high-pass filter, which guarantees two vanishing moments of the analysis (dual) wavelet function. That double zero automatically corresponds to the double zero at the aliasing frequency of the interpolating analysis low-pass filter.

Figure 4.10 shows the magnitude frequency responses for the corresponding analysis and synthesis filters obtained by using $P = P_2 + 3(P_4 - P_2)$. We see that necessity of zeros at the aliasing frequency for both synthesis and analysis low-pass filters is satisfied. Figures 4.11 and 4.12 show the magnitude frequency responses of the high-pass analysis filter H_1 obtained with different values of the p_2 parameter. As expected, zero at the DC frequency is preserved for all values of the p_2 parameter. For positive values of p_2 higher than 1, there appear additional zeros forming a "ditch" around the DC frequency (see figure 4.13). As the p_2 increases, the zero ditch widens and grows into a

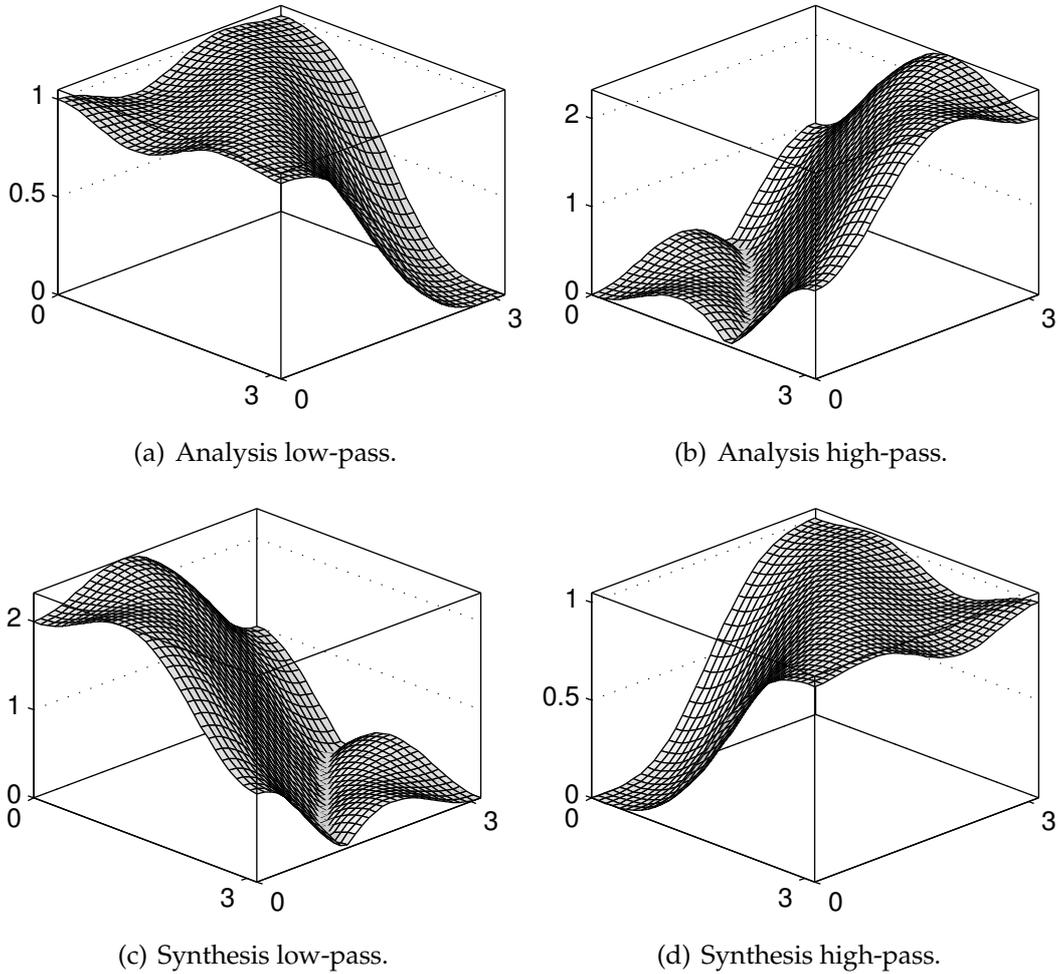


Figure 4.10: Magnitude frequency responses obtained by using weighted prediction filter $P = P_2 + 3(P_4 - P_2)$ and update filter U_2 .

diamond shape with the limit $|\omega_1 \pm \omega_2| = \pi$. The zero ditch can be moved by changing the p_2 parameter and it can be adjusted to cancel the undesired frequency components of the analyzed image. In chapter 5 we show how this property can be used to minimize the detail signal.

For larger negative values of p_2 , zero ditches appear in the high frequency areas outside $|\omega_1 \pm \omega_2| = \pi$ as shown in figure 4.13. The analysis high-pass filter turns into a kind of band-pass and there appears a problem with convergence.

In figure 4.14, there are examples of too big p_2 parameter values causing convergence and regularity problems. It is evident that not all p_2 parameters result in good wavelet filter banks. "Good" p_2 parameters are the ones of small positive and negative values. The analysis and synthesis limit wavelet functions with p_2 parameters from an acceptable range are shown in figures 4.15 and 4.16 respectively.

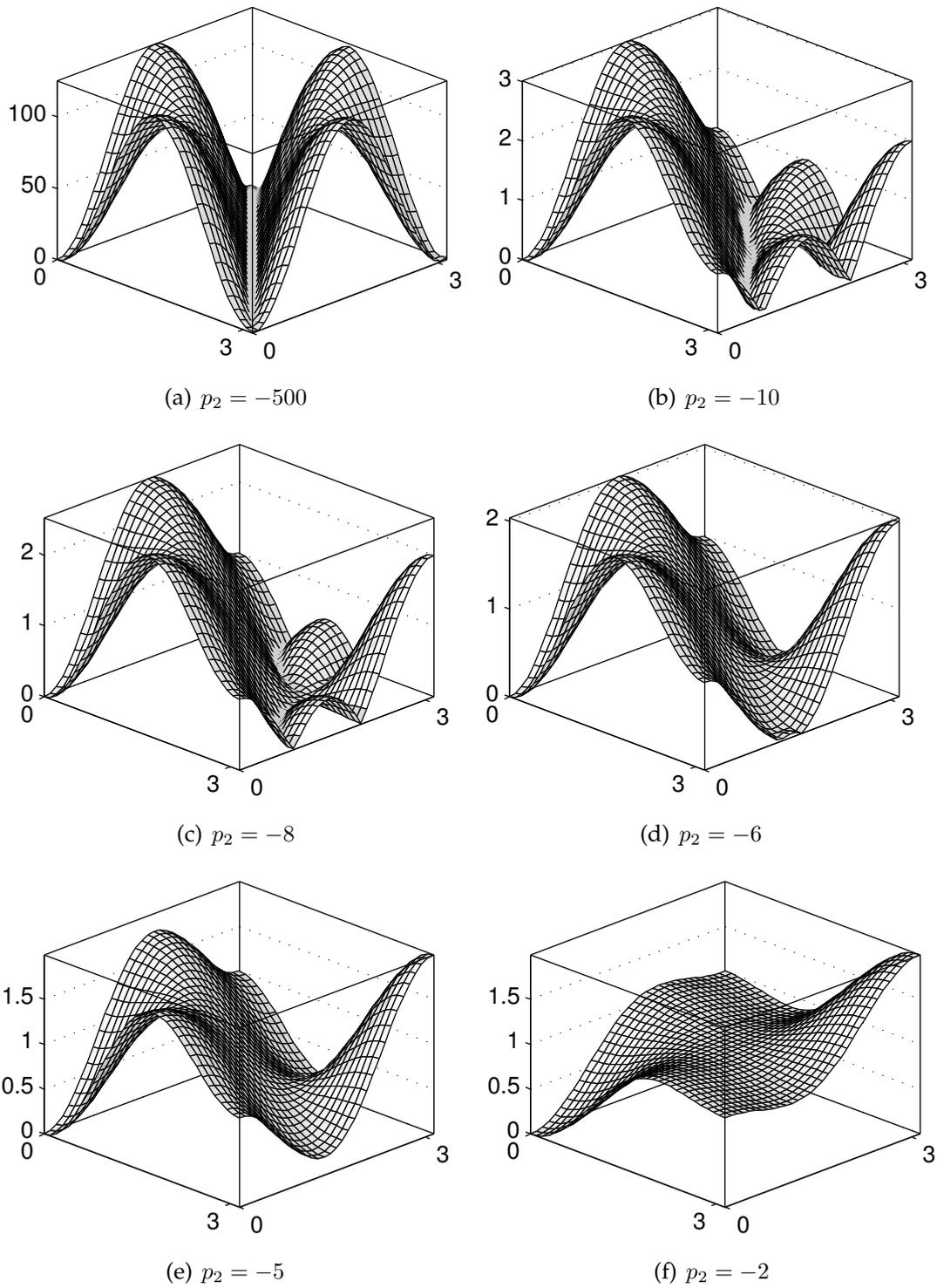


Figure 4.11: Magnitude frequency responses of the analysis high-pass filter obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ (with a range of negative values of p_2) and update filter U_2 .

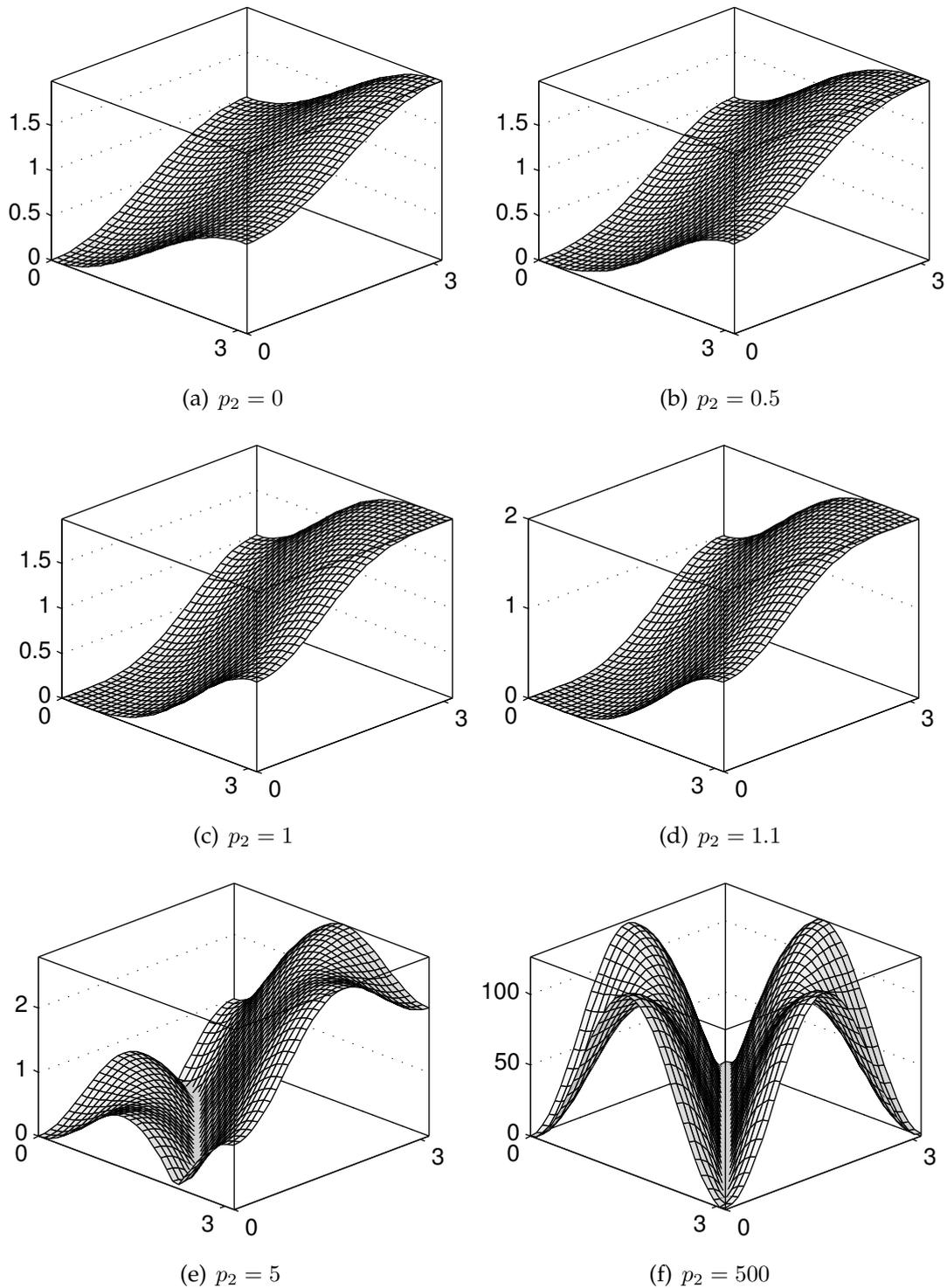


Figure 4.12: Magnitude frequency responses of the analysis high-pass filter obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ (with a range of positive values of p_2) and update filter U_2 .

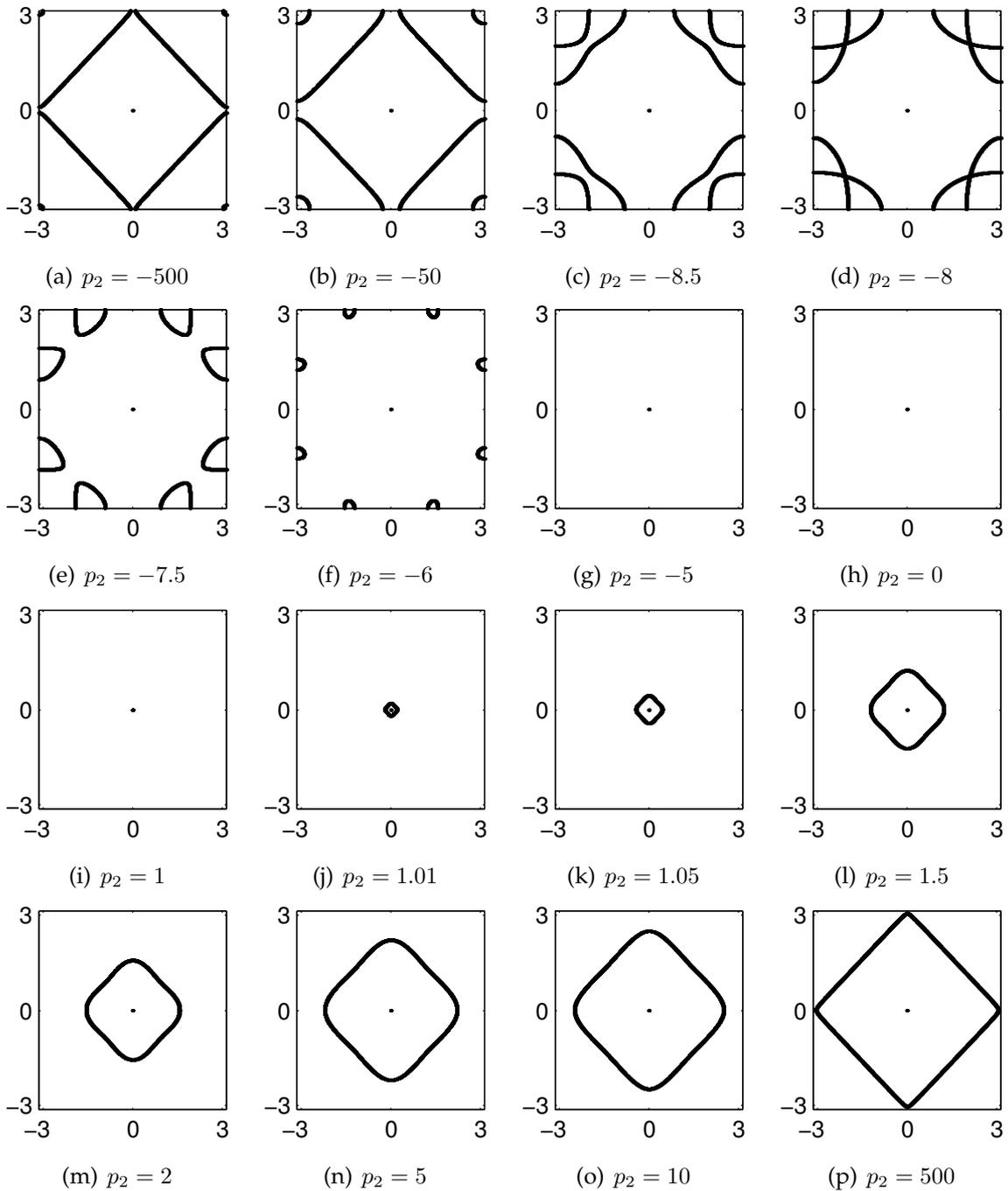


Figure 4.13: Zero locations of the magnitude frequency responses of the analysis high-pass filter obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ and update filter U_2 (see figures 4.11 and 4.12).

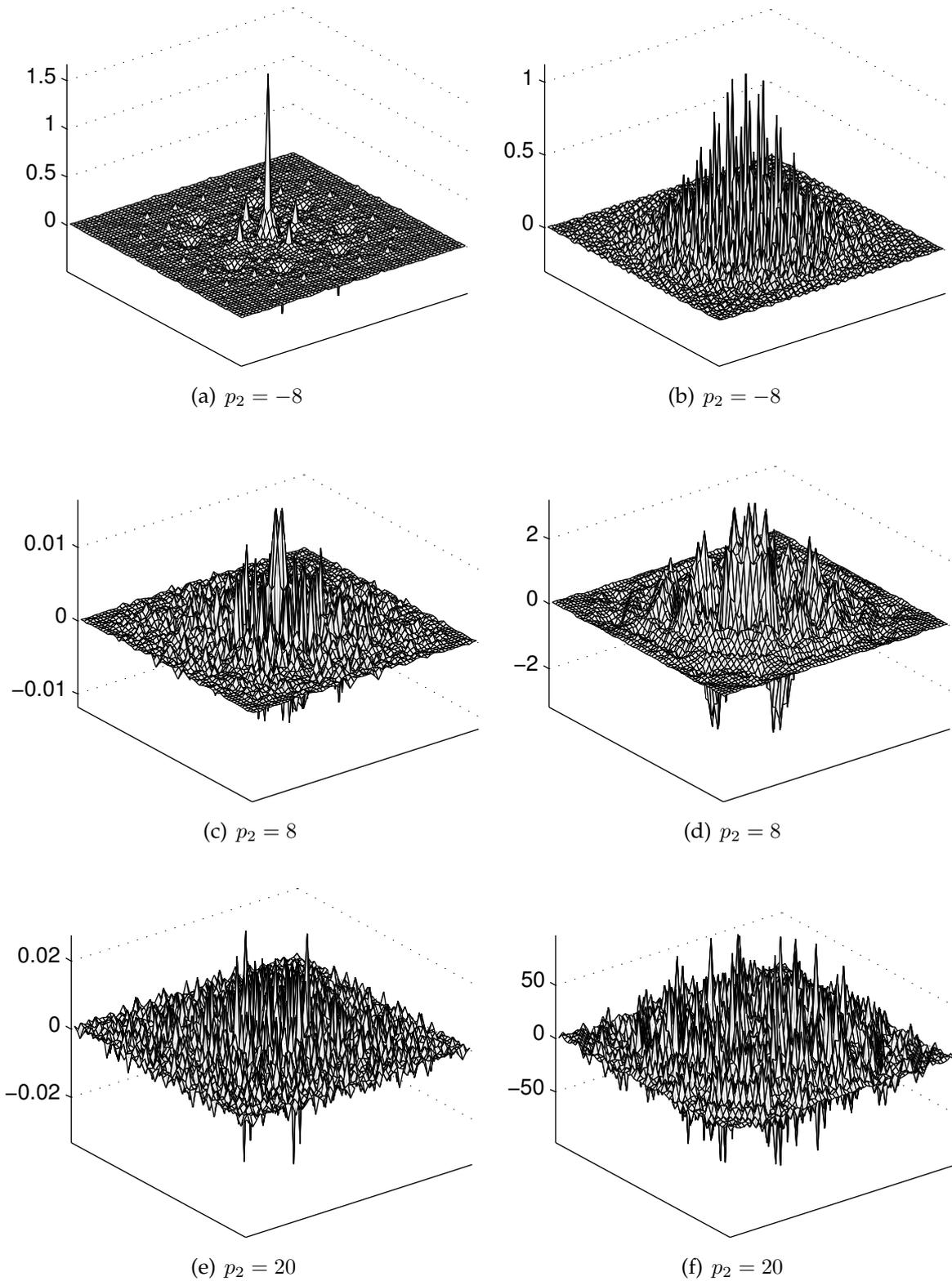


Figure 4.14: Analysis (left column) and synthesis (right column) limit wavelet functions obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ and update filter U_2 . Values of p_2 parameters used in these examples are too big, causing convergence and regularity problems.

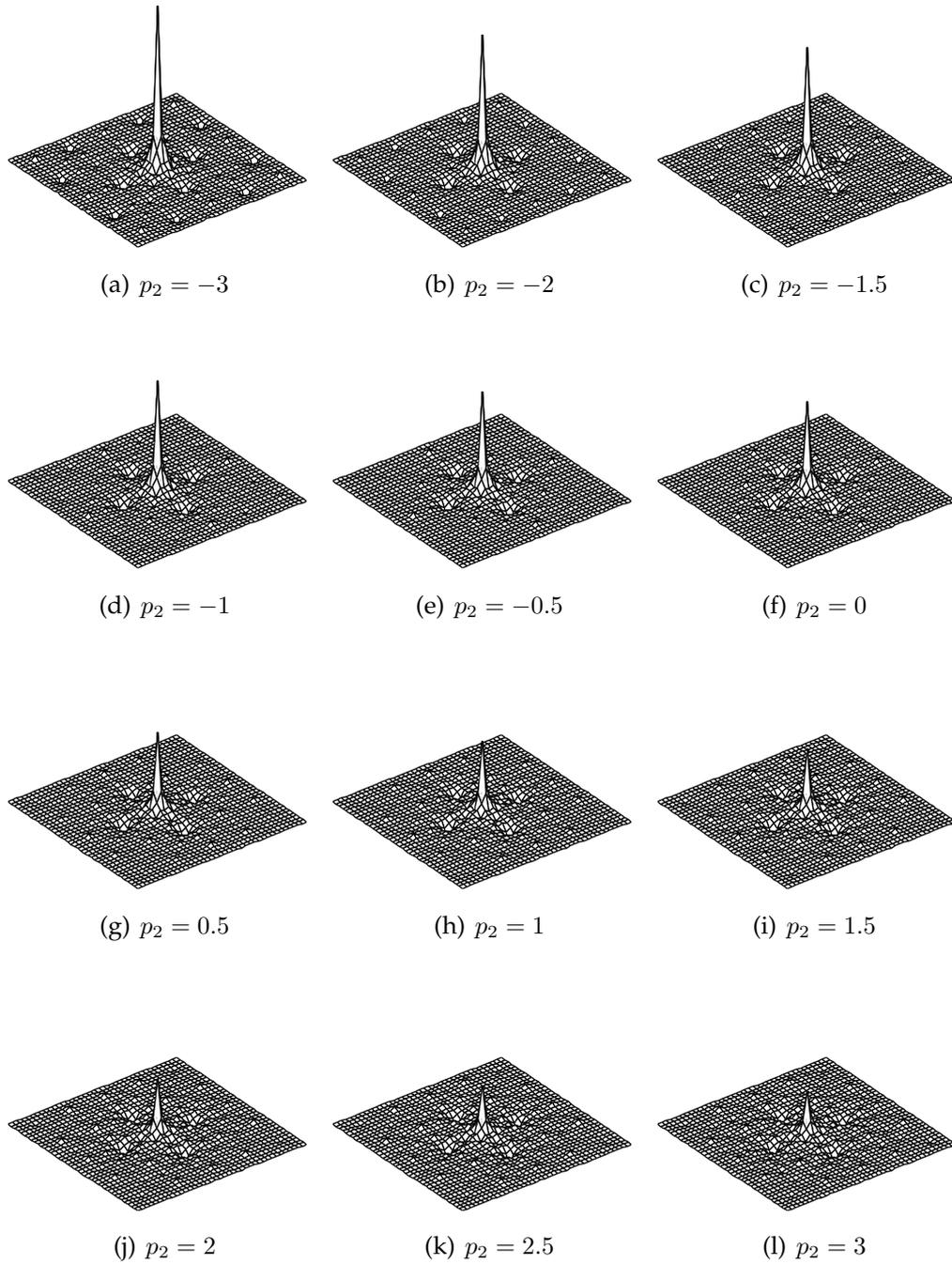


Figure 4.15: Analysis limit wavelet functions obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ and update filter U_2 .

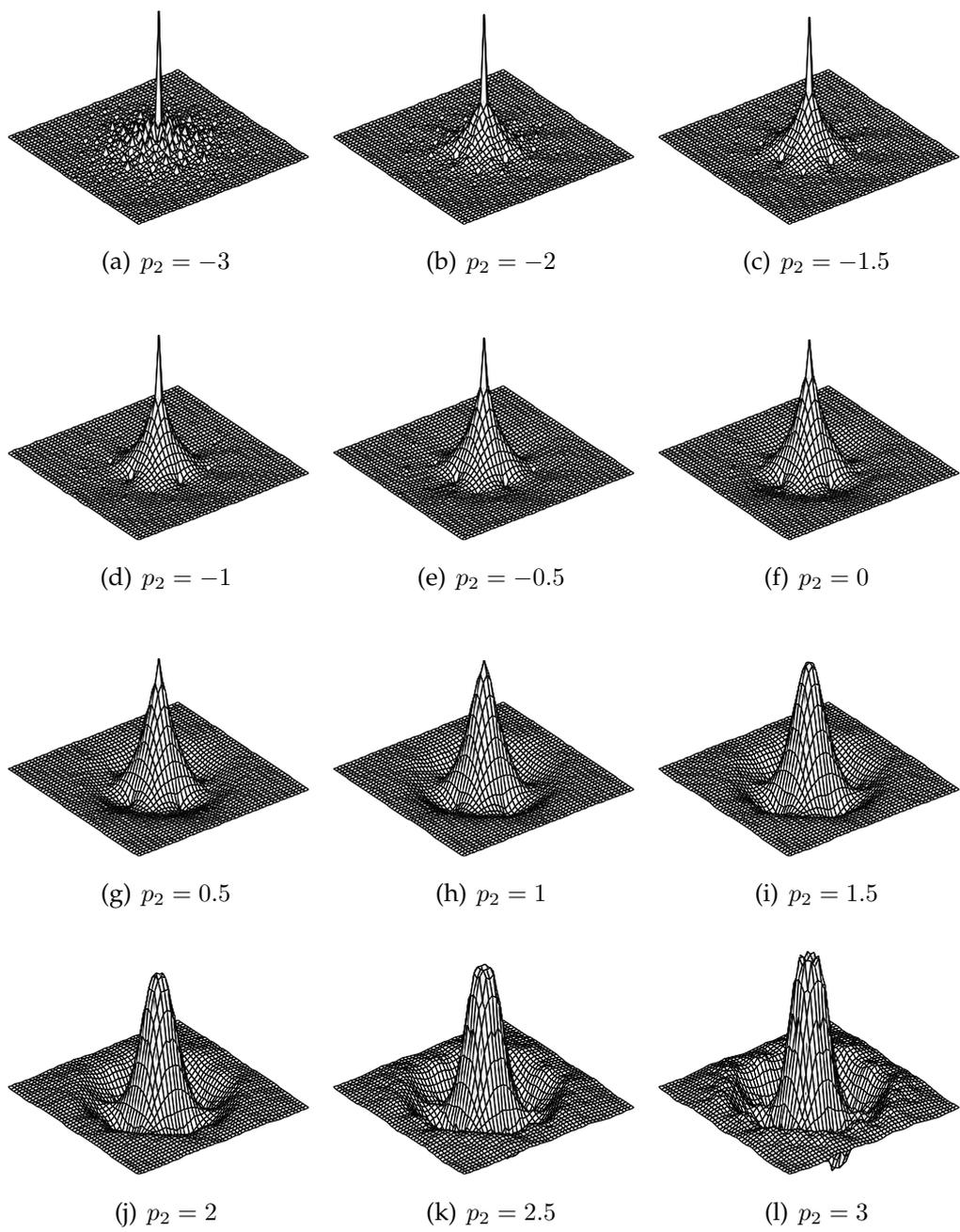


Figure 4.16: Synthesis limit wavelet functions obtained by using weighted prediction filter $P_2 + p_2(P_4 - P_2)$ and update filter U_2 .

4.2.2 Update Step

The update lifting step is split in a number of branches (figure 4.17) in a way similar to the predict step:

$$U = u_1 U_2 + u_2 (U_4 - U_2) + u_3 (U_6 - U_4) + u_4 (U_8 - U_6), \quad (4.12)$$

where $U_N = 1/2P_N^*$. Let us now prove that if the predict step guarantees two or more

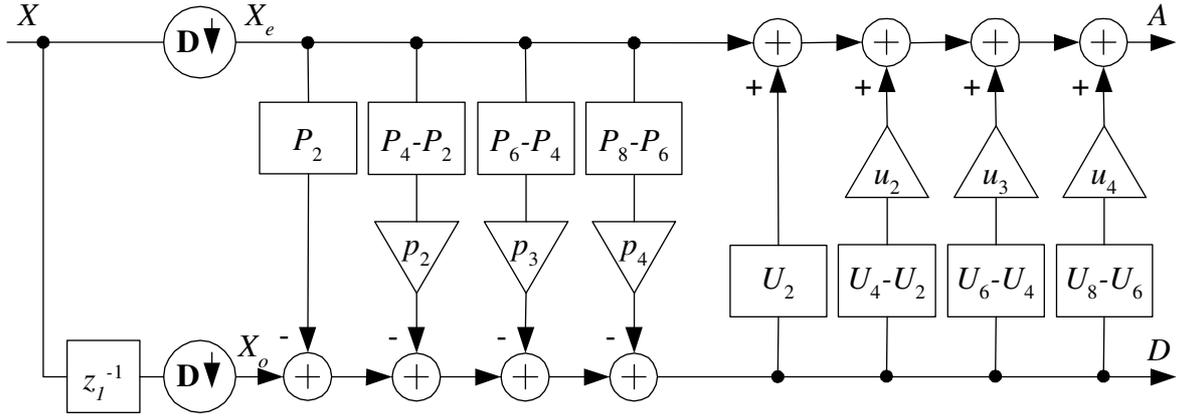


Figure 4.17: Filter bank with 2 fixed primal and 2 fixed dual vanishing moments and 3 parameters in both the predict stage and the update stage.

dual vanishing moments (it is the $P_{\tilde{N}+}$ filter with $\tilde{N} \geq 2$), the update structure having $u_1 = 1$, i.e.

$$U = U_2 + u_2 (U_4 - U_2) + u_3 (U_6 - U_4) + u_4 (U_8 - U_6), \quad (4.13)$$

provides 2 vanishing moments of the primal wavelet. By using the relation 3.30 the PM condition can be stated as

$$U^* \pi(\mathbf{D}\mathcal{Z}^2) + \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) - U^* P_{2+}^* \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_2, \quad (4.14)$$

where

$$\begin{aligned} P_{2+}^* \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) &= (P_2^* + p_2(P_4^* - P_2^*) + p_3(P_6^* - P_4^*) + p_4(P_8^* - P_6^*)) \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \\ &= P_2^* \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) + \underbrace{p_2(P_4^* - P_2^*) \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t})}_{=0} \\ &\quad + \underbrace{p_3(P_6^* - P_4^*) \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t})}_{=0} + \underbrace{p_4(P_8^* - P_6^*) \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t})}_{=0}, \\ &= \pi(\mathbf{D}\mathcal{Z}^2), \end{aligned} \quad (4.15)$$

since P_2^* is a Neville filter of order 2 and shift $-\tau = -\mathbf{D}^{-1}\mathbf{t}$. Therefore, the PM condition becomes

$$2U^* \pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \quad \text{for } \pi \in \Pi_2, \quad (4.16)$$

and by using the relation 4.13,

$$2(U_2^* + u_2(U_4^* - U_2^*) + u_3(U_6^* - U_4^*) + u_4(U_8^* - U_6^*))\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}). \quad (4.17)$$

By using the fact that $2U_N^* = P_N$, the relation 4.17 becomes

$$u_2(P_4 - P_2)\pi(\mathcal{Z}^2) + u_3(P_6 - P_4)\pi(\mathcal{Z}^2) + u_4(P_8 - P_6)\pi(\mathcal{Z}^2) = 0 \quad \text{for } \pi \in \Pi_2, \quad (4.18)$$

which is always true (see equations 4.8 and 4.9), no matter what values the parameters u_2 , u_3 and u_4 may take.

Let us define the update filter U_{N+} that is obtained by using a structure from figure 4.17 where the first $N/2$ prediction parameters are set to 1, i.e. $u_1 = \dots = u_{N/2} = 1$ and the remaining parameters are free to change. Therefore,

$$U_{2+} = U_2 + u_2(U_4 - U_2) + u_3(U_6 - U_4) + u_4(U_8 - U_6), \quad (4.19a)$$

$$U_{4+} = U_4 + u_3(U_6 - U_4) + u_4(U_8 - U_6), \quad (4.19b)$$

$$U_{6+} = U_6 + u_4(U_8 - U_6). \quad (4.19c)$$

The former result can be generalized:

If the predict step is a $P_{\tilde{N}+}$ filter (which guarantees \tilde{N} dual vanishing moments), and if the update step is a U_{N+} filter with $N \leq \tilde{N}$ then N primal vanishing moments are guaranteed no matter what values the remaining free parameters may take.

Predict and Update Steps Can Be Altered Independently

It is important to note that if the number of fixed parameters in the update structure does not exceed the number of fixed parameters in the predict structure, the remaining free update parameters can be changed independently of the free predict parameters! The number of primal (and dual vanishing moments) defined by the fixed filter section will be preserved!

Changing the u_2 Parameter

Figure 4.18 shows the magnitude frequency responses of the analysis low-pass filter H_0 obtained by using prediction filter P_4 and update filter $U_2 + u_2(U_4 - U_2)$. It can be seen that zero at the aliasing frequency is always guaranteed, no matter what the value of u_2 . This is a good start to obtaining the convergence of the limit scale (and wavelet) functions (see figure 4.19).

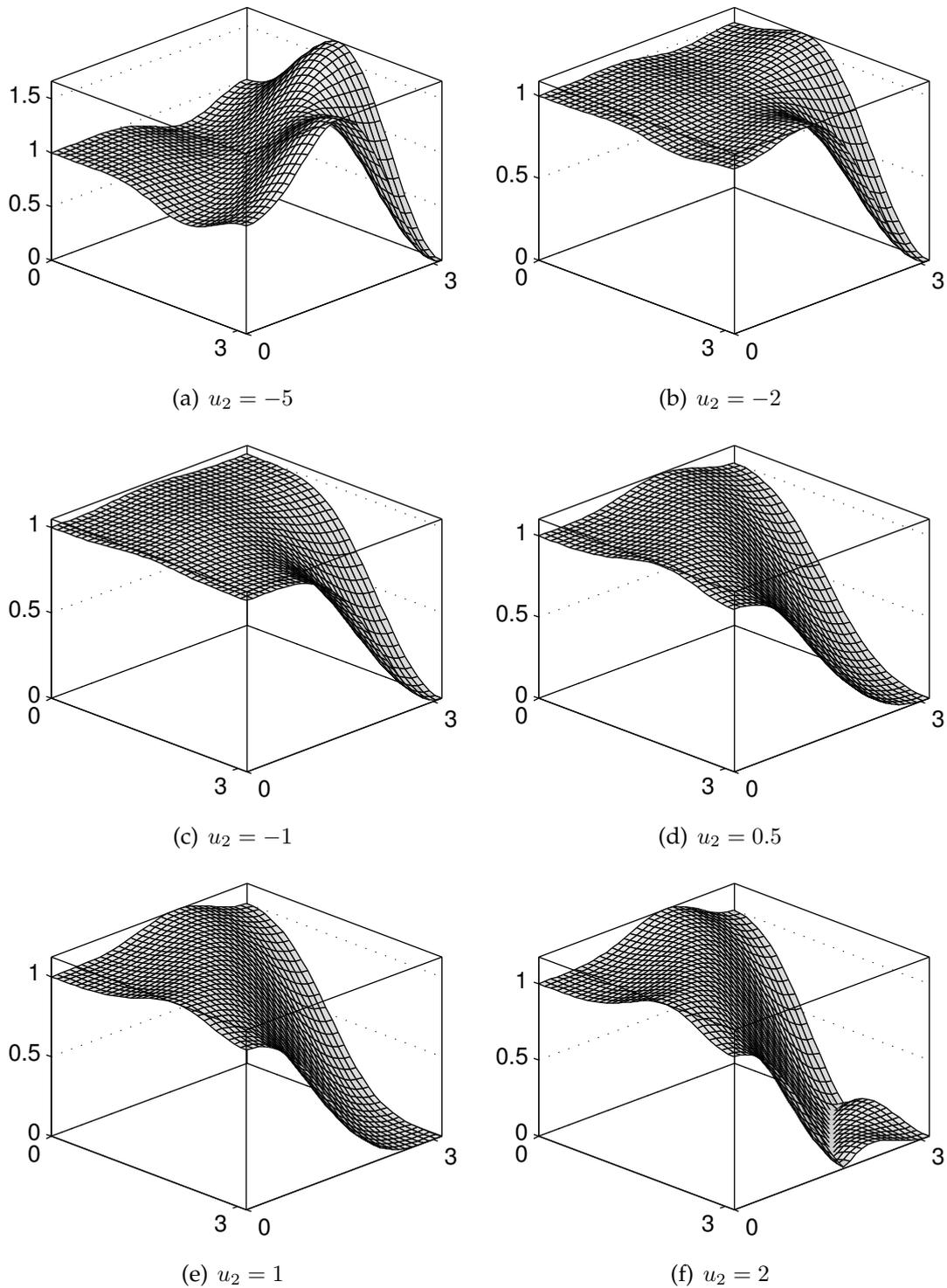


Figure 4.18: Magnitude frequency responses of the analysis low-pass filter H_0 obtained by using prediction filter P_4 and update filter $U_2 + u_2(U_4 - U_2)$.

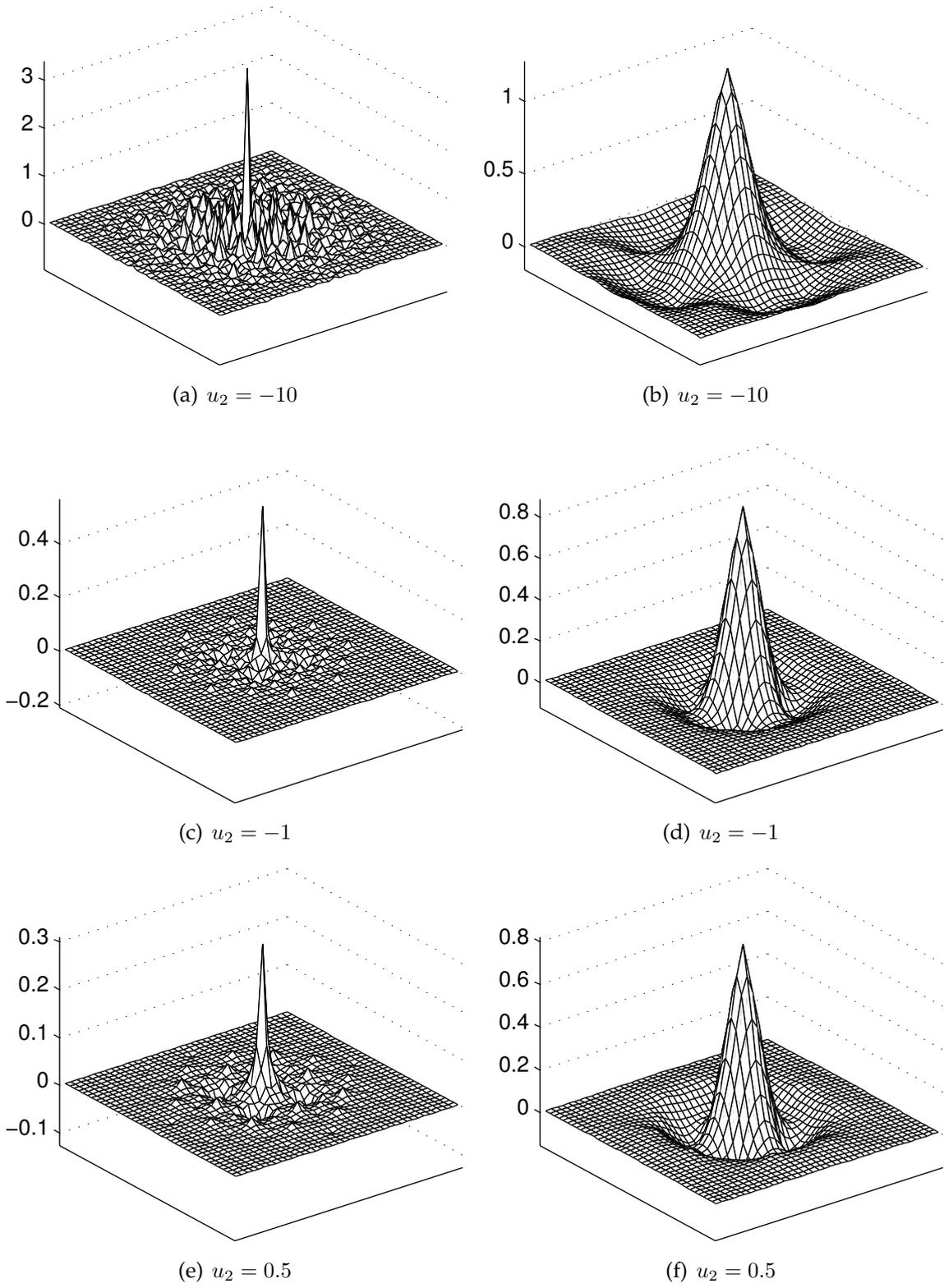


Figure 4.19: Analysis (left column) and synthesis (right column) limit wavelet functions obtained by using prediction filter P_4 and weighted update filter $U_2 + u_2(U_4 - U_2)$. *Continued...*

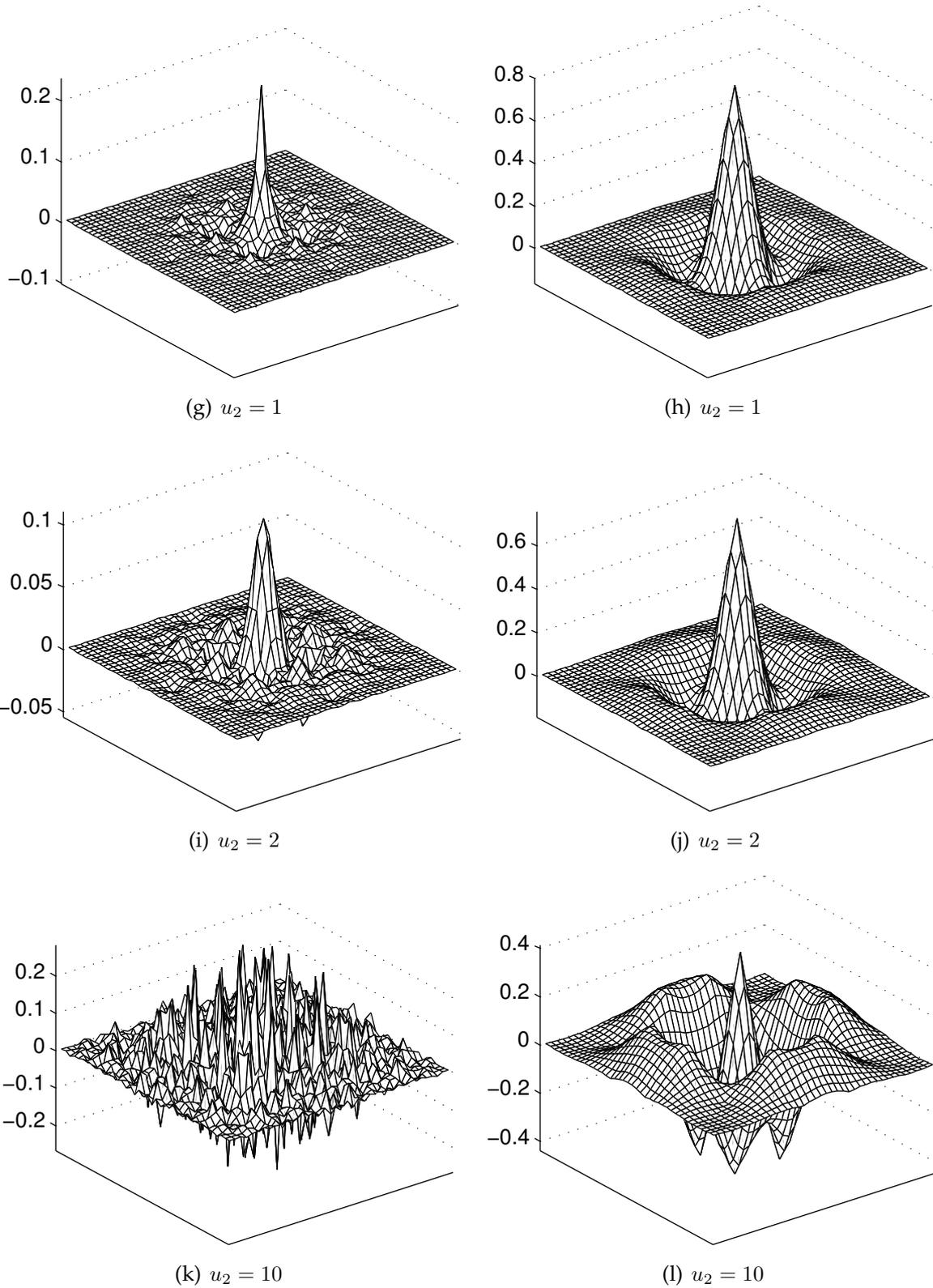


Figure 4.19: *Continuation*: Analysis (left column) and synthesis (right column) limit wavelet functions obtained by using prediction filter P_4 and weighted update filter $U_2 + u_2(U_4 - U_2)$.

4.3 Importance of Lower Prediction Sections

The prediction structure from figure 4.9 is split into four prediction sections allowing a maximum of three free filter parameters. In case of the maximum number of free parameters, p_2 , p_3 and p_4 can be changed to tune the desired filter properties, while p_1 is being set to 1 to guarantee two dual vanishing moments no matter what values the other three parameters may take. The magnitude frequency responses of all prediction sections are shown in figure 4.20 (compare with figure 3.11). The second, third and fourth filter are combinations of prediction filters and as such they have band-pass properties.

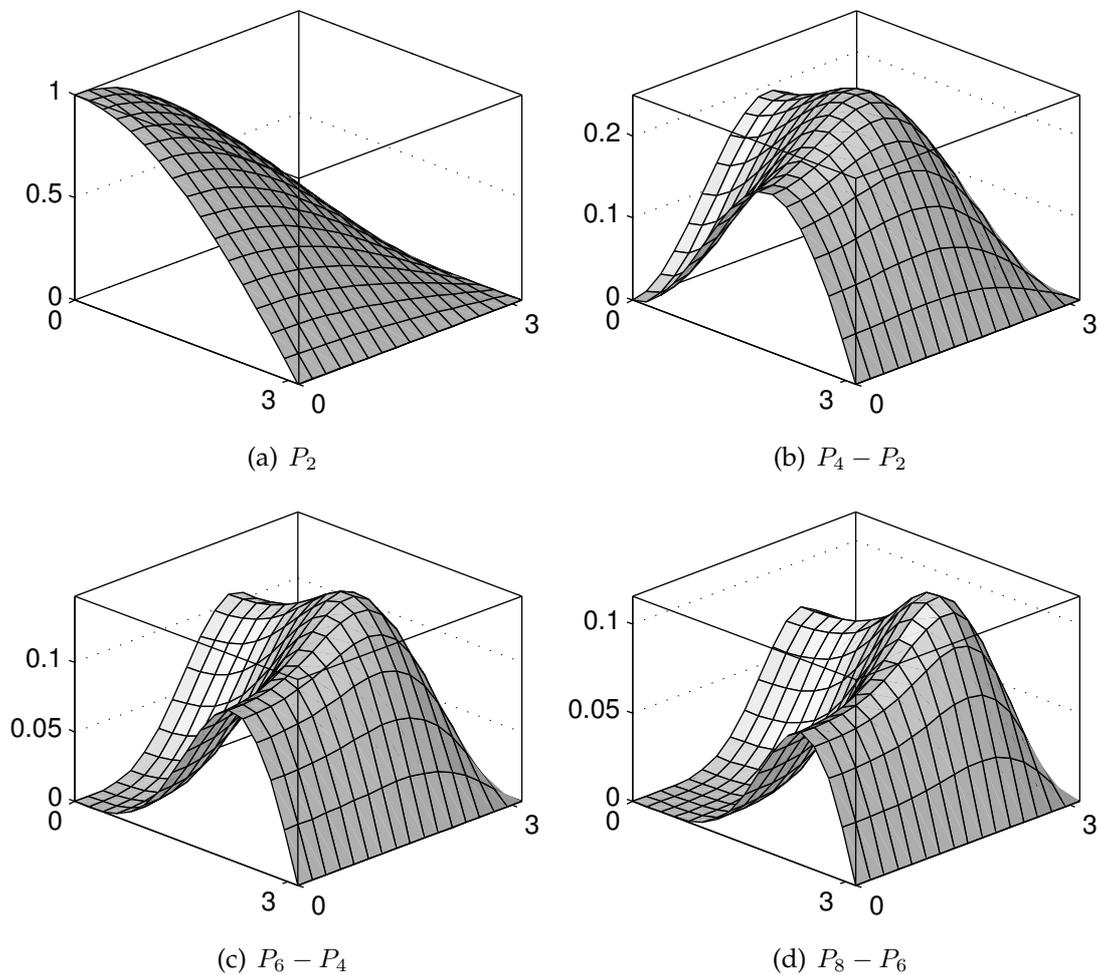


Figure 4.20: Magnitude frequency responses for all prediction sections.

The example of decomposition of a real image is given in figure 4.22. The first section contributes mostly to the overall prediction. The contribution of the second section (filter $P_4 - P_2$) is much smaller but still important for fine-tuning of the filter properties.

The contribution of the third and fourth sections to the overall prediction is almost insignificant. An example is shown in figure 4.22 where the energy of the signal from the second prediction section is 0.017% as compared to the energy of the signal from the first prediction section, while the third and fourth prediction sections contain 0.0026% and 0.0009% of the amount of energy of the first section.

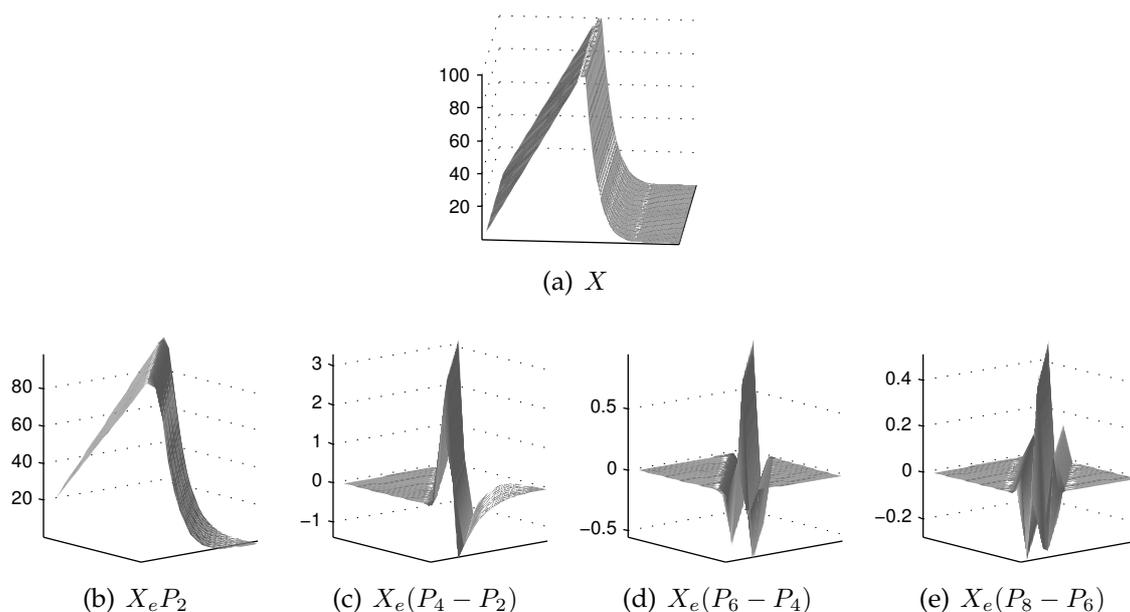


Figure 4.21: The input signal (a) is composed out of two polynomials of 1st and 8th order. Central parts of the outputs from all prediction sections are shown in (b), (c), (d) and (e). Higher order prediction is paid with wider transition effects.

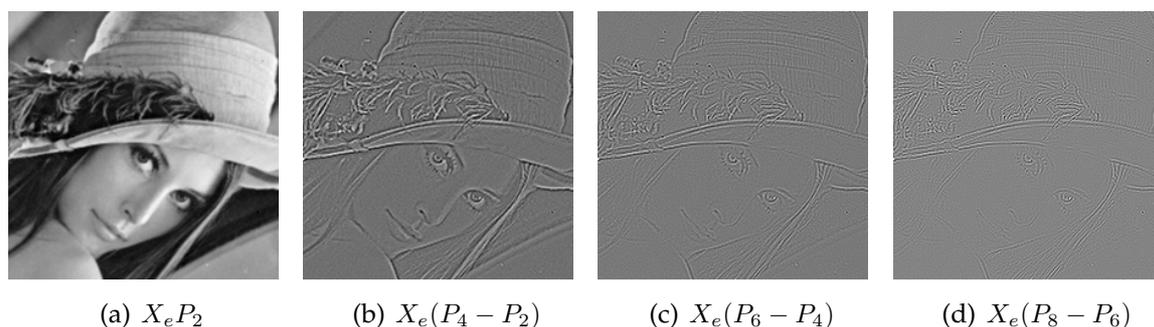


Figure 4.22: Outputs from all the prediction sections for image Lena (magnified central parts are shown).

Other than giving insignificant contribution to the overall prediction, the higher order prediction sections introduce higher boundary and transition effects. One example of this behavior is shown in figure 4.21. The input signal (figure 4.21(a)) is composed of

two polynomial sequences of different orders. As the prediction order increases, the transition area (caused by the boundary of the two polynomials) in the output of the prediction section widens.

Considering the effects mentioned above, it is obvious that a selection of an optimal number of prediction sections is a matter of compromise. Higher prediction sections lead to the further minimizing prediction error but they also introduce wider transition effects. In this case, an optimal solution would be only two prediction sections used, one of them being fixed, and the second one being variable. The first section is a second-order predictor P_2 (multiplied by $p_1 = 1$). The second section contains filter $P_4 - P_2$ whose output is multiplied by parameter p_2 in order to tune filter properties. Similar reasoning holds for the update step. Therefore, an optimal filter bank structure based on the above conclusions has two sections in the predict step and two sections in the update step, each of the predict and update steps having one free parameter. Such a structure is shown in figure 4.23.

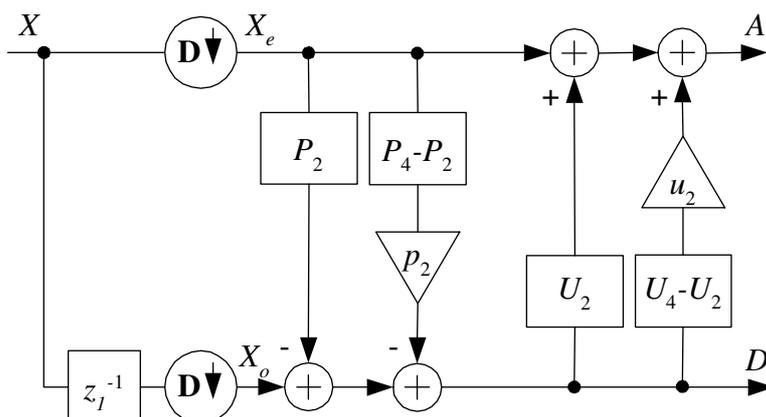


Figure 4.23: Filter bank with one free parameter in the predict stage and one free parameter in the update stage.

Chapter 5

Proposed Adaptive Structure

The purpose of the filter banks proposed in this thesis is to provide an adaptive wavelet decomposition of an input image that will be superior to the fixed decomposition scheme. The proposed filter bank will change its properties for every pixel of the analyzed image in every decomposition level. There are various adaptation schemes based on the minimal squared error criterion presented in this chapter.

5.1 Adaptive Filter Bank Structure

In chapter 4, the influence of filter parameters' variations on overall filter bank properties was analyzed. Those were static analyses, considering that the given values of the filter parameters are the same across the whole image. In this chapter, we deal with a filter bank that can change its parameters according to the local image properties. Therefore, the properties of this filter bank can change in respect of every pixel in the image. In general, for such space-varying filter banks analytical solutions are not known. In this thesis we will assume that filter parameters are varying slowly enough so that the static analysis from chapter 4 can be applicable.

The construction of the adaptive analysis filter bank starts with the predict step shown in figure 4.9. Firstly, the n filter parameters p_1 to p_n are fixed to 1 to provide the desired number of zero moments, while the remaining parameters are free to change according to the image properties. Those free filter parameters will be changed for every pixel of the downsampled image in order to minimize the prediction error measure. Similar procedure is followed by the update step resulting in the general adaptive filter bank structure shown in figure 5.1.

For the sake of simplicity of the adaptation algorithm, and because of the fact that the higher prediction sections introduce bigger transition artifacts while having insignificant contribution to the overall prediction (see section 4.3), the structure shown in

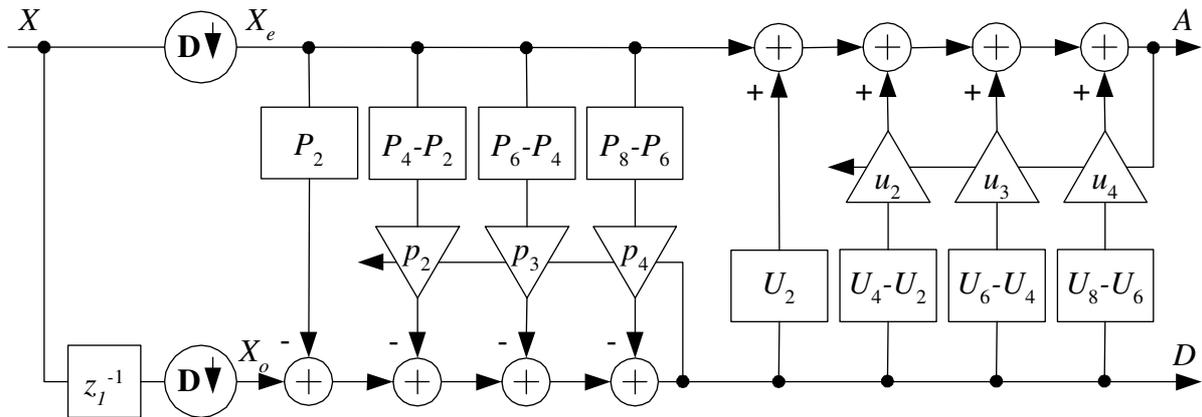


Figure 5.1: Adaptive filter bank with three free parameters in the predict stage and three free parameters in the update stage.

figure 5.2 will be preferred. This structure has only two sections in the predict step and only two sections in the update step. The predict step has one free parameter that will be varied in order to adjust the properties of the output detail signal. The update step has also only one free parameter that is adapted based on the properties of the average signal. This filter bank structure and it's modifications will be discussed in the sequel.

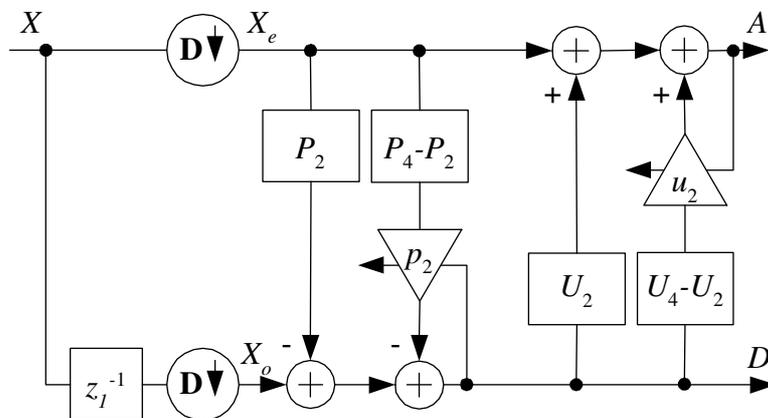


Figure 5.2: Adaptive filter bank with one free parameter in the predict stage and one free parameter in the update stage.

5.2 Adaptation Criteria

Regarding the shape of the adaptation area, the adaptation algorithms used in this thesis can be divided in two distinct groups:

an additional parameter which gives different weights to prediction errors in different positions of the adaptation area. Since detail coefficients are obtained as an error of predicting the even phase samples from the odd phase samples, the minimization of the squared prediction error actually leads to the minimization of the energy of the detail coefficients on a given adaptation area.

The equation similar to 5.1 stands for the squared prediction error calculated on a true 2-D adaptation area:

$$\xi[n_1, n_2] = \sum_{k_1=n_1-N}^{n_1+N} \sum_{k_2=n_2-N}^{n_2+N} w[k_1, k_2] |d[k_1, k_2]|^2. \quad (5.2)$$

For the sake of simplicity, the prediction error is calculated on a square window containing pixels that symmetrically surround the pixel the adaptation is being performed for. Typical values for N are 0, 1, and 2 giving adaptation areas of 1, 3×3 , and 5×5 pixels in size.

The consequence of minimizing the prediction error on an area is the change of one fundamental predictor property: instead of staying interpolating, predictor becomes approximating. It means that the values of the predicted samples from the odd phase are not the ones that would be obtained by the interpolation of a given order from even samples, but rather their variations that minimize the prediction error.

5.2.1 Adapted Filter Parameters and Perfect Reconstruction

The lifting scheme automatically provides a perfect reconstruction property if the filters on both the analysis and synthesis sides are the same. Therefore, in order to obtain perfect reconstruction of the analyzed image, on the reconstruction side there are not only the detail and approximation coefficients required, but also the values of the filter parameters corresponding to each detail coefficient. To obtain an additional information about filter parameters, there are two possibilities available:

Transfer the information about the filter parameters from the analysis side to the reconstruction side, or

Reconstruct the adapted values of the filter parameters on the reconstruction side.

The latter possibility seems appealing because it saves bandwidth/memory resources required for the transfer of the information about the filter parameters on the reconstruction side, yet imposing some severe constraints at the same time. In order to be self-reproducible on the reconstruction side, the adaptation algorithm has to be strictly causal. It means that the filter parameters have to be computed based on a set of points

"before" the point adaptation is calculated for. This makes the adaptation area strictly asymmetric, which is generally considered a bad choice in image processing. Likewise, such a filter bank is very sensitive to errors in the transmission of wavelet coefficients. The reason is that an error in transmitting one wavelet coefficient affects all further reconstructions of filter parameters and the original image samples. For these reasons, the transmission of filter parameters rather than their reconstruction will be used in this thesis.

One Part of the Information is Contained in the Adapted Filter Parameters

After the adaptation, the energy of the detail coefficients is minimized. The values of the detail coefficients will be more grouped around zero. Since their entropy became smaller, they will be coded more effectively. One part of the information that originally should have been contained in the detail coefficients was stored in the filter parameters p_1 , p_2 , p_3 and p_4 . The information that originally belongs to the detail coefficients is split among the detail coefficients and the (adapted) filter parameters. As already mentioned, in order to make reconstruction possible, the filter parameters will also have to be coded and transmitted to the reconstruction side. If the adaptation area is large enough, the filter parameters will vary relatively slowly and should be coded very effectively, thus adding insignificant burden to the overall coding performance. As an adaptation area decreases, the spatial variations of the filter parameters become faster and more information content enters the filter parameters while minimizing the entropy of the detail coefficients. The extreme case occurs when the adaptation area becomes only one pixel wide. In that case, the detail coefficients will be completely set to zero and all the information will be contained in the filter parameters. The optimal choice of the adaptation area is somewhere in between and depends on the spatial properties of the given image.

5.3 Least Squares Adaptation Methods

In this section, we introduce the notation that will simplify the subsequent mathematical relations. As shown in figure 5.4, y is the signal obtained by subtracting the output of the fixed prediction section from the odd phase of the input signal. The signal at the input of the variable multiplier p_2 is marked as x . The signal at the output of p_2 is marked as \hat{y} since it represents the prediction of the y signal. The prediction error can

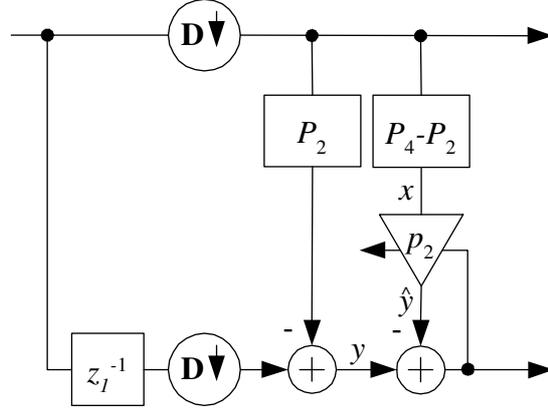


Figure 5.4: The adaptive predict structure from figure 5.2. The x and y signals are inputs to the least squares adaptation algorithm. The results of the adaptation algorithm are the p_2 parameter and the \hat{y} signal.

be expressed as:

$$\begin{aligned}\varepsilon[n] &= y[n] - \hat{y}[n] \\ \varepsilon[n] &= y[n] - X_N^T[n]W_N[n],\end{aligned}\tag{5.3}$$

where $X_N[n]$ is the input vector for the position n and $W_N[n]$ is the filter parameters' vector such that $X_N^T[n]W_N[n]$ is a convolution of the input signal X_N with the filter's impulse response for the position n . For the case of the prediction structure shown in figure 5.4, the relation 5.3 becomes

$$\varepsilon[n] = y[n] - x[n]p_2[n].\tag{5.4}$$

The parameters that minimize the sum of squares of the prediction error can be obtained as:

$$W_N^* = (\mathbf{X}_{KN}^T \mathbf{A}_{KK} \mathbf{X}_{KN})^{-1} \mathbf{X}_{KN}^T \mathbf{A}_{KK} \mathbf{Y}_N,\tag{5.5}$$

where \mathbf{X}_{KN} is a matrix that contains the delayed copies of the input samples:

$$\mathbf{X}_{KN} = [X_N[0], X_N[1], \dots, X_N[k]]^T.\tag{5.6}$$

For the filter bank structure with only one adaptable parameter shown in the figure 5.2, W_N^* is a scalar representing the p_2 parameter. In that case \mathbf{X}_{KN} will contain only one column. Vector \mathbf{Y}_N contains the output samples:

$$\mathbf{Y}_N = [y[0], y[1], \dots, y[k]]^T.\tag{5.7}$$

The \mathbf{A}_{KK} matrix is the weight matrix. In case \mathbf{A}_{KK} is the identity matrix, the relation 5.5 becomes:

$$W_N^* = (\mathbf{X}_{KN}^T \mathbf{X}_{KN})^{-1} \mathbf{X}_{KN}^T \mathbf{Y}_N. \quad (5.8)$$

Figure 5.5(a) shows a synthetic image that will be used for testing the adaptation algorithms. It is composed of two spatial sine waves of different frequencies and orientations. When processed with a filter bank with fixed P_2 , the detail image with pretty large values is obtained (see figures 5.5(b) and 5.5(e)). When P_4 is used, the resulting detail coefficients (see figures 5.5(c) and 5.5(f)) are smaller yet still of significant values. It is our goal to completely cancel the two sine waves in the detail image. The adapta-

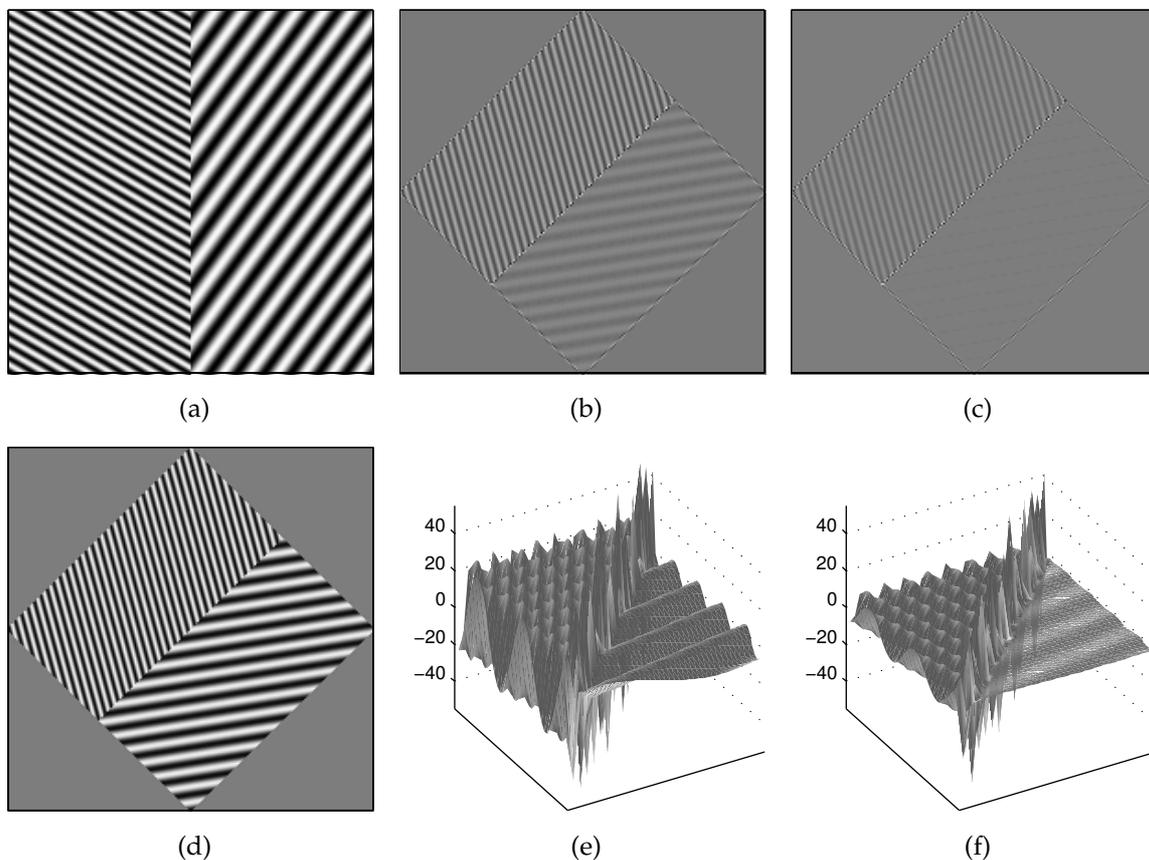


Figure 5.5: Input to the wavelet filter bank is a synthetic image (a) composed of two spatial sine waves of different frequencies ($\omega_1 = 2\pi/6$, $\omega_2 = 2\pi/12$), different orientations ($\varphi_1 = -\pi/3$, $\varphi_2 = \pi/5$) and amplitude 100. Firstly fixed P_2 and U_2 have been used resulting in approximation image (d) and detail image (b) whose central part values are shown in (e). Fixed filters P_4 and U_2 were used next, resulting in smaller yet still significant detail coefficients: (c) and (f).

tion algorithm should find the optimal p_2 parameters which would position the zeros

of the resulting high-pass filter on the frequencies of the two sine waves. As shown in figure 4.13, when the p_2 parameter grows, the zero ditch widens and tends to cancel the higher frequencies. Therefore, we expect our adaptation algorithm to set the p_2 parameter to one value for the first part of the image containing the first sine wave, and to another value in the second part of the image containing the second sine wave. Since the first sine wave has a higher frequency than the second one, the value of the p_2 parameter should be higher for the first part of the image.

5.3.1 1-D LSW

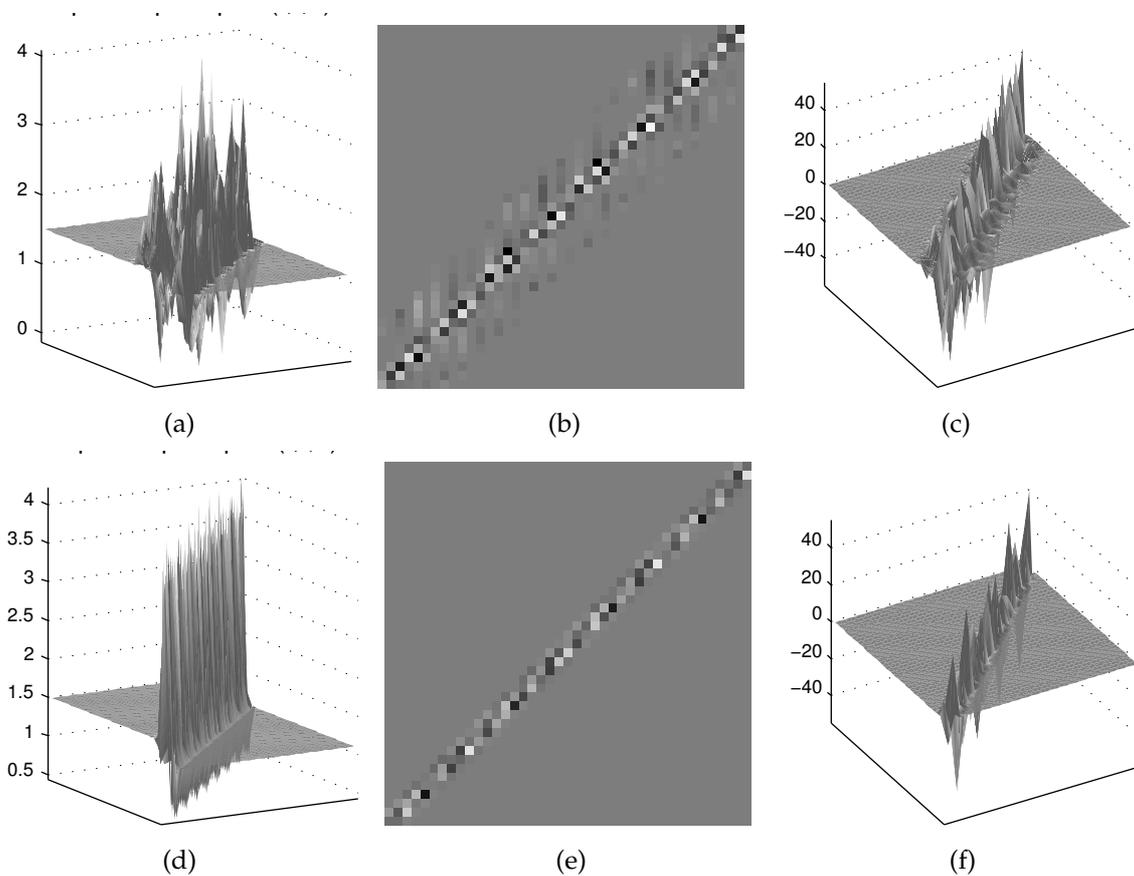


Figure 5.6: Results of the p_2 parameter adaptation obtained by using 1-D LSW on a window of 6 samples. First column represents values of p_2 . Second and third column represent values of adapted detail coefficients. First row represents 1-D LSW with vertical collection of pixels (adaptation line intersects the boundary of the two sine waves). Second row represents 1-D LSW with diagonal collection of pixels (adaptation line is parallel with the boundary). There are central parts of the images shown. Detail coefficients are completely set to zero except those on the boundary of the two sine waves.

The one-dimensional least squares method calculates the least squares solution on a given block of data as given in equation 5.8. Firstly, the image elements are mapped using vertical or horizontal collection of pixels into a one-dimensional data sequence (see figure 5.3). There were windows with 3, 6, 9 or 12 pixels used in the calculations. Figure 5.6 shows the results of the 1-D LSW algorithm applied to the test image from figure 5.5. As expected, the adaptation algorithm found the two values of p_2 that completely cancel the two sine waves. The adaptation algorithm decided to set $p_2 = 1.4981$ for the first sine wave and $p_2 = 1.1065$ for the second sine wave having a smaller frequency. The detail coefficients are completely set to zero in the areas of the two sine waves and there is a transition area in between. Figure 5.6(b) shows the central part of the detail coefficients after using 1-D LSW with the vertical collection of pixels. There are the directional artifacts near the boundary clearly visible. When the diagonally oriented adaptation is used, the transition area is minimal as shown in figure 5.6(e).

5.3.2 1-D RLS

The recursive least squares algorithm [Haykin 86] uses the values obtained in the previous step to calculate the filter parameters for the current step. This method is numerically more efficient. The total squared prediction error is calculated as

$$\xi[n] = \sum_{i=0}^n \alpha^{k-i} \varepsilon^2[n], \quad (5.9)$$

where α is called the *forgetting factor* with $0 < \alpha < 1$. The weight matrix from equation 5.5 equals

$$\mathbf{A}_{KK} = \begin{bmatrix} \alpha^k & 0 & \dots & 0 \\ 0 & \alpha^{k-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}. \quad (5.10)$$

The figure 5.7 shows results obtained by using the 1-D recursive least squares algorithm (RLS) with the vertical collection of pixels. This algorithm bases the calculation of the least squares problem on the number of "previous" samples that are taken into account with exponentially decaying weights. The stationary values of p_2 are the same as before, but the transition area is somewhat wider, clearly demonstrating slowly decaying vertical artifacts.

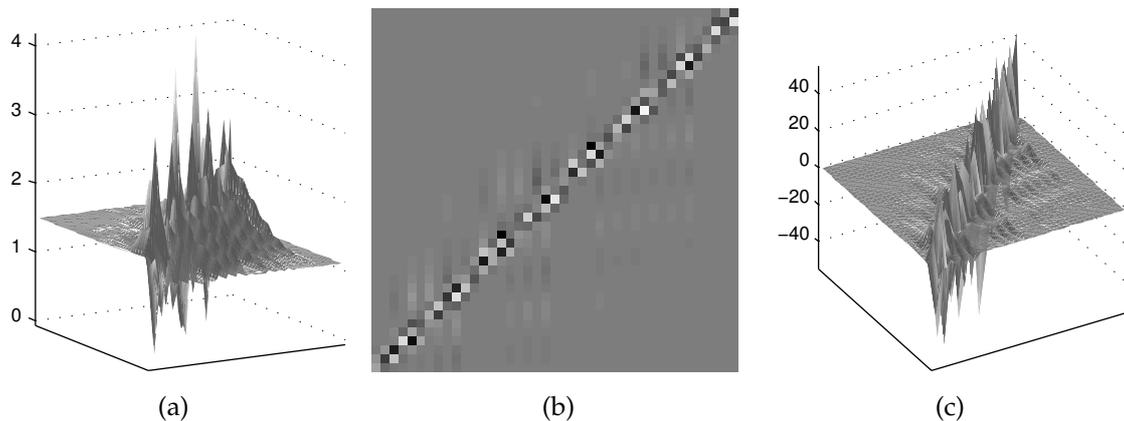


Figure 5.7: Results of the p_2 parameter adaptation obtained by using 1-D RLS algorithm with vertical collection of pixels and forgetting factor $\lambda = 0.82$. There are central parts of adapted p_2 parameters (a) and resulting detail coefficients (b) and (c) shown.

5.3.3 2-D LSW

It is evident that one-dimensional adaptation algorithms cannot give best results for images. True two-dimensional algorithms are a more natural choice. The 2-D LSW adaptation algorithm takes a number of pixels that form a square around a pixel which is being adapted and solves the least squares problem for that window. The results obtained by using 2-D LSW algorithm on a window of 3×3 and 5×5 pixels are shown in figure 5.8. The use of the 2-D LSW reduced the variance of the p_2 parameters around the boundary, which is good. By comparing figures 5.8(a) and 5.8(d), we see that the greater the adaptation window, the smaller the variances of the p_2 parameters.

5.3.4 Robust 2-D LSW

We are close to obtaining a representation of a synthetic image 5.5(a) that will have almost all the detail coefficients set to zero and practically two-valued p_2 parameters. Such a representation is much more compact than the one obtained with fixed filters. In order to further improve the transition behavior of the adaptation algorithm, we will try to make it more robust. The idea is to trace the prevalent image features by excluding those equations that result in a largest quadratic prediction error on a given window. The procedure goes as follows and is repeated iteratively 3 times:

Calculate the value of p_2 that gives the least squares solution for a given window.

Apply that value of p_2 on all points in a given window and calculate the corresponding quadratic prediction errors.

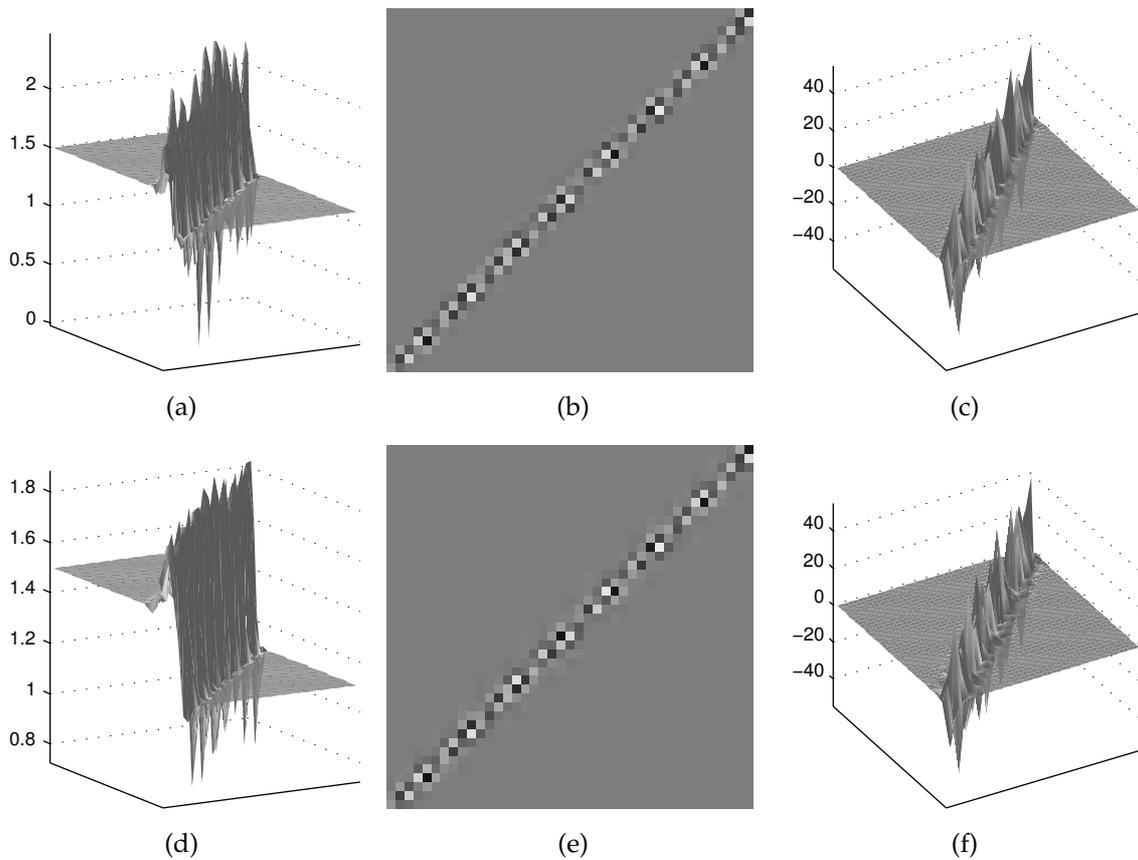


Figure 5.8: Results of the p_2 parameter adaptation obtained by using 2-D LSW. First column represents values of p_2 . Second and third column represent values of adapted detail coefficients. There are central parts of the images shown. First row: window size of 3×3 pixels. Second row: window size of 5×5 pixels.

Discard $N - M$ points that give highest prediction errors and calculate the least squares solution based on the remaining M points.

The results shown in figure 5.9 refer to the two different values of M . The results are almost optimal. The transition from the "left" value to the "right" value of p_2 is almost abrupt.

5.3.5 BLUE

Minimizing the squared value of the detail coefficients seems to be a good choice. But, in the update step, such a minimization would destroy the the DC component of the average coefficients. Of course, we do not want to do that since the average value should remain unaffected. Therefore, it is necessary to use some kind of spectrally corrected criterion which would leave the DC component unaltered.

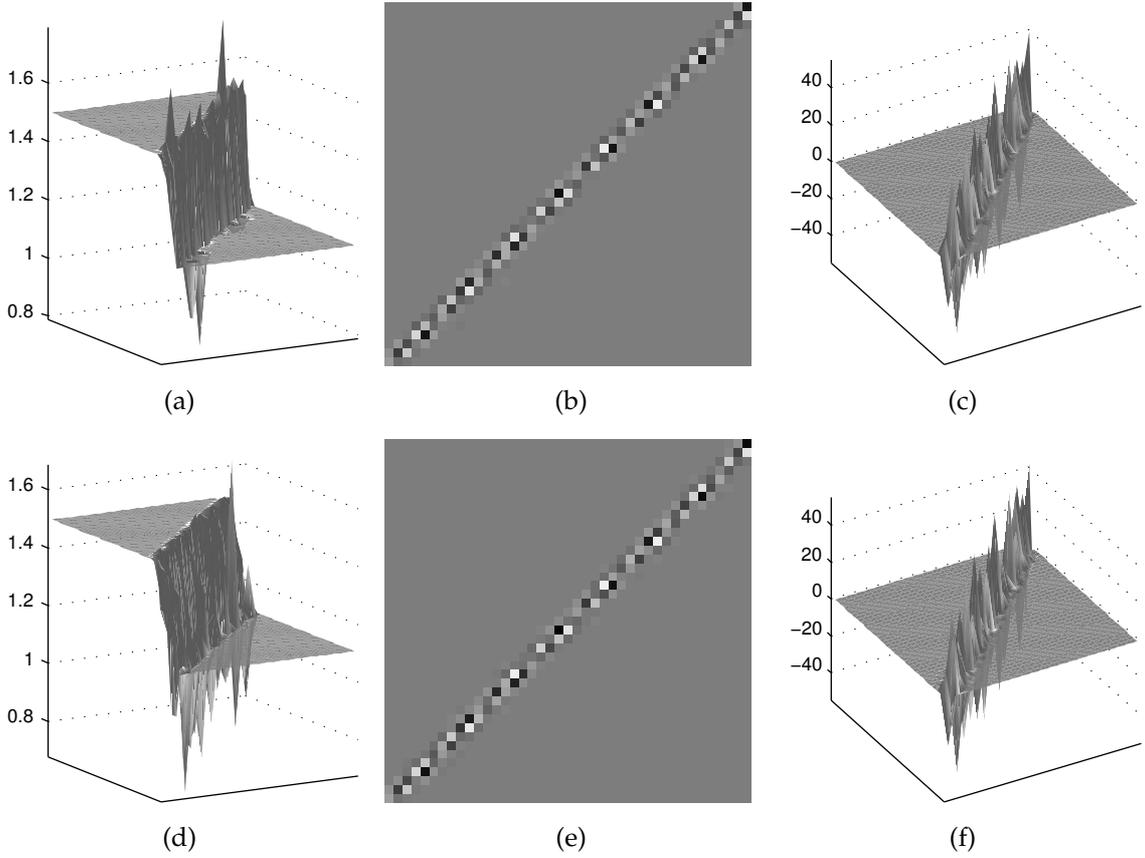


Figure 5.9: Results of the p_2 parameter adaptation obtained by using robust 2-D LSW on a 5×5 window. First column represents values of p_2 . Second and third column represent values of adapted detail coefficients. Central parts of the images are shown. First row: $M = 10$. Second row: $M = 20$. Robust algorithm gives shorter transition area.

Although at first it may not seem so, such a criterion would also serve good in the prediction step. Let us consider an input signal that contains a component at the aliasing frequency. After a polyphase decomposition, that component will become a DC frequency and the prediction step will try to cancel it. We do not want that to happen, so the spectrally corrected adaptation criterion mentioned before will help us preserving the aliasing frequency component from being cancelled from the detail coefficients. It is known that the Best Linear Unbiased Estimate (BLUE) [Haykin 86] is given by the relation:

$$W_N^* = (\mathbf{X}_{KN}^T \mathbf{V}_{KK}^{-1} \mathbf{X}_{KN})^{-1} \mathbf{X}_{KN}^T \mathbf{V}_{KK}^{-1} \mathbf{Y}_N, \quad (5.11)$$

where \mathbf{V}_{KK} is an autocorrelation matrix of the noise signal. Therefore, we will treat the DC signal in the average and detail coefficients as a noise signal. The adaptation will be done solving the equation 5.11 on a rectangular window of 3×3 or 5×5 pixels in

size.

Since the autocorrelation matrix of the DC signal is singular, it is not possible to find it's inverse. A proposal to the solution to this problem is given in [Seršić 99]. It proposes modelling of the DC component $v[n]$ as:

$$v[n] = (1 - \varepsilon)^n, \quad (5.12)$$

where ε tends to go to zero. The final result is a tri-diagonal inverse autocorrelation matrix

$$\mathbf{V}_{KK}^{-1}(\varepsilon) = \begin{bmatrix} 1 & -1 + \varepsilon & 0 & \dots & 0 \\ -1 + \varepsilon & 2 - 2\varepsilon + \varepsilon^2 & -1 + \varepsilon & \dots & 0 \\ 0 & -1 + \varepsilon & 2 - 2\varepsilon + \varepsilon^2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (5.13)$$

with the limit

$$\mathbf{V}_{KK}^{-1} = \lim_{\varepsilon \rightarrow 0} \mathbf{V}_{KK}^{-1}(\varepsilon) = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & \dots & \vdots \\ 0 & -1 & \ddots & \dots & 0 \\ \vdots & \vdots & \vdots & 2 & -1 \\ 0 & \dots & 0 & -1 & 1 \end{bmatrix}. \quad (5.14)$$

One good example that needs spectrally corrected adaptation criterion in the predict branch is shown in figure 5.10. The primal image consists of two parts which contain two spatial sine waves of different frequencies and amplitude 100 (figure 5.10(a)). An image which contains a checkered pattern with values -20 and 20 (figure 5.10(b)) is superimposed to the primal image. This checkered pattern represents an aliasing frequency component in the final image. When the analyzed image is being divided in the two polyphase components using the quincunx sampling scheme, the aliasing components turns into DC components. Figures 5.10(e) and 5.10(g) respectively show that the first polyphase component signal gained a DC value of -20 and the second polyphase signal gained a DC value of 20 .

Figure 5.11(d) shows the obtained p_2 predict filter parameters for the image from 5.10(c). The 2-D LSW adaptation method has been used on a 5×5 window. Because of the DC component in the polyphase signals, filter parameters fluctuate. There are not two stationary values corresponding to the two sine waves like in case of analyzing the signal without the superimposed aliasing component (see figure 5.11(a)). Such fluctuating filter parameters cause big values of detail coefficients being additionally offset for -40 (figures 5.11(f) and 5.11(e)). It is obvious that in this case a plain least squares criterion

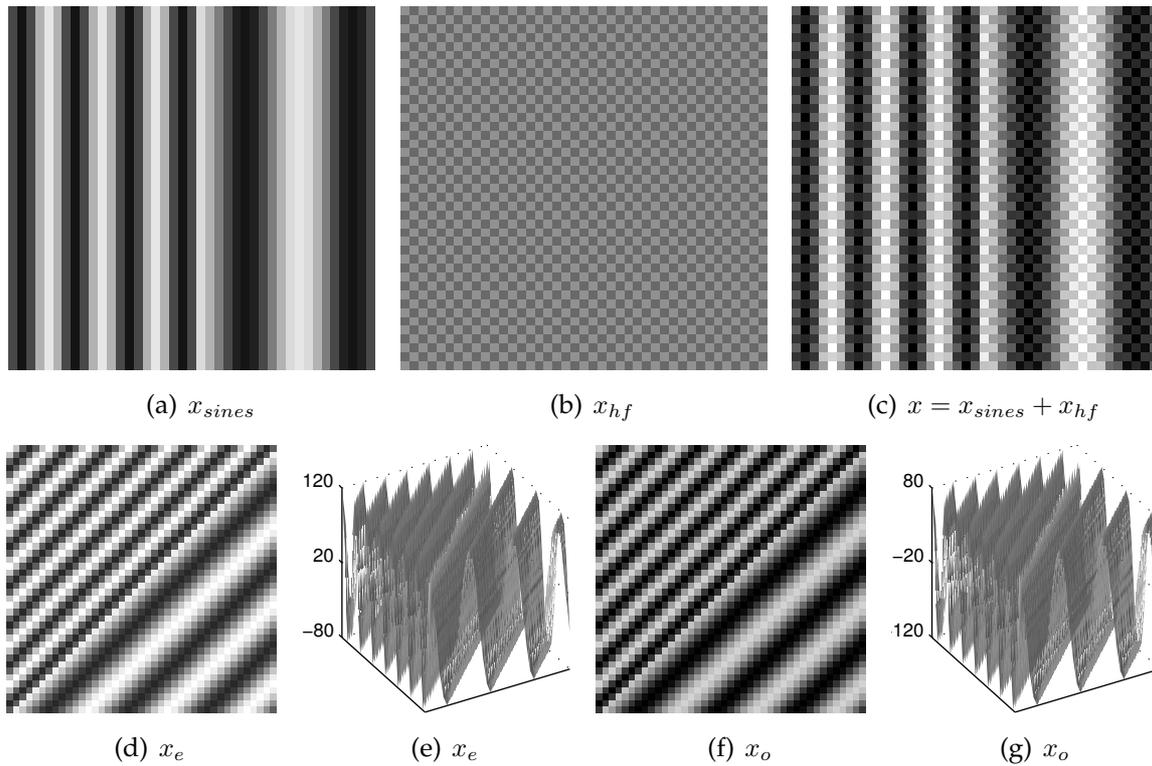


Figure 5.10: Analyzed image (c) is obtained as a superposition of an aliasing frequency component (b) to an image composed of two spatial sine waves with amplitude 100 whose central part is shown in (a). Aliasing frequency component is a checker-board pattern with values of -20 and 20 . (d) and (e) show that the aliasing frequency is transformed into a DC component of value 20 in the even image phase. Figures (f) and (g) show the DC offset of -20 in the odd image phase.

gave results that are much worse than the ones obtained for the fixed prediction filter, without the adaptation of filter parameters.

This problem is surpassed by using the BLUE adaptation method that makes the adaptation criterion ignore the DC component. Figure 5.11(g) shows the obtained filter parameters after using the BLUE method. The resulting filter parameters are two-valued again: $p_2 = 1.3333$ for the left sine wave, $p_2 = 1.0718$ for the right sine wave and some transitory fluctuations in between. These are exactly the same filter parameters as the ones obtained by using a plain 2-D LSW for the two-sine image without the aliasing component (see figure 5.11(a)). As shown in figure 5.11(i) the two sine waves are perfectly cancelled from the detail coefficients (except for the transition area) and the DC component (aliasing frequency of the original image) has been preserved!

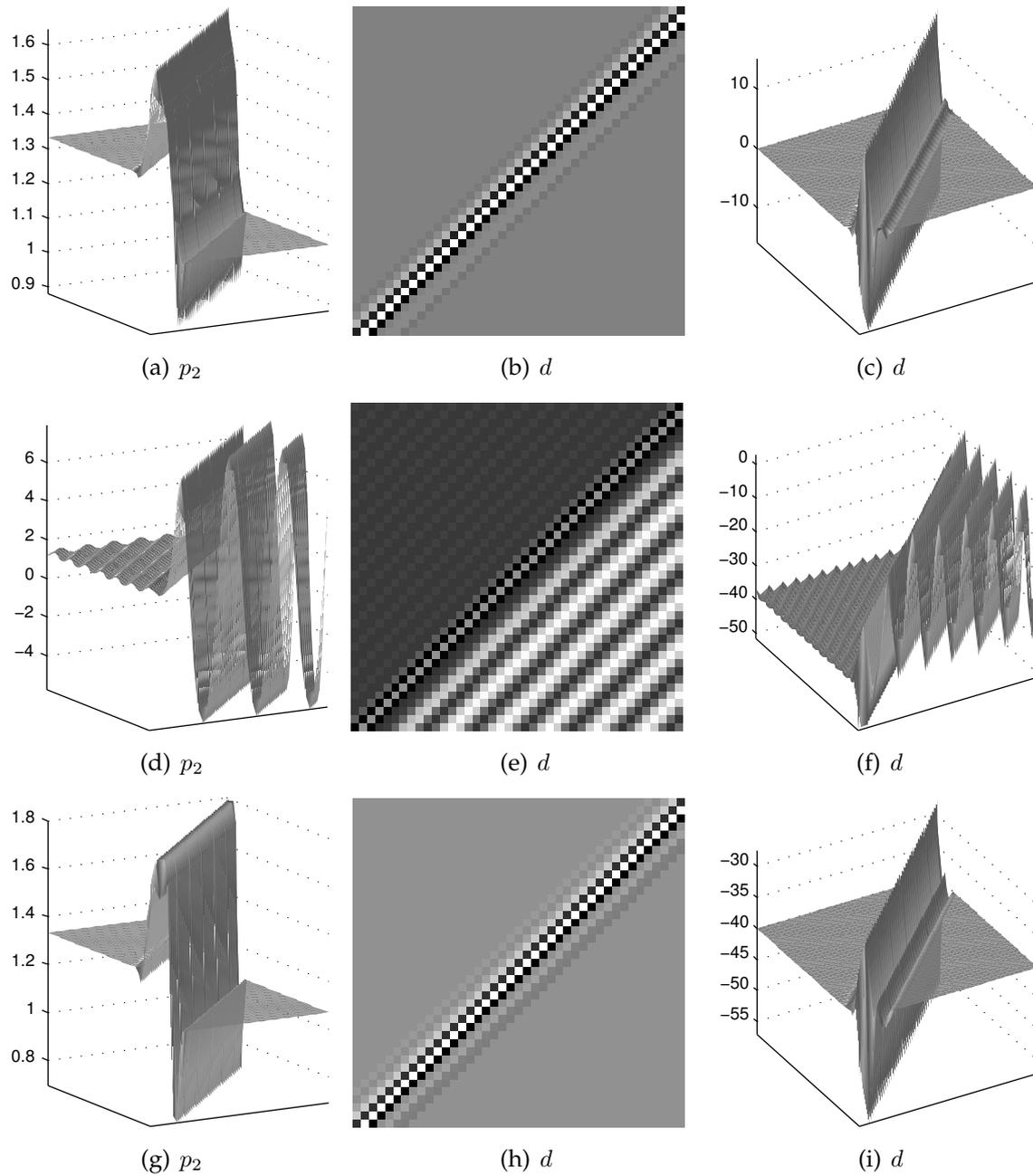


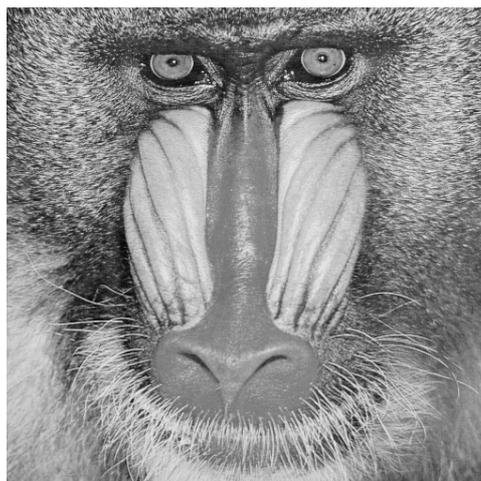
Figure 5.11: First row: plain 2-D LSW on a 5×5 window for a two-sine image from 5.10(a). Second row: plain 2-D LSW for a two-sine image with added checkered pattern (figure 5.10(c)). Third row: BLUE 2-D LSW adaptation for the same image. First column: adapted p_2 filter parameters. Second and third columns: corresponding detail coefficients.

5.4 Results

We will now compare the properties of the above mentioned adaptation algorithms on a set of real-world images. The four typical test images are used (see figure 5.12): Barbara, Mandrill, Lena and Goldhill. Image Barbara is especially interesting since it combines low-frequency (e.g. Barbara's face) and high-frequency areas (striped robe).



(a) Barbara



(b) Mandrill



(c) Lena



(d) Goldhill

Figure 5.12: Four 8-bit images of size 512×512 used for testing the adaptation algorithms.

Figure 5.13 shows detail coefficients obtained for one part of the Barbara image (Barbara's knee) after using fixed prediction filters (P_2 and P_4 respectively) and the adapted prediction filters. The prediction filters were adapted by using the 1-D LSW algorithm with different window sizes. As seen in appropriate histograms, the adaptation turned

detail coefficients' values closer to zero. Therefore, the adapted detail coefficients are expected to be coded more efficiently by using an entropy based coder than the detail coefficients that are obtained with the fixed prediction filters.

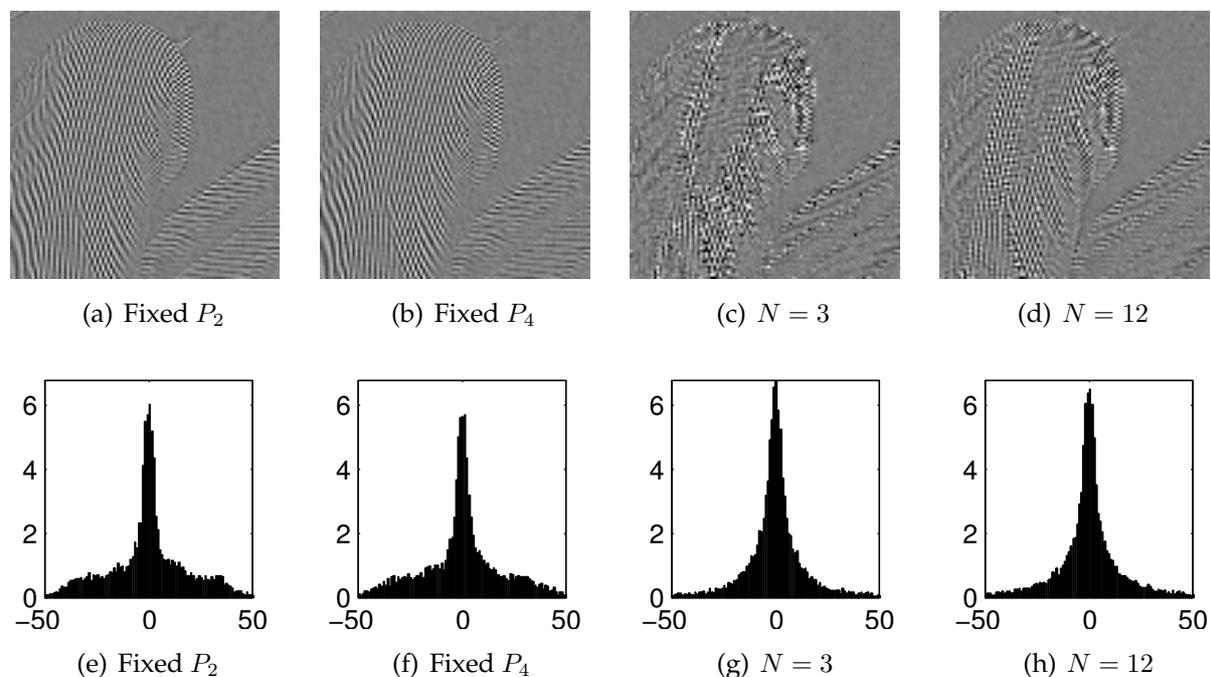


Figure 5.13: Detail coefficients (and their histograms) obtained for one part of image Barbara (Barbara's knee) using fixed predict filters and predict filters adapted with 1D-LSW algorithm of different window lengths.

This adaptation was performed along a diagonal line. Figure 5.14 shows the filter parameters p_2 obtained for different window sizes (N equals 3, 6, 9 and 12 pixels). The diagonal artifacts are clearly visible. The range of filter parameters' values is pretty wide, it also covers the values of p_2 for which the convergence and regularity problems are inevitable. It is worth noting that as adaptation window increases, the range of filter parameters tightens.

As expected, the 2-D adaptation methods give better results. When comparing 1-D LSW and 2-D LSW computed on the same number of pixels (1-D LSW with $N = 9$ in figure 5.15(b) and 2-D LSW with $N = 3 \times 3$ in figure 5.15(c)), detail coefficients are more turned to zero when using the 2-D algorithm. Additionally, filter parameters tend to be more correlated and slowly varying in all directions. As the window size increases, the smoothness of p_2 parameters increases as well. At the same time, the value range of p_2 narrows around zero, bringing most filter parameters in the range of well-behaved values.

Figure 5.17 shows the p_2 predict parameters for central parts of different test images.

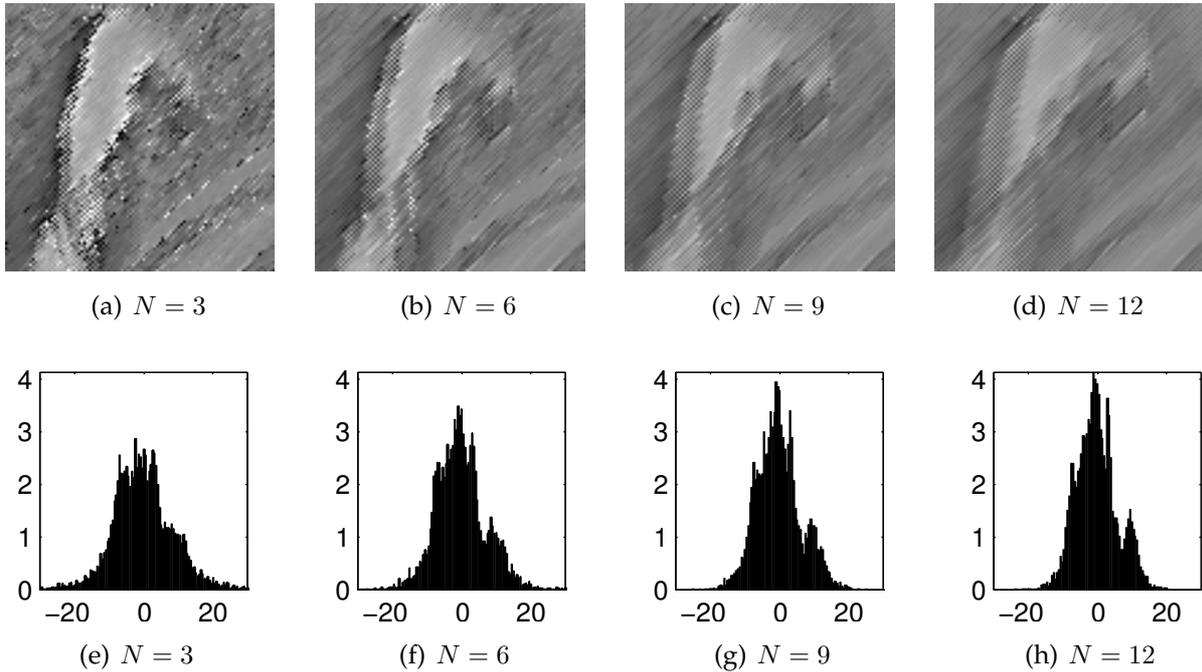


Figure 5.14: Adapted p_2 parameters (and their histograms) obtained for one part of image Barbara (Barbara's knee) using fixed predict filters and predict filters adapted with 1D-LSW algorithm of different window lengths.

The adaptation was done by using 2-D LSW algorithm on a 3×3 and 5×5 pixels wide window. It can be seen that images which are more low-pass give filter parameters that are more concentrated around zero and therefore more favored. For a wider adaptation window the range of p_2 values tightens even more. Figure 5.18 gives a comparison of robust adaptation methods. Adaptation results are shown for a part of image Lena shown in 5.17(g). The adaptation was performed on a 5×5 window, with $M = 8$ and $M = 16$ retained samples. While additionally decreasing the entropy of the detail coefficients (as M decreases), the robust adaptation methods introduce noise in the filter parameters, making them more high-pass. Although robust methods give excellent results for synthetic images, in real-world images they introduce additional variations of the filter parameters.

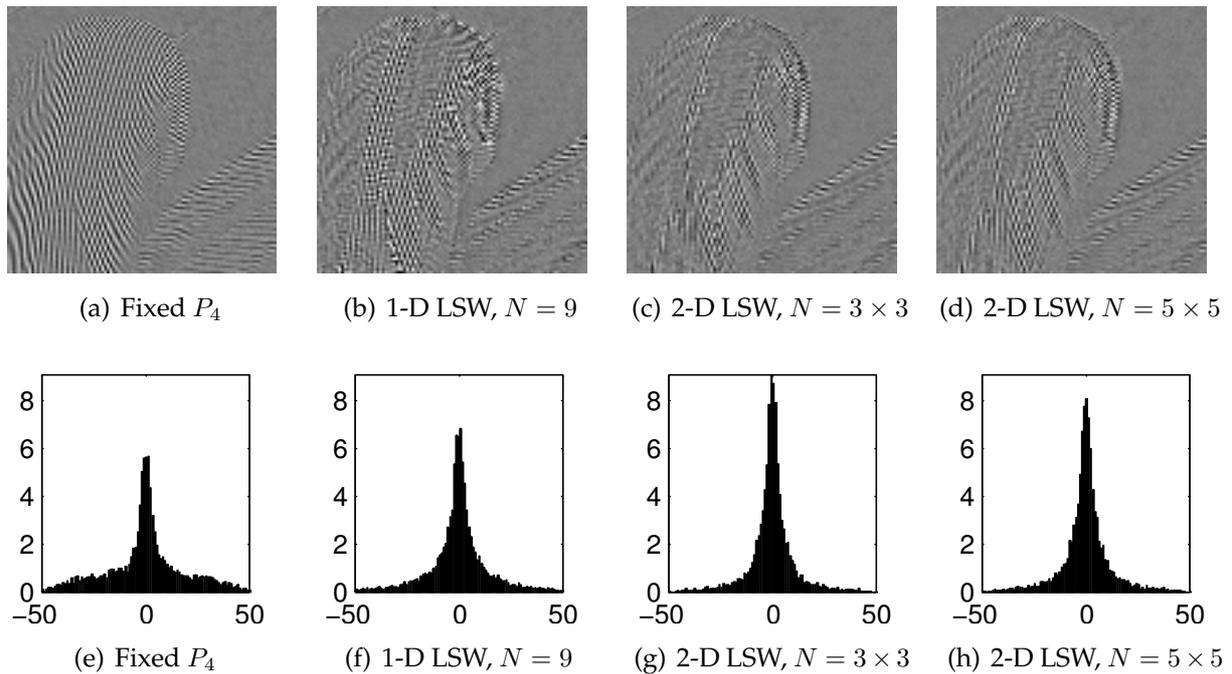


Figure 5.15: Detail coefficients (and corresponding histograms) obtained for one part of image Barbara (Barbara's knee) using fixed prediction filter, 1D-LSW algorithm and 2D-LSW algorithm.

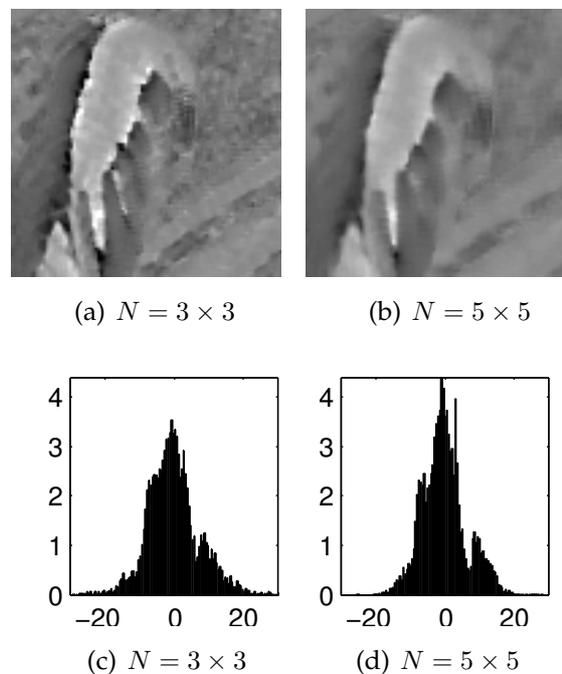


Figure 5.16: Adapted p_2 parameters (and corresponding histograms) obtained for one part of image Barbara (Barbara's knee) using 2D-LSW algorithm with different window sizes.

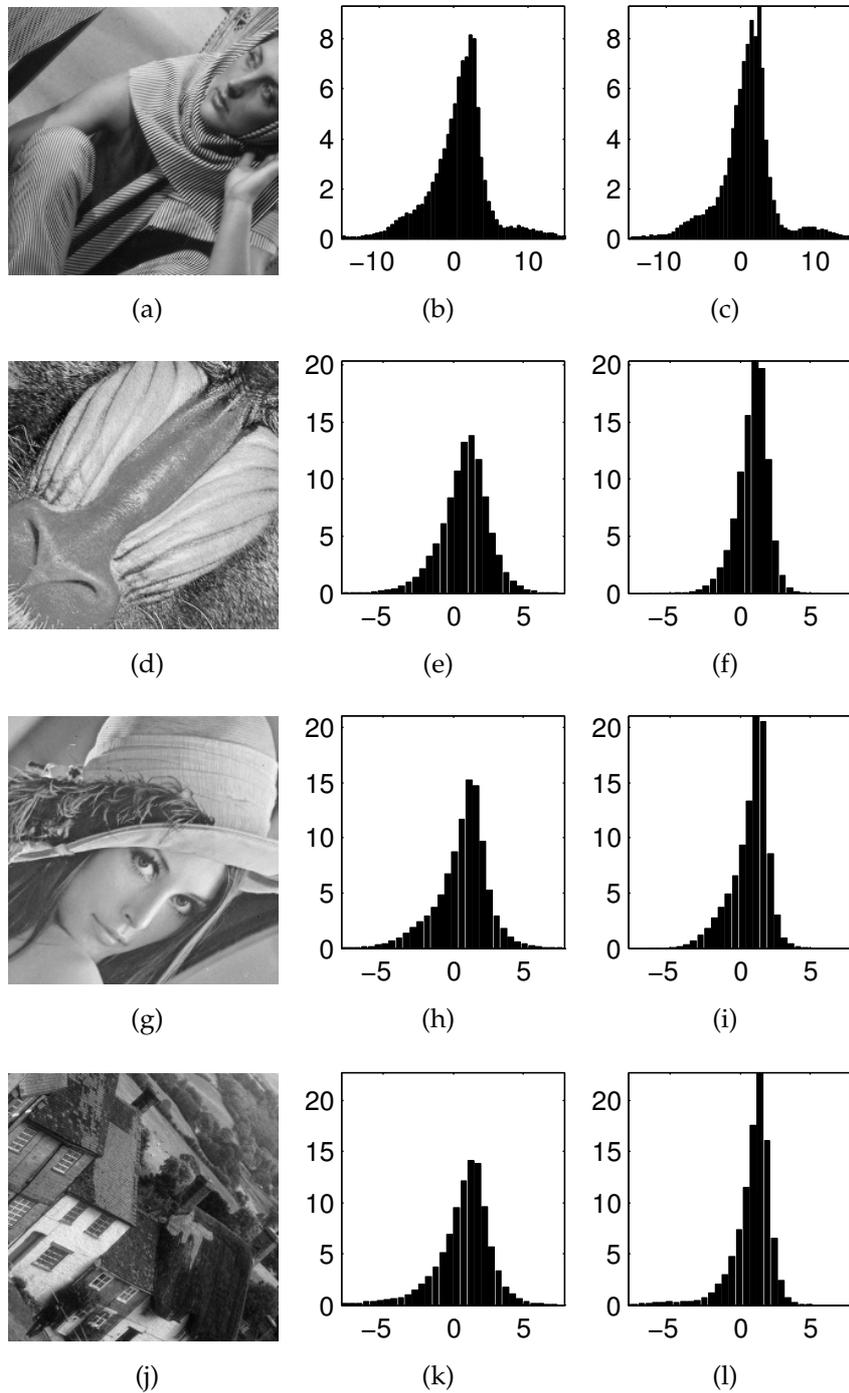


Figure 5.17: Histograms of adapted p_2 parameters for different test images. Second column: 2-D LSW on a 3×3 window. Third column: 2-D LSW on a 5×5 window. First column: average images that correspond to the areas the filter parameters' histograms are shown for.

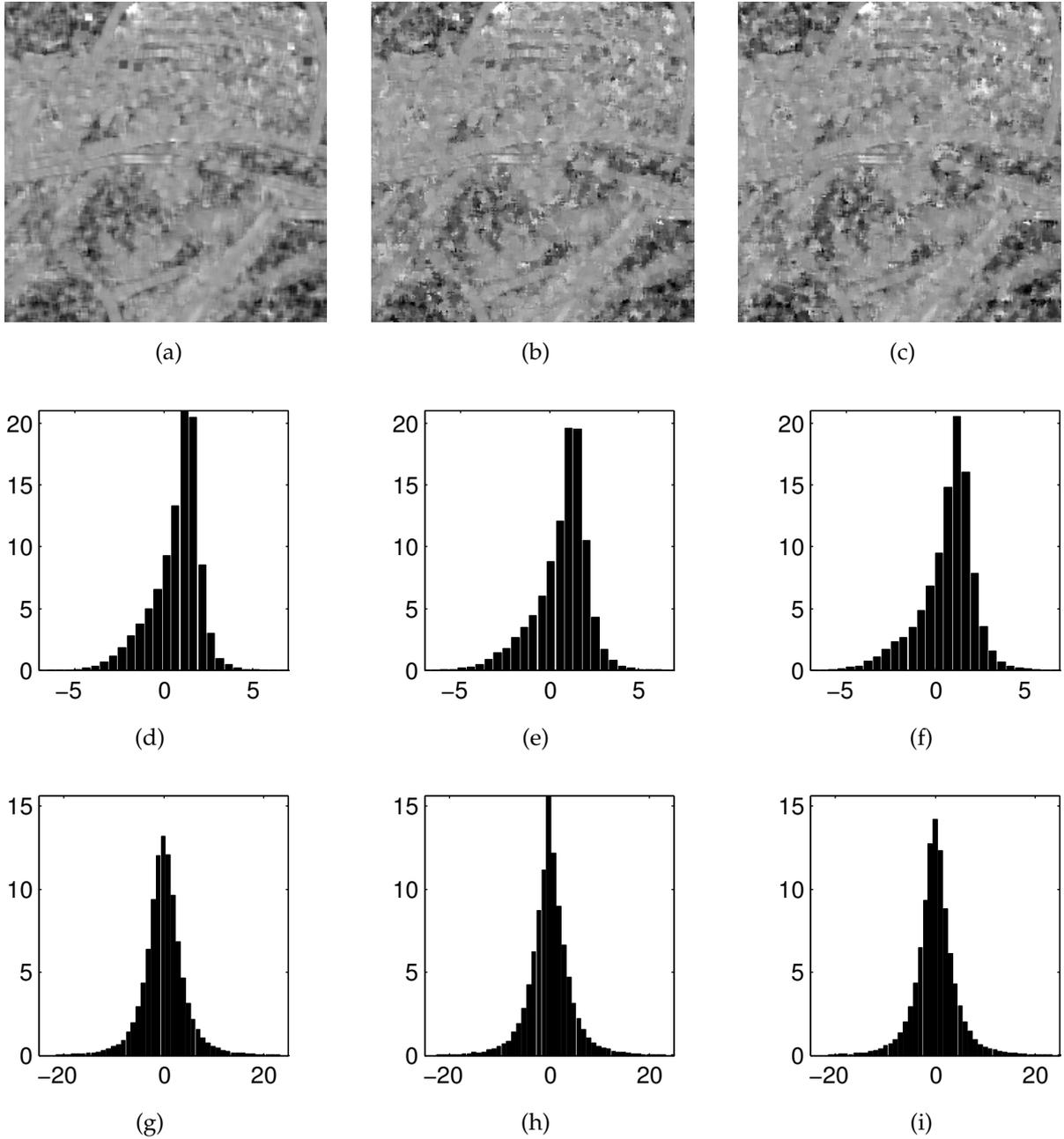


Figure 5.18: Plain 2-D LSW on the 5×5 window (first column), and its robust modifications with $M = 8$ (second column) and $M = 16$ (third column). First row: adapted p_2 parameters for central part of image Lena (see figure 5.17(g)). Second row: histograms of p_2 parameters. Third row: histograms of the resulting detail coefficients.

Entropy of Detail Coefficients and Spectrum Bandwidth of Filter Parameters

In order to compare different adaptation methods, we will use some numerical measures for the obtained detail coefficients and filter parameters. Since the detail coefficients are highly uncorrelated, the entropy-based measure would be appropriate.

On the other hand, we want filter parameters to vary as slowly as possible. Therefore, we provide a spectrum bandwidth measure that tells us how much the filter parameters are low-pass.

$$\omega_{LP} = \sqrt{\frac{\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} (\omega_1^2 + \omega_2^2) |X(e^{j\omega_1}, e^{j\omega_2})|^2}{\int_{-\pi}^{\pi} \int_{-\pi}^{\pi} |X(e^{j\omega_1}, e^{j\omega_2})|^2}} \quad (5.15)$$

Values of these measures for different adaptation algorithms are given in tables 5.1, 5.2 and 5.3.

Adaptation method	Entropy of D	Spectrum bandwidth of p_2
No adaptation		
$p_2 = 0$ (fixed P_2)	6.06	-
$p_2 = 1$ (fixed P_4)	5.98	-
1-D LSW		
$N = 3$	1.89	2.36
$N = 6$	4.42	2.26
$N = 9$	4.87	2.32
$N = 12$	5.15	2.42
2-D LSW		
$N = 3 \times 3$	5.37	0.85
$N = 5 \times 5$	5.36	0.51
Robust 2-D LSW		
$N = 3 \times 3, M = 3$	4.91	1.24
$N = 3 \times 3, M = 6$	4.97	1.19
$N = 5 \times 5, M = 8$	5.24	0.79
$N = 5 \times 5, M = 16$	5.09	0.83
BLUE 2-D LSW		
$N = 3 \times 3$	5.31	0.98
$N = 5 \times 5$	5.38	0.57

Table 5.1: Entropy of the detail coefficients and the spectrum bandwidth of p_2 parameters for different adaptation methods used on the central part of image Barbara.

These tables show that the smaller the adaptation area, the smaller the entropy of the

	Barbara	Mandrill	Lena	Goldhill
Fixed P_2	5.98	5.99	5.32	5.98
1-D LSW, $N = 9$	4.87	5.61	4.70	5.67
2-D LSW, $N = 3 \times 3$	5.37	5.84	5.14	5.85
Robust, $N = 3 \times 3, M = 6$	4.97	5.16	4.94	5.57
BLUE, $N = 3 \times 3$	5.31	5.80	5.19	5.88

Table 5.2: Entropy of the detail coefficients for different adaptation methods calculated on the central parts of different test images.

	Barbara	Mandrill	Lena	Goldhill
Fixed P_2	-	-	-	-
1-D LSW, $N = 9$	2.32	2.19	2.27	2.14
2-D LSW, $N = 3 \times 3$	0.85	1.27	1.50	1.15
Robust, $N = 3 \times 3, M = 6$	1.19	1.58	1.57	1.45
BLUE, $N = 3 \times 3$	0.98	1.41	1.46	1.40

Table 5.3: Spectrum bandwidth of p_2 parameters for different adaptation methods calculated on the central parts of different test images.

detail coefficients. On the other hand, as the adaptation area increases, the spectrum bandwidth of the filter parameters decreases thus making the filter parameters more low-pass.

Adaptation of the Update Filter

In order to improve the resulting approximation image, the update filter section should be adapted. To preserve its average value, the BLUE adaptation method should be used rather than the plain 2-D LSW adaptation algorithm. The BLUE method preserves the DC component of the approximation image while trying to cancel its higher frequency components. Therefore, the approximation image obtained by using the adapted update filter should be more low-pass than the one obtained with the fixed update filter.

The results obtained by using the adapted update filter are shown in figure 5.19. The BLUE method has been used on a window 3×3 pixels wide. For the sake of simplicity, the fixed P_4 predict filter has been used (i.e. p_2 has been fixed to one). As it can be seen in figure 5.19(d) and 5.19(e), approximation image obtained by using the adapted update step became more low-pass. Originally smooth areas (e.g. Barbara's face) became even more smooth. Yet, sharp edges have been degraded and blurred.

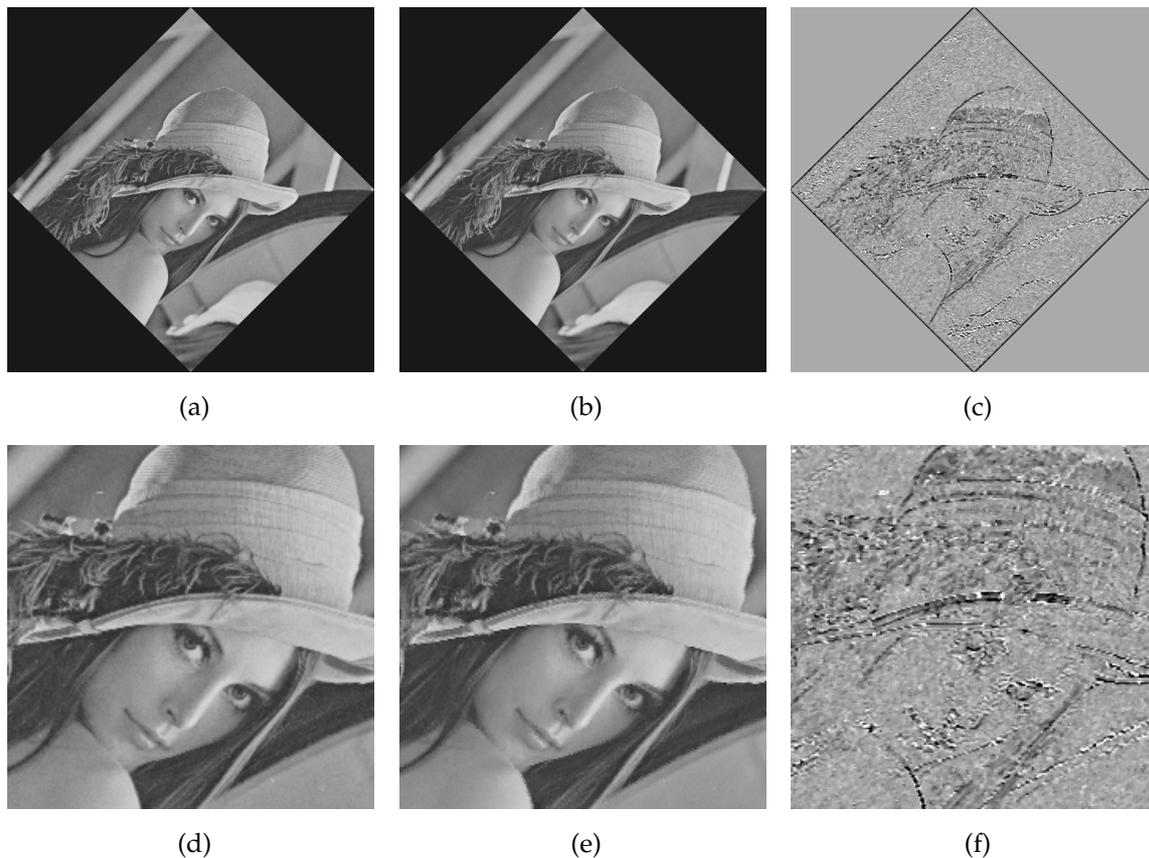


Figure 5.19: Adaptation of the update filter. First column: the average image obtained by using fixed P_4 and U_4 filters. Second column: the average image obtained by using fixed P_4 and the adapted update filter $U_2 + u_2(U_4 - U_2)$. The adaptation was performed by using the BLUE method on a 3×3 window. Third column: corresponding values of the u_2 parameter. Second row: central parts of the images from the first row.

It is important to note that the update step cannot cancel the frequency components of the average image if those frequency components don't exist in the detail image. In case that both the predict and the update step are being adapted, frequency components that are cancelled in the detail coefficients will remain untouched in the average image. Therefore, the gain of adapting the update step will be smaller when the predict step is being adapted simultaneously.

Also, the adaptation algorithm will unsuccessfully try to cancel those frequency components already cancelled in the detail image by using big values of the update filter parameters. To avoid such a behavior the update filter parameters should be confined by the adaptation algorithm inside an acceptable range.

5.4.1 Lossy Image Reconstruction

The advantages of the adapted filter bank schemes over the fixed filter banks can be shown for lossy image reconstruction. The input image is decomposed in a number of decomposition levels. A certain percentage of less significant, i.e. lower wavelet coefficients (values under a certain threshold) is then turned to zero using hard thresholding. The reconstruction is performed using the thresholded wavelet coefficients.

Since the adaptation tends to lower the values of the wavelet coefficients making them contain less information, it is logically expected that such a lossy reconstruction will show better results. There is just one problem here. If the adaptation algorithm chooses very big values of the filter parameters, there will appear the high-valued artifacts in the reconstructed image. The reason is in the divergence of the filters obtained for high positive or negative values of the filter parameters.

An example of lossy reconstruction using fixed filters is shown in figure 5.20. The synthetic image composed of two spatial sine waves has been decomposed in five decomposition levels and then 95% of the detail coefficients have been turned to zero. The artifacts caused by the lossy reconstruction are clearly visible in the output image.

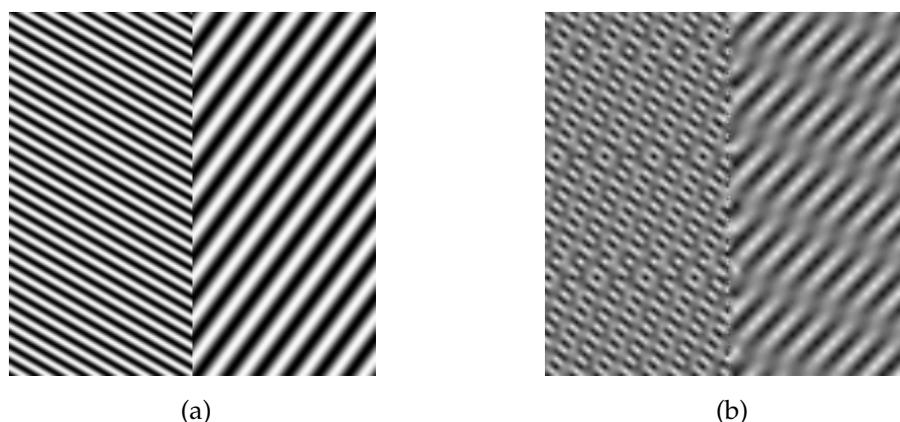


Figure 5.20: (a) Analyzed image. (b) Reconstructed image after 5 decomposition levels using fixed prediction filter P_4 and fixed update filter U_2 . Reconstruction was done after 95% detail coefficients have been turned to zero.

Figure 5.21(f) shows the results of the lossy reconstruction of the same image yet with the adaptation of the predict filter parameters. There has been used the robust adaptation method on a 3×3 window with $M = 6$. There is no visible difference between the original image and the reconstructed image!

Figure 5.22 shows the results for a lossy reconstruction of the image Barbara. The reconstruction was performed after five decomposition levels with 85% of the detail

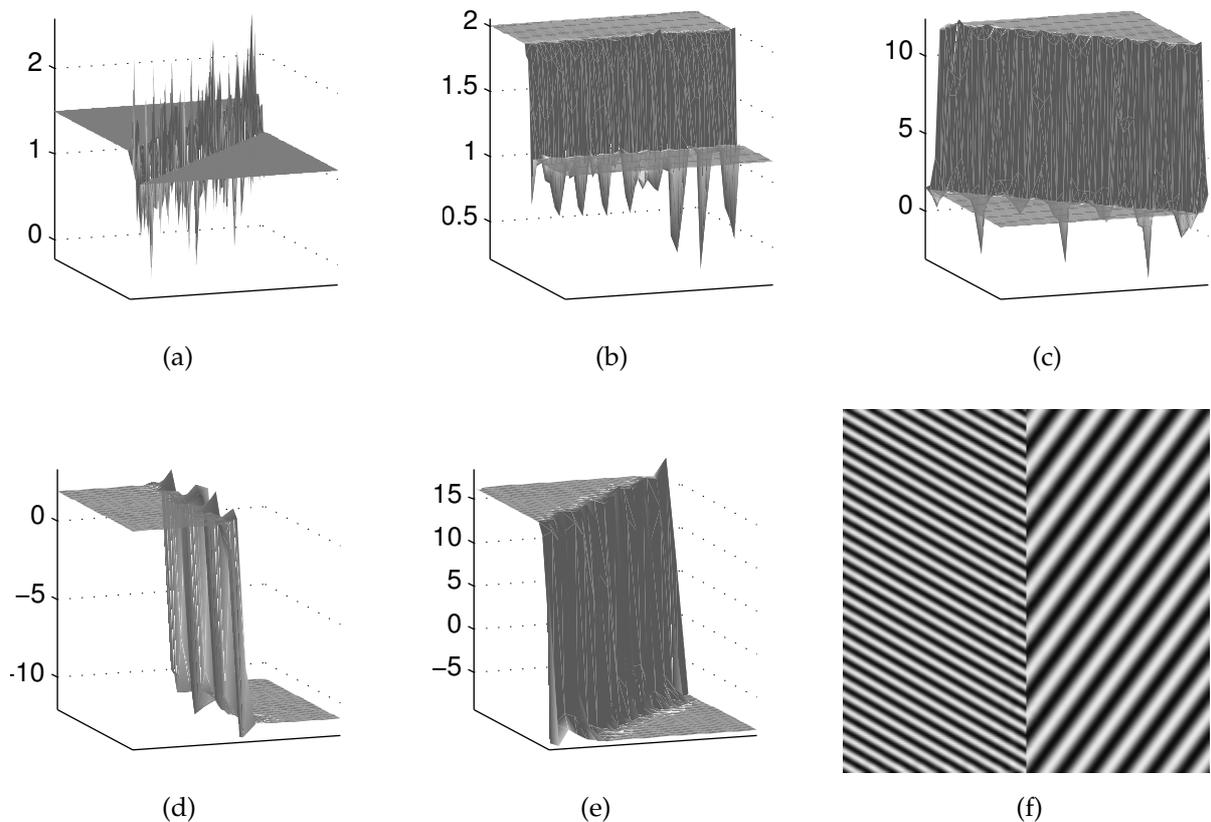


Figure 5.21: Reconstructed image (f) after 5 decomposition levels for adapted predict filter parameter using robust 2-D LSW adaptation. $N = 3 \times 3$ and $M = 6$. Adapted filter parameters for all five decomposition levels are shown in (a), (b), (c), (d) and (e). Fixed update filter U_2 has been used. Reconstruction was done after 95% detail coefficients have been turned to zero.

coefficients being set to zero. The decomposition was firstly performed by using the fixed wavelet filter bank with the P_4 predict filter and the U_2 update filter. Blurring artifacts caused by the lossy reconstruction are clearly visible in the magnified images 5.22(e) and 5.22(f). Secondly, the filter parameters have been adapted by using the robust 2-D LSW adaptation with $N = 3 \times 3$ and $M = 6$. The resulting reconstructed image looks perceptually better. Those blurring artifacts have diminished. Yet, there is a strong ringing artifact on Barbara's leg (figure 5.22(i)) caused by high values of the adapted filter parameters in that high-frequency region. Filter parameters that are so big cause the divergence of the corresponding limit wavelet functions. Therefore, there was an error introduced by the thresholding of detail coefficients that was augmented across all the five decomposition levels to a greater extent than it would be augmented with fixed, "well-behaved" filters.



Figure 5.22: First row: original image. Following rows: lossy reconstruction after 5 decomposition levels and with 85% of detail coefficient being set to zero in every decomposition level. Second row: fixed P_4 and U_2 filters have been used. Third row: p_2 predict filter parameters have been adapted by using robust adaptation ($N = 3 \times 3$, $M = 6$).

Saturation of Filter Parameters

Another improvement in the lossy reconstruction is introduced by using saturation of the adapted filter parameters. The range of the allowed filter parameters is tightened

around zero to avoid too big positive or negative values of filter parameters. Therefore, all the values of p_2 parameters that have been set too big by the adaptation algorithm are replaced by appropriate maximum allowed values. The results of such an adaptation for different allowed ranges of filter parameters are shown in figure 5.23. The results obtained are perceptually better than the ones without the saturation used.



Figure 5.23: Lossy reconstruction same as in 5.22 with saturation of adapted filter parameters. Filter parameters p_2 have been limited to ranges $[-20, 20]$ (first row), $[-10, 10]$ (second row) and $[-5, 5]$ (third row).

Quantization of Filter Parameters

For the sake of a more compact representation of the filter parameters, their quantization has been additionally introduced (with recalculation of the detail coefficients). Figures 5.25(a), 5.25(b) and 5.25(c) show the reconstructed image after the quantization of the adapted filter parameters. Because of the ill-posed quanta, the results are perceptually very poor.

In order to make the quanta better fit into the true distribution of filter parameters' values we have used a simple CDF-based algorithm. Firstly, the cumulative distribution function of filter parameters has been calculated. Secondly, the algorithm finds N quantization values that are equidistant in terms of CDF values and $N - 1$ boundary values among the chosen quanta that make the quantizing partitions. All the values that fit into one given partition (between two boundary values) will be coded as the quantization value that lies in that partition. The final result is that the quantized filter parameters' values are equally distributed among N quanta. Figure 5.24 shows CDFs for p_2 in all the decomposition levels. There are four quanta chosen in every decomposition level. Their values are given in table 5.4.

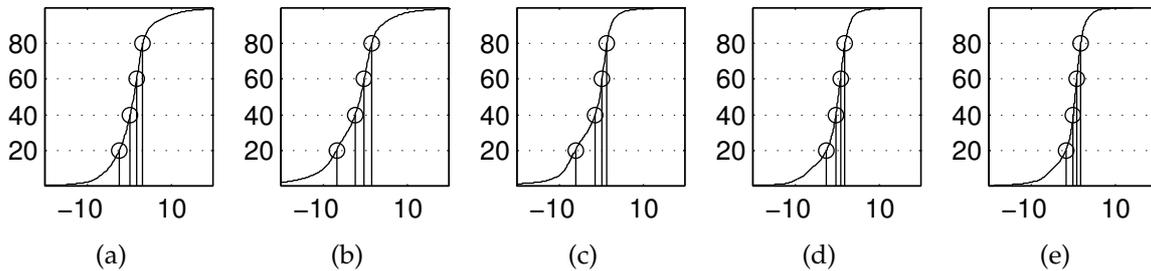


Figure 5.24: Cumulative distribution function for p_2 parameters in all five decomposition levels of image Barbara. Chosen equidistant quanta are marked.

1. lev.	2. lev.	3. lev.	4. lev.	5. lev.
-4.2	-9	-7.9	-5.3	-3.7
0.1	-3	-2	-0.39	0.01
2.2	0.06	0.37	0.95	1.1
4.5	3.1	2.1	2.7	2.5

Table 5.4: Values of the four CDF-based quanta chosen for all five decomposition levels shown in figure 5.24.

The results obtained with such a quantization of the filter parameters are comparable with those obtained without any quantization (see figure 5.25(d) and 5.25(g)). Perceptually, there is almost no difference between the images obtained by using such a

quantization and the ones obtained without any quantization and plain saturation of filter parameters. Very good results are even obtained with just four quanta!



Figure 5.25: First row: previously adapted p_2 parameters have been additionally quantized with the ill-posed quanta: [-20 -15 -10 -5 5 10 15 20]. Second row: quantization of p_2 is done by using 4 CDF-based quanta. Third row: quantization of p_2 is done by using 8 CDF-based quanta.

Lossy Reconstruction Quality Measure

To quantitatively express the reconstruction error, we will use the peak signal to noise ratio. PSNR in decibels (dB) is computed as

$$PSNR = 20 \log_{10} \frac{255}{RMSE} \quad (5.16)$$

where 255 is the highest pixel value (pixel values range from 0 to 255) and RMSE is the root mean squared error. Therefore, the PSNR measures the ratio of the peak value and the difference between the original and the reconstructed image. The root mean squared error is obtained by using

$$RMSE = \sqrt{MSE} = \frac{\sqrt{\sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} (x[n_1, n_2] - \hat{x}[n_1, n_2])^2}}{N} \quad (5.17)$$

where N is the total number of pixels ($N = N_1 N_2$), $x[n_1, n_2]$ is the original image and $\hat{x}[n_1, n_2]$ is the reconstructed image.

Table 5.5 gives the values of PSNR for different lossy reconstruction types performed on the four test images. It is obvious that the adaptation gives better PSNR as com-

	Fixed P_4	No qnt.	Saturation	Bad qnt.	CDF4 qnt.	CDF8 qnt.
Barbara	26.34	37.33	33.08	28.63	32.25	32.22
Mandrill	22.56	26.65	26.67	25.96	26.74	26.25
Lena	31.87	38.86	38.65	31.82	38.34	38.58
Goldhill	28.6	33.76	33.6	28.63	33.39	33.66

Table 5.5: PSNR calculated for reconstructed images after 5 decomposition levels and with 85% of detail coefficient being set to zero in every decomposition level. First column: fixed P_4 and U_2 filters have been used. Following columns: p_2 predict filter parameters have been adapted by using the robust adaptation ($N = 3 \times 3$, $M = 6$). Second column: neither saturation nor quantization of the p_2 filter parameters has been performed. Third column: values of p_2 are limited to a range from -10 to 10. Fourth column: values of p_2 have been quantized with the ill-posed quanta: [-20 -15 -10 -5 5 10 15 20]. Fifth and sixth column: values of p_2 have been quantized with 4 and 8 CDF-based quanta respectively.

pared to the fixed wavelets. The additional CDF-based quantization degrades the reconstructed image insignificantly while improving the compactness of the representation of the filter parameters.

Chapter 6

Conclusion

In this thesis there are presented the adaptive wavelet filter banks for subband image analysis. These filter banks are a kind of two-dimensional generalization of the one-dimensional filter banks previously reported by Seršić [Seršić 00].

These filter banks perform a wavelet analysis of a given image by adapting to its local properties. These are the second generation wavelets since the prototype wavelet function can change for every pixel of the image at all decomposition levels in order to obtain a wavelet representation that is as close to the optimal one as possible.

The filter bank structure is based on a lifting scheme introduced by Sweldens (section 2.3). The analysis and synthesis filter banks consist of two parts: the predict step and the update step. The proposed predict step is divided into a number of prediction branches. The first part of the prediction section is fixed, and the second part of the filter bank is variable, with one free prediction parameter in every prediction branch. The update step is realized by using a similar structure.

Although a number of filter parameters can be changed, some basic good properties of the wavelet decomposition are retained, such as the number of vanishing moments of the wavelet functions. They are guaranteed by the fixed part of the filter bank. A number of free parameters in the variable part of the filter bank can be changed without affecting the basic properties guaranteed by the fixed part of the filter bank (section 4.2).

Furthermore, if the number of fixed parameters of the update step (guaranteeing zeros of the corresponding low-pass filter) does not exceed the number of fixed parameters of the predict step (guaranteeing zeros of the corresponding high-pass filter), the update step can be varied independently of the predict step [Seršić 02a]!

The adaptation of the filter bank parameters has been performed by using various one-dimensional and two-dimensional least squares algorithms [Vrankić 02]. As expected, it has been demonstrated that the two-dimensional window-based adap-

tation algorithms show better results as compared to their one-dimensional counterparts. Spectrally corrected algorithms have been developed to further improve the adaptation properties. The BLUE method has been presented in section (section 5.3.5). In addition, a modification of the adaptation algorithm has been used to improve the robustness of adaptation to the transient image components (section 5.3.4).

The superiority of such an adaptive decomposition over a fixed wavelet decomposition is demonstrated on a lossy reconstruction of synthetic and real-world images (section 5.4.1). There have been the effects of quantization of the filter parameters discussed as well. A simple CDF-based algorithm has been used to choose the desired number of quanta. It has been shown that the quantization of filter parameters with as only as four quanta gives satisfying results that still outperform the results obtained with fixed filter banks.

6.1 Future Research

The adaptive filter banks proposed in this thesis show very good properties of the multiresolution image decomposition. Therefore, there is a wide range of applications which can utilize these filter banks such as the image compression, denoising and feature extraction. Future researches will be mostly concentrated towards applications based on these filter banks.

The primal goal is to implement algorithms for the lossy image compression. Since the proposed filter bank provides a transform that is tailored for the analyzed image while at the same time it retains good convergence and regularity properties of the limit wavelet functions, we expect to obtain very good compression ratios.

Bibliography

- [Beylkin 91] G. Beylkin, R. Coifman & V. Rokhlin. *Fast Wavelet Transforms and Numerical Algorithms*. Comm. Pure Applied Math., vol. 44, pages 141–183, 1991.
- [Daubechies 96] I. Daubechies & W. Sweldens. *Factoring Wavelet Transforms into Lifting Steps*. Rapport technique, Bell Laboratories, Lucent Technologies, 1996.
- [de Boor 90] C. de Boor & A. Ron. *On multivariate polynomial interpolation*. Constr. Approx, pages 6:287–302, 1990.
- [de Boor 92] C. de Boor & A. Ron. *Computational aspects of polynomial interpolation in several variables*. Math. Comp., pages 58:705–727, 1992.
- [Dubois 85] E. Dubois. *The sampling and reconstruction of time-varying imagery with application in video systems*. Proc.IEEE, vol. 73, no. 4, page 502–522, 1985.
- [Grossmann 84] A. Grossmann & J. Morlet. *Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape*. SIAM Journal of Mathematical Analysis, vol. 15, no. 4, pages 723–736, 1984.
- [Haykin 86] Simon Haykin. *Adaptive filter theory*. Prentice-Hall Information and System Science Series. Prentice-Hall, Englewood Cliffs, New Jersey, 1986.
- [Kovačević 92] J. Kovačević & M. Vetterli. *Nonseparable Multidimensional Perfect Reconstruction Filter Banks and Wavelet Bases for \mathcal{R}^n* . IEEE Trans. Inform. Th., special issue on Wavelet Transforms and Multiresolution Signal Analysis, vol. 38, no. 2, pages 533–555, March 1992. Invited paper.
- [Kovačević 00] J. Kovačević & W. Sweldens. *Wavelet Families of Increasing Order in Arbitrary Dimensions*. IEEE Trans. Image Proc., vol. 9, no. 3, pages 480–496, March 2000.

- [Mallat 89a] S. G. Mallat. *Multifrequency Channel Decompositions of Images and Wavelet Models*. IEEE Trans. Acoust. Speech Signal Process., vol. 37, no. 12, pages 2091–2110, 1989.
- [Mallat 89b] Stephane G. Mallat. *Multiresolution Approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$* . Transactions of the American Mathematical Society, vol. 315, no. 1, pages 69–87, September 1989.
- [Mallat 89c] Stephane G. Mallat. *A Theory for Multiresolution Signal Decomposition: The Wavelet Representation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 7, pages 674–693, July 1989.
- [Mayer 93] Y. Mayer. *Wavelets: Algorithms and applications*. SIAM, Philadelphia, 1993.
- [Meyer 90] Y. Meyer. *Ondelettes et opérateurs*. Hermann, Paris, 1990.
- [Mintzer 85] F. Mintzer. *Filters for Distortion-free Two-band Multirate Filter Banks*. IEEE Trans. ASSP, pages 626–630, June 1985.
- [Seršić 99] D. Seršić. *Razlaganje signala vremenski promjenjivim i nelinearnim filterskim slogovima*. PhD thesis, University of Zagreb, Croatia, 1999.
- [Seršić 00] D. Seršić. *A Realization of Wavelet Filter Bank With Adaptive Filter Parameters*. In Proc. of the European Signal Processing Conference, pages 3:1733–1736, Tampere, Finland, September 2000.
- [Seršić 02a] D. Seršić & M. Vrankić. *2-D Nonseparable Wavelet Filter Bank With Adaptive Filter Parameters*. In Proc. of the European Signal Processing Conference, pages 137–140, Toulouse, France, September 2002.
- [Seršić 02b] D. Seršić & M. Vrankić. *Adaptation of a 2-D Nonseparable Wavelet Filter Bank With Variable Number of Zero Moments*. In Proc. of the Second IASTED International Conference on Visualization, Imaging and Image Processing, pages 257–260, Malaga, Spain, September 2002.
- [Smith 86] M. J. Smith & T. P. Barnwell III. *Exact Reconstruction Techniques for Tree-Structured Subband Coders*. IEEE Trans. ASSP, vol. 34, pages 434–441, 1986.
- [Stein 75] E. M. Stein & G. Weiss. *Introduction to fourier analysis on euclidean spaces*. Princeton University Press, Princeton, NJ, 1975.

- [Strang 96] G. Strang & T. Nguyen. *Wavelets and filter banks*. Wellesley-Cambridge Press, Wellesley, MA, 1996.
- [Sweldens 95a] W. Sweldens. *The lifting scheme: A construction of second generation wavelets*. Rapport technique 1995:6, Department of Mathematics, University of South Carolina, 1995.
- [Sweldens 95b] W. Sweldens. *The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions*. In Andrew F. Laine, Michael A. Unser & Mladen V. Wickerhauser, editeurs, *Wavelet applications in signal and image processing III*, volume 2569 of *Proceedings of SPIE*, pages 68–79, 1995. 12-14 July, 1995, San Diego, California.
- [Sweldens 96] W. Sweldens. *The lifting scheme: A custom-design construction of biorthogonal wavelets*. *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pages 186–200, 1996.
- [Vaidyanathan 92] P. P. Vaidyanathan. *Multirate systems and filter banks*. Prentice Hall, Englewood Cliffs, NJ, 1992.
- [Vetterli 95] M. Vetterli & J. Kovacevic. *Wavelets and subband coding*. Prentice Hall, Englewood Cliffs, NJ, 1995.
- [Vrankić 02] M. Vrankić & D. Seršić. *Adaptation Methods Of 2-D Nonseparable Wavelet Filter Bank*. In *Proc. of the Second International Workshop on Spectral Methods and Multirate Signal Processing*, pages 235–242, Toulouse, France, September 2002.

Curriculum Vitae

Basic Information

Name: **Miroslav Vrankić**

Position: Research and teaching assistant at the
Faculty of Electrical Engineering and Computing,
Department of Electronic Systems and Information Processing,
Laboratory of Systems and Signals,
University of Zagreb, Croatia.

E-mail: Miroslav.Vrankic@FER.hr

WWW: <http://www.zesoi.fer.hr/~mvrankic>

Education

1994-1999 Diploma in EE,
Faculty Of Electrical Engineering and Computing, University of Zagreb, Croatia.

1990-1994 High school "Hrvatski kralj Zvonimir", Krk, Croatia.

Knowledge of Languages

Languages: 1. Croatian (native)
 2. English
 3. Italian

Professional Experience

- 1999-2003 Research and teaching assistant at the Faculty of Electrical Engineering and Computing, Department of Electronic Systems and Information Processing, Laboratory of Systems and Signals, University of Zagreb, Croatia.
- 2000-2001 Project leader on the implementation of the SMS and WAP services in Croatian Academic and Research Network (CARNET).

Teaching Activities

Random Processes in Systems	2002
Digital Signal Processing	2001
Process Measurement and Control Systems	2000, 2001
Software Design for Measurement and Control Systems	1999, 2000
Multimedia Data Communication and Computer Networks	1999, 2000

Publications

1. D. Seršić, M. Vrankić & I. Štajduhar. *Lossy Image Reconstruction Using Adaptive Wavelet Filter Bank*. Fifth IASTED International Conference on Signal and Image Processing, Honolulu, Hawaii, USA, August 2003. Accepted for presentation.
2. M. Vrankić & D. Seršić. *Adaptation Methods Of 2-D Nonseparable Wavelet Filter Bank*. In Proc. of the Second International Workshop on Spectral Methods and Multirate Signal Processing, pages 235–242, Toulouse, France, September 2002.
3. D. Seršić & M. Vrankić. *2-D Nonseparable Wavelet Filter Bank With Adaptive Filter Parameters*. In Proc. of the European Signal Processing Conference, pages 137–140, Toulouse, France, September 2002.

4. D. Seršić & M. Vrankić. *Adaptation of a 2-D Nonseparable Wavelet Filter Bank With Variable Number of Zero Moments*. In Proc. of the Second IASTED International Conference on Visualisation, Imaging and Image Processing, pages 257–260, Malaga, Spain, September 2002.
5. M. Vrankić. *WAP - Concepts, Technologies and Open Solutions*. Open Systems' Days, Zagreb, Croatia, April 2001.
6. M. Vrankić. *WAP and Open Source Community: Kannel Gateway*. In Proc. of Communication Technologies and Standards in Informatics, Opatija, Croatia, December 2000.

Abstract

More efficient coding, modelling and image analysis generate the need for search for better methods of the multiresolution image analysis, i.e. for more efficient wavelet filter bank structures. This thesis is based on a previous research of one-dimensional filter bank structures that had the possibility to adapt filter parameters to the properties of the analyzed signal.

In this thesis we present the construction of a two-dimensional adaptive wavelet filter bank that is based on a lifting scheme. The filter bank is nonseparable, based on a quincunx polyphase decomposition and nonseparable filters. The lifting scheme has been chosen since it allows for an easy construction of space varying filter banks with a perfect reconstruction property. The proposed filter bank adapts to the analyzed image for every pixel in all decomposition levels while still preserving the good properties of the wavelet decomposition. A number of vanishing moments is guaranteed by the fixed part of the filter bank. Without degrading the overall filter bank properties, the variable part can be changed in order to adapt to the analyzed image. The paper explores various one-dimensional and two-dimensional adaptation methods based on the least squares criterion.

Adaptation results have been shown for a number of synthetic and real-world images. Effects of lossy image reconstruction and impact of filter coefficients' quantization to the efficiency of the image decomposition have been presented.

Keywords: wavelet filter banks, second generation wavelets, multiresolution analysis, adaptive lifting scheme, quincunx polyphase decomposition, nonseparable filters, interpolating filters, lossy image reconstruction.

Sažetak

Učinkovitije kodiranje, modeliranje i analiza slika stvaraju potrebu za traženjem boljih metoda višerezolucijskog razlaganja slika odnosno potrebu za učinkovitijim strukturama dvodimenzionalnih wavelet filtarskih slogova. Rad se temelji na prethodnom istraživanju struktura jednodimenzionalnih filtarskih slogova koji su imali mogućnost adaptacije filtarskih parametara svojstvima signala.

U ovom radu prikazujemo izvedbu dvodimenzionalnog adaptivnog wavelet filtarskog sloga koji se temelji na shemi podizanja. Filtarski slog je neseparabilan, temelji se na quincunx polifaznom razlaganju i neseparabilnim filtrima. Shema podizanja je odabrana jer omogućava jednostavnu izvedbu prostorno promjenjivih filtarskih slogova sa svojstvima savršene rekonstrukcije. Predloženi filtarski slog se prilagođuje analiziranoj slici u svakom slikovnom elementu u svim razinama razlaganja u isto vrijeme zadržavajući dobra svojstva wavelet razlaganja. Dovoljan broj nul-momenata pridruženih wavelet funkcija zagarantiran je nepromjenjivim dijelom filtarskog sloga. Promjenjivi dio filtarskog sloga može se prilagođivati svojstvima slike bez narušavanja cjelokupnih svojstava filtarskog sloga. Istražene su razne jednodimenzionalne i dvodimenzionalne metode adaptacije zasnovane na kriteriju najmanjih kvadrata.

Rezultati adaptacije prikazani su za razne sintetske i realne slike. Prezentirani su rezultati rekonstrukcije slike s gubicima i utjecaj kvantizacije filtarskih koeficijenata na učinkovitost adaptivnog razlaganja slike.

Ključne riječi: wavelet filtarski slogovi, waveleti druge generacije, multirezolucijska analiza, adaptivna shema podizanja, quincunx polifazno razlaganje, neseparabilni filtri, interpolacijski filtri, rekonstrukcija slike s gubicima.