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

Miroslav Vrankić

ADAPTIVE LIFTING SCHEME FOR NONSEPARABLE TWO-DIMENSIONAL WAVELET TRANSFORMS

ADAPTIVNA SHEMA PODIZANJA ZA NESEPARABILNE DVODIMENZIONALNE VALIĆNE TRANSFORMACIJE

DOCTORAL THESIS
DOKTORSKA DISERTACIJA

Zagreb, 2006

The doctoral thesis has been realized at the Department of Electronic Systems and Information Processing, Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia, and at the Tampere International Center for Signal Processing, Tampere University of Technology, Finland.

Supervisor: Dr.sc. Damir Seršić, assistant professor

The doctoral thesis consists of 165 pages

The dissertation evaluation committee:

1. Professor Sven Lončarić
2. Assistant professor Damir Seršić
3. Professor Hrvoje Šikić, Faculty of Science, University of Zagreb

The dissertation defense committee:

1. Professor Sven Lončarić
2. Assistant professor Damir Seršić
3. Professor Hrvoje Šikić, Faculty of Science, University of Zagreb
4. Professor Sonja Grgić
5. Dr. Atanas Gotchev, Tampere International Center for Signal Processing, Tampere University of Technology, Finland

Date of dissertation defense: 6^{th} July, 2006

Contents

Chapter 1

Introduction

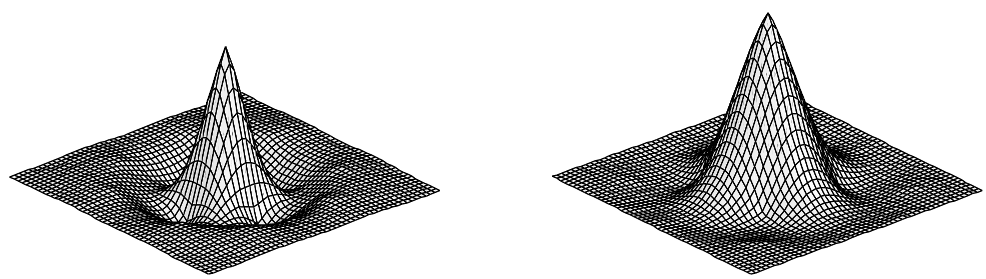
1.1 Motivation

In this thesis, images will be treated as two-dimensional signals, so the words signals and images will be interchanged with the similar meaning. We will use a wavelet transform in order to process images. By using the transform one wants to obtain a more compact representation of an image. Applying a transform means splitting an image into a set of elementary functions, under some conditions called basis functions [Vetterli 95]. The image will be represented as a weighted sum of those basis functions. The values of those weights, obtained as scalar products with the basis functions, are called the transform coefficients.

One of the best known transforms is the Fourier transform. It decomposes an analyzed signal into a set of weighted complex exponentials [Katznelson 76]. It is very useful for analyzing periodic signals. Yet, the treatment of some short transients present in the signal is rather poor. One says that the Fourier transform has good frequency resolution but poor temporal resolution. An improvement of the given problem is to use windowed basis functions. It leads to the well-known Short-Time Fourier transform (STFT) [Cohen 94]. It creates uniform tilings in time and frequency.

Compared to the STFT, the wavelet transform gives more convenient splitting of the time-frequency plane [Mertins 99]. The frequency axis is logarithmic with larger time step in the time-frequency plane for smaller frequencies and finer time sampling for higher frequencies. The decomposition relies on a set of locally supported functions that are translates and dilates of one "mother" function. Such a decomposition can cope better with local variations in the signal. It means that although the analyzed signal is composed of some temporal glitches, the resulting transform coefficients will be sparse, leading to a more efficient description of the analyzed signal. The sparser the signal representation the better the transform is.

An important concept that comes with wavelets is the multiresolution [Mallat 89a]. Wavelet transform leads to a set of different resolution levels called scales. Every resolution level is represented with basis functions that are appropriately scaled. There are two types of basis functions: wavelet functions and scale functions (see Figure 1.1). The scale functions are used to approximate the analyzed signal in different resolution levels while the wavelet functions are used to add more detail to the approximation from a given scale. The wavelet transform coefficients corresponding to the wavelet function are called detail coefficients while the coefficients corresponding to the scaling functions are called approximation coefficients. Starting from a coarse approximation of a given image that is obtained with very wide scaling functions, one can obtain more precise approximations of the original image by adding wavelet detail coefficients from finer and finer resolution levels in order to obtain the original image.



(a) The wavelet function.

(b) The scale function.

Figure 1.1: An example of two-dimensional nonseparable wavelet and scale functions derived in [Kovacevic 00].

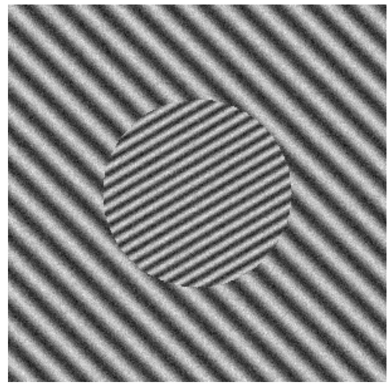
Additional important property of wavelets is the polynomial approximation property [Burrus 97]. Since most real-world images can be approximated with piecewise polynomials, the wavelets will be appropriate to analyze them. Detail coefficients for such signals will be mostly set to zero, since the approximation coefficients will hold the most of the information.

While being very efficient in representing real-world images, wavelets still do not give the best possible, i.e. the sparsest possible representation of the signal. In order to obtain a more efficient image decomposition, one should change the basis functions according to the local properties of the analyzed image. This way, one will approach the goal to obtain a transform that gives more compact representation of an image.

1.1.1 Custom Designed Image Decomposition

Let us consider a simple test image shown in Figure 1.2(a). It consists of two distinct parts, each part contains a single spatial sine wave with a different frequency and orientation.

(a) Original image.



(b) Noisy image.

Figure 1.2: Test images composed of two different patterns.

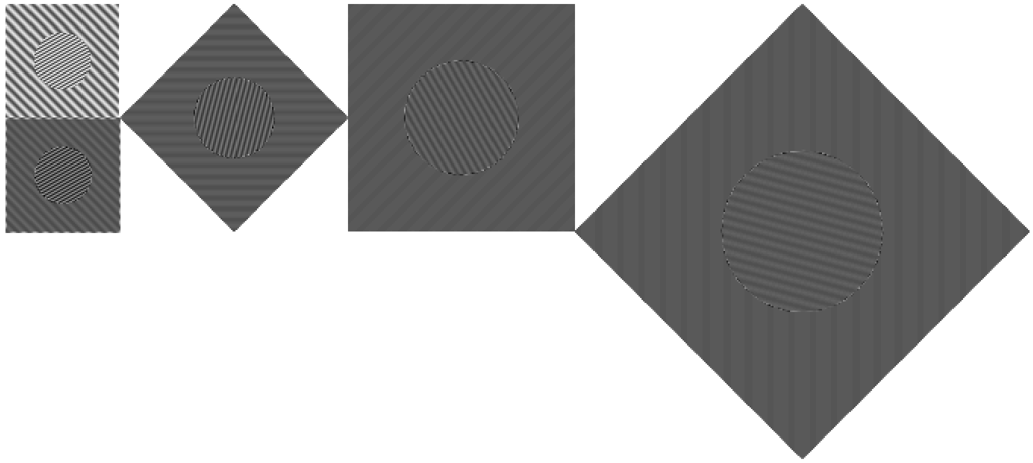
Figure 1.3(a) represents the image decomposition obtain by using nonseparable interpolating quincunx wavelets that are presented in Section 2.3.3. The original image is represented with a single image of coarse approximation coefficients and a number of detail coefficients at different resolution levels.

It can be seen that for both regions the wave pattern is still present in the detail coefficients, meaning that the decomposition is not tailored to the underlying signal. So besides the coarse approximation additional details are needed to have a complete description of the analyzed image. In order to tailor a decomposition for the analyzed image, one should adapt it to the local image properties. For a given image, one would like to have two flavors of the basis functions for the two distinct regions of the image shown in Figure 1.2(a). Therefore, an appropriate decomposition should be found for each distinct region of the given image.

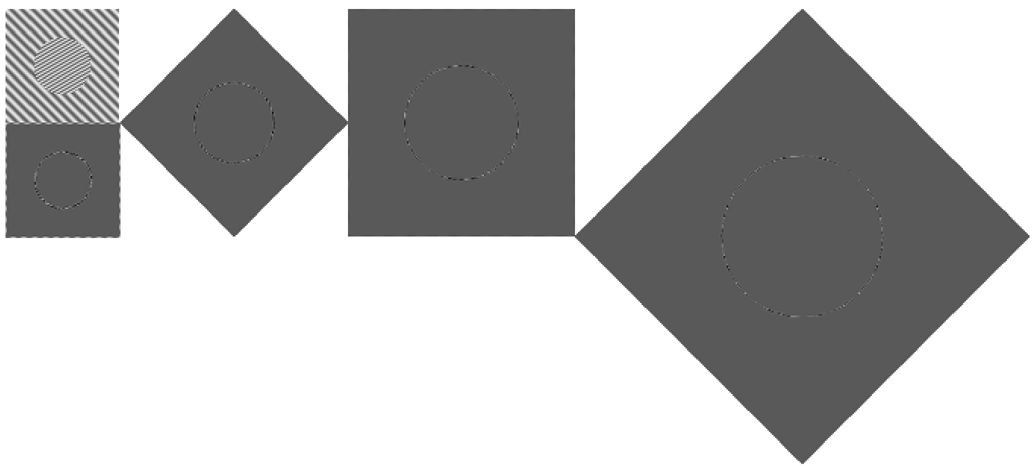
The result of applying different decompositions in two different regions is shown in Figure 1.3(b). The obtained representation is much more sparse. Values of detail coefficients are more close to zero since the decomposition is now more appropriate for a given image.

1.1.2 Oracle Image Denoising

Such an improved transform is much more appropriate for analysis and processing of the given image. The reason is that it is better tailored to the features that character-



(a) Fixed wavelet decomposition.



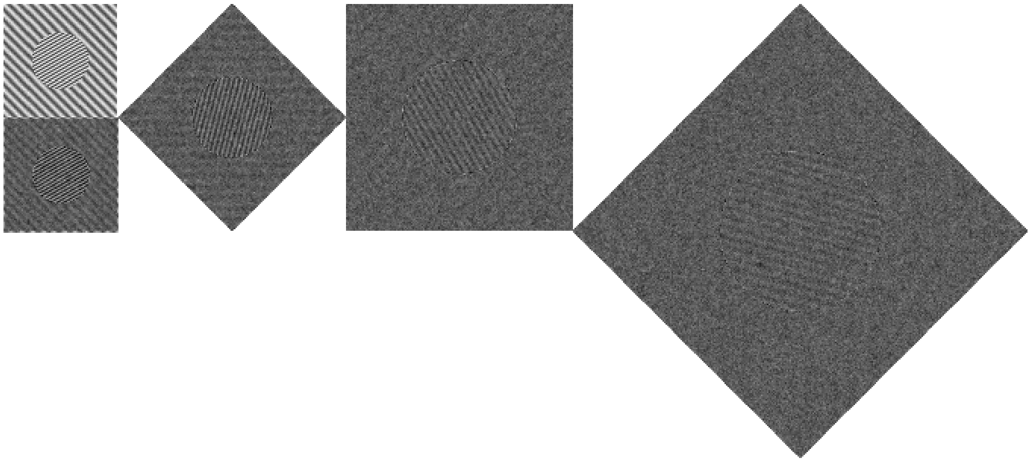
(b) Adaptive wavelet decomposition.

Figure 1.3: Approximation and detail coefficients for nonseparable quincunx wavelet decomposition of the image shown in Figure 1.2(a). Coarse approximation coefficients are shown top left. Images of detail coefficients are shown from the left to the right from the coarsest to the finest resolution levels.

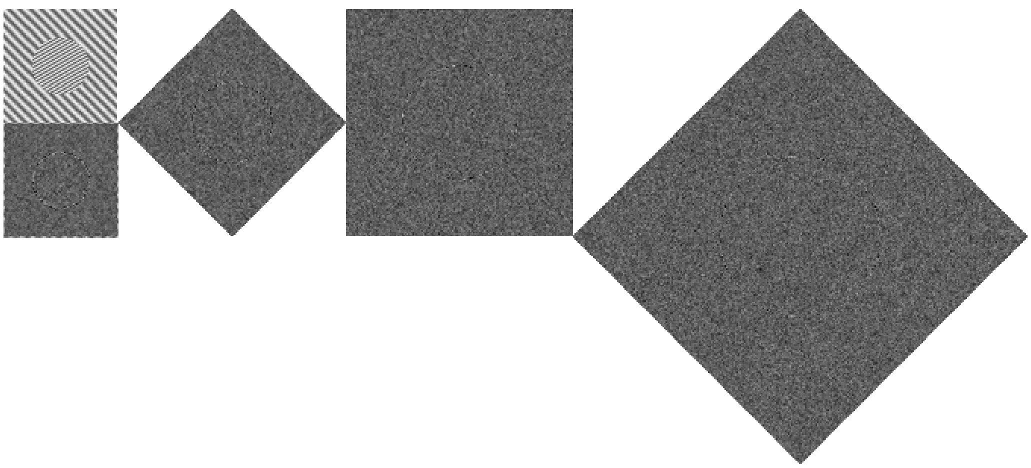
ize the given image. To justify the usefulness of this approach of custom designing a wavelet transform, we will demonstrate its use in an image denoising application.

Wavelet domain strategies for image denoising rely on the threshold operation of the wavelet detail coefficients [Donoho 95a, Donoho 95b]. Simply, the detail coefficients smaller than a given threshold will be set to zero. One expects those small wavelet coefficients to represent insignificant information present in the image and (hopefully) mostly related to the noise. The reconstruction is performed by using those thresholded wavelet coefficients. The choice of the threshold determines the compromise between the noise reduction and the original image features degradation.

First, let us consider the noisy image given in Figure 1.2(b). It is the original image from Figure 1.2(a) degraded with additive zero mean Gaussian noise. As shown in Figure 1.4(a), detail coefficients obtained with a nonadaptive wavelet transform contain information on both the original image features and the noise.



(a) Fixed wavelet decomposition.



(b) Adaptive wavelet decomposition.

Figure 1.4: Wavelet decomposition of a noisy image from Figure 1.2(b).

Consider now the previously presented locally adaptive wavelet transform. Since it is tuned to the main features of the image, the detail coefficients will predominantly contain the information on the noise, i.e. the content not inherent to the image 1.4(b). Now, applying a rather high threshold will not be a problem. It will reduce the noise efficiently without degradation of the original image. These results will be presented in Chapter 6.

The presented example emphasizes the need to improve the wavelet transform and make it sensitive to the local features of the image.

1.2 Problem Statement

Continuing the above discussion, we state that we want to obtain a wavelet transform that adapts its properties according to the features of the analyzed image in order to obtain a more compact representation of the analyzed image.

Wavelet Transform by Filter Banks

Since we are primarily interested in analyzing digital images, we will use the discrete wavelet transform. As shown in [Mallat 89b], the wavelet transform coefficients can be calculated using a discrete-time filter bank. Once having the filters, one can easily come to the underlying analysis and synthesis functions as it will be shown in Chapter 2. So, we are shifting our problem from pure mathematical domain to the filter bank world. In other words, in order to obtain a different wavelet transform, we will use different filter banks. Furthermore, to have locally adaptive wavelet transform, we will create a wavelet filter bank that can adapt its properties according to the local image features.

Vanishing Moments

While building the adaptive filter bank we would also like it to retain some important wavelet properties. First of all, we want to retain the polynomial approximation property of the resulting wavelet decomposition. In other words, we would like to retain a desired number of vanishing moments of dual and primal wavelet functions [Mallat 99].

The Lifting Scheme

In order to build such an adaptive wavelet filter bank we choose to use a concept called lifting scheme [Sweldens 96b]. Besides being algorithmically very efficient, its main feature is that it allows rather straightforward construction of the wavelet filter bank. It automatically satisfies the desired perfect reconstruction property of the filter bank. Being constructed as a tool to build second generation wavelets, wavelets that are not necessarily translates and dilates of a single function, it is very susceptible to introduction of the adaptivity in the filter bank structure. By using the lifting scheme, we can modify the existing wavelet transform, retain its good approximation properties, and further improve it with additional lifting steps, as shown in Chapter 3.

Nonseparable Versus Separable

Furthermore, we want to make our decomposition isotropic, i.e. we want it to have similar properties in all directions in the image. Typically, separable wavelet filter banks are more often used in image processing than nonseparable wavelet filter banks. Their most important advantage is that the transform coefficients can be obtained by using one-dimensional filters which are first being applied over rows and then over the columns of the analyzed image [Mallat 89b]. However, these separable wavelet transforms are biased in horizontal and vertical directions. On the other hand, nonseparable wavelet filter banks require two-dimensional image processing which is computationally more demanding but can result in more isotropic wavelet decompositions. For that reason we choose to deal with nonseparable wavelet filter banks. In particular, we choose the quincunx sampling, which is the simplest nonseparable sampling scheme.

Adaptation of the Wavelet Filter Bank

The adaptation of the filter bank parameters will be based on the local properties of the image. For each pixel, the filter bank properties will be tuned according to some pixel neighborhood. In order to obtain better results we want the neighborhood to cover only pixels that have similar properties as the pixel for which the adaptation is being performed. It is desirable to have a neighborhood that is wide enough, in order to obtain a more reliable estimate, while still covering only a part of the image of the same property.

Denoising Applications

We want to make our adaptation robust to the presence of noise. In other words, we want to make the adaptation algorithm focus on the content of the original image and disregard the locally uncorrelated noise.

1.3 Related Work

1.3.1 Wavelet Revolution

The term wavelet appeared in the literature for the first time in 1984 [Grossmann 84]. The petroleum engineer Jean Morlet introduced "wavelets of constant shape" that he found very useful in analyzing geologic signals. Soon after that, in 1985, Meyer discovered the first smooth orthogonal wavelet [Meyer 95].

The true breakthrough for the engineering community happened when Stephane Mallat has shown the relation between wavelets and filter banks [Mallat 89b]. The filter bank theory was connected with multiresolution analysis and wavelets. Finally, the wavelet decomposition was made algorithmically very efficient, using the iterated filter bank structure.

Ingrid Daubechies presented the construction of the first orthogonal wavelets with compact support [Daubechies 88]. Her wavelet decomposition is obtained by using iterated filter bank structure with finite impulse response (FIR) filters [Daubechies 92].

Further improvement in wavelet construction and implementation came with the introduction of the lifting scheme by Wim Sweldens [Sweldens 95], [Sweldens 96a]. The lifting scheme provided an effective tool to build biorthogonal wavelet filter banks. Furthermore, because of its inherent perfect reconstruction properties, it gave rise to the second generation wavelets [Sweldens 97], wavelets that are not necessarily translated and dilated copies of a single function.

1.3.2 Nonseparable Wavelets

Wavelet decomposition relies heavily on the underlying subsampling scheme used. In the one-dimensional case a signal is typically being split into two phases corresponding to even and odd-numbered samples. In the two-dimensional case, an image can be subsampled in many different ways [Dubois 85, Theussl 01, Dudgeon 83]. The simplest possible choice is the separable sampling, obtained by sampling and filtering along rows and columns of the image separately [Mallat 89b].

The alternative is to use nonseparable sampling scheme leading to nonseparable wavelets. The treatment of two-dimensional and multidimensional FIR perfect reconstruction filter banks for arbitrary sampling lattices can be found in [Karlsson 90] and [Viscito 91]. A design of multidimensional wavelet filter banks is presented in [Kovacevic 92, Simoncelli 90, Chen 93]. The constructions of two-dimensional nonseparable wavelet bases [Kovacevic 95, Cohen 93] allows one to obtain a wavelet decomposition with different orientational properties than the separable wavelet decomposition. In [Kovacevic 00], the construction of compactly supported biorthogonal wavelets and perfect reconstruction filter banks for any lattice in any dimension with any number of primal and dual vanishing moments was presented.

1.3.3 Adaptive Wavelets and Adaptive Lifting Schemes

Wavelets that can adapt their properties to the features of the analyzed signal represent a step further towards more efficient image processing algorithms. The adaptations can be generally done in two ways: globally or locally. Global adaptation searches for a wavelet decomposition that does not change across the image. Instead, it is tuned for the whole image. Typical representative for this group is the best basis algorithm [Coifman 92]. Depending on the criterion (e.g. signal entropy), the optimal basis for the whole image is obtained. However, the adaptation that is performed locally, based on the local features of the analyzed image, can outperform the global adaptation methods.

An example of a locally adapted wavelet transform is the so-called ENO-wavelet transform [Chan 02]. The authors do not change the filter coefficients but rather the input signal in order to treat discontinuities in a more efficient way. The idea comes from essentially non-oscillatory (ENO) schemes for numerical shock capturing [Harten 97]. The idea is to locate the discontinuities in the signal and then to extrapolate smooth signals on opposite sides of the discontinuity.

The lifting scheme structure, being based on predict and update steps that automatically lead to the perfect reconstruction property, allows one to directly introduce the adaptation in the filter bank. Claypoole et al. proposed the so-called space-adaptive transform (SpAT) based on the lifting scheme that for every signal sample changes the predictor order in order to minimize detail coefficients [Claypoole 97, Claypoole 03]. Therefore, near the edges the shorter prediction filters will be used in order to avoid prediction over the edges. In order to obtain greater flexibility in the prediction filter design while still retaining desirable properties of the wavelet transform, they perform the update step first, followed by the adaptive prediction. The update first scheme allows to maintain multiresolution properties despite the introduced nonlinearity through the adaptive predict step.

A locally adaptive filter bank structure for image coding has been proposed by Gerek and Cetin [Gerek 00]. Their filter bank is based on the lifting scheme with the adaptation of the prediction filter. The least mean squares (LMS) [Haykin 86] adaptation algorithm was used in order to improve the properties of the resulting polyphase structure. They use both separable and quincunx prediction filters. However, the proposed adaptive structures do not have wavelet properties. In [Gerek 06] they present edge sensitive adaptive prediction based on the similar lifting scheme structure.

Adaptive prediction stage is also used in [Boulgouris 01] to improve the image decorrelation efficiently for the purpose of lossless image compression. Similar to the previous approach, these filter banks do not have wavelet properties either.

The approach presented in [Piella 02a] and [Piella 02b] is different. They use the update first structure with the adaptation of the update step. The update filter is being changed based on the local gradient information in order to smoothen less the sharp variations in the signal than the more homogenous regions. Also, the choices of update filters can be automatically reconstructed on the synthesis side so no bookkeeping is required in order to have perfect reconstruction.

In another approach presented in [Gouze 04], in order to obtain multidimensional nonseparable wavelet transform, the prediction step is designed in order to minimize the variance of the signal, and the update step is designed in order to minimize a reconstruction error.

In [Seršić 02, Vrankić 04], Seršić and Vrankić proposed a nonseparable two-dimensional adaptive wavelet filter bank based on the one-dimensional construction proposed in [Seršić 00]. The proposed two-dimensional construction is based on the lifting scheme and it introduces the adaptation in the predict stage of the filter bank while still retaining a desired number of vanishing moments of the corresponding wavelet functions.

1.4 Thesis Outline

Fundamentals

Chapter 2 gives an overview of the wavelet and filter bank theory. The concepts will be presented for the general two-dimensional sampling and the quincunx sampling in particular. The theory will be supplemented with the examples obtained by using the quincunx interpolating filter bank proposed in [Kovacevic 00]. The filter bank will be used in the following chapters in order to obtain an improved wavelet decomposition by means of adaptive lifting scheme structure.

Making the Wavelet Filter Bank Adaptive

In Chapter 3, the quincunx interpolating filter bank based on the lifting scheme will be made adaptive. This chapter presents the construction of adaptive wavelet filter banks based on the adaptation of the predict filter. Although being adaptive, the filter bank will retain the important wavelet properties such as the number of vanishing moments.

Tuning the Filter Parameters

Once having the adaptive filter bank structure, in Chapter 4 we introduce the adaptation criterion: The energy of detail coefficients will be minimized. The adaptation is being performed for every pixel of the image based on the neighborhood of a free shape.

In Pursuit for Appropriate Adaptation Regions

Chapter 5 provides a method for obtaining the appropriate adaptation neighborhood for each pixel of the image. It is based on the Intersection of Confidence Intervals (ICI) rule [Katkovnik 99].

Image Denoising

In Chapter 6 the proposed adaptation algorithm is used in the framework of image denoising by wavelet thresholding. The image denoising results are presented for both synthetic and real-world images.

Chapter 2

Two-Dimensional Quincunx Wavelets and Filter Banks

2.1 Introduction

This chapter gives an overview of the wavelet and filter bank theory. It is rather short and giving only the most important results. A more detailed treatment of these basic concepts can be found in [Vetterli 95, Vaidyanathan 92].

First, we give an introduction to two-dimensional filter banks. Although the theory can be generalized for multiple dimensions, all the results are mostly stated for 2-D case, and for quincunx sampling in particular. The lifting scheme fundamentals are also explained in brief.

Following the filter bank theory overview, the wavelet theory overview is provided. It emphasizes the main concepts of multiresolution analysis and the connection between wavelets and filter banks. The results are again presented for two-dimensional case and in particular for the interpolating quincunx wavelet filter bank designed by Kovacevic and Sweledens [Kovacevic 00].

2.2 Two-dimensional Sampling

2.2.1 Notation

In order to simplify mathematical expressions which treat two-dimensional signals, we introduce vector notation

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}, \qquad X(\mathbf{z}) = X\left(\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}\right) = X\left(z_1, z_2\right). \tag{2.1}$$

Taking another vector $\mathbf{k} = [k_1\ k_2]^T$ we define

$$\mathbf{z}^{\mathbf{k}} = z_1^{k_1} \cdot z_2^{k_2} \tag{2.2}$$

Raising the $\mathbf{z}$ vector to a 2×2 matrix $\mathbf{M}$ power yields

$$\mathbf{z}^{\mathbf{M}} = \begin{bmatrix} \mathbf{z}^{m_1} \\ \mathbf{z}^{m_2} \end{bmatrix}, \tag{2.3}$$

where $\mathbf{m}_i$ denotes the i-th column of the matrix $\mathbf{M}$. Combining relations 2.2 and 2.3 yields

$$\mathbf{z}^{\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}} = \begin{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}^{\begin{bmatrix} 1 \\ -1 \end{bmatrix}} & \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}^{\begin{bmatrix} 1 \\ 1 \end{bmatrix}} \end{bmatrix}^T = \begin{bmatrix} z_1 z_2^{-1} \\ z_1 z_2 \end{bmatrix}. \tag{2.4}$$

Finally, the Schur product of two vectors is defined as:

$$\mathbf{m} \circ \mathbf{n} = \begin{bmatrix} m_1 n_1 \\ m_2 n_2 \end{bmatrix}. \tag{2.5}$$

2.2.2 Sampling Lattices

The fundamental part of the multirate system is the subsampling operation. The sampling operation takes sets of signal elements (image pixels) that are positioned on a certain sampling grid called lattice [Conway 93]. Examples of some typical sampling lattices are shown in Figure 2.1.

The sampling lattice is completely defined with a dilation matrix [Dubois 85]

$$LAT(\mathbf{D}) = \mathbf{Dn}, \qquad \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \in \mathcal{Z}^2. \tag{2.6}$$

A dilation matrix is an integer matrix whose columns are basis vectors of a given lattice.

The separable sampling lattice from Figure 2.1(a) is defined with

$$\mathbf{D}_s = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \tag{2.7}$$

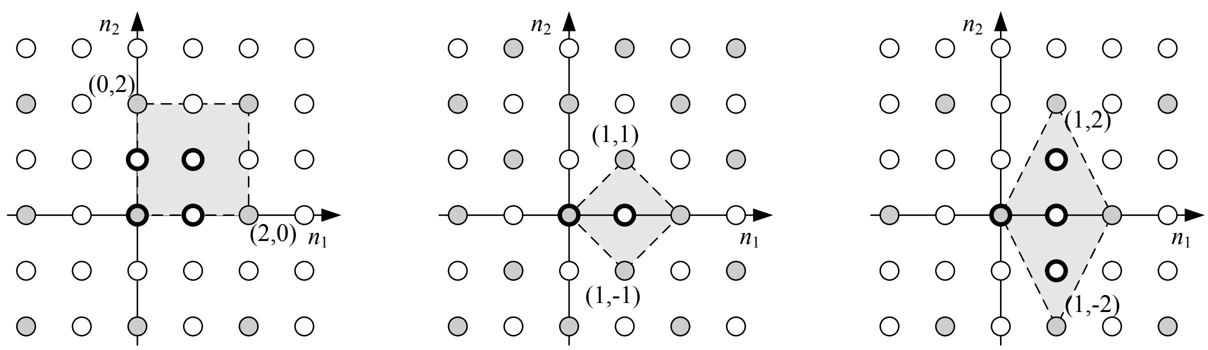
The quincunx sampling lattice from Figure 2.1(b) is defined with

$$\mathbf{D}_q = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}. \tag{2.8}$$

The same lattice can also be obtained with the sampling matrices

$$\mathbf{D}_{q2} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \mathbf{D}_{q3} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \tag{2.9}$$

A unit cell of a lattice spanned by colum vectors of a dilation matrix is called fundamental parellelepiped (*FPD*). The number of integer points belonging to the *FPD* is obtained as $|\det(\mathbf{D})|$. For the separable case, as shown in Figure 2.1(a), there are $M = |\det(\mathbf{D}_s)| = 4$ elements of the *FPD*, and for the quincunx case (Figure 2.1(b)) there are $M = |\det(\mathbf{D}_q)| = 2$ elements of the *FPD*. It is important to note that only the element of the *FPD* positioned at the origin belongs to the lattice. All the other elements of the *FPD* can be regarded as origins of the other shifted lattices. Therefore, we can conclude that the integer rectangular lattice can be split into $M = |\det(\mathbf{D})|$ disjoint (sub)lattices.



(a) Separable sampling. (b) Quincunx sampling. (c) Hexagonal sampling.

Figure 2.1: Three different sampling lattices. The fundamental parallelepiped is marked gray. Samples belonging to the FPD are marked with bold circles.

2.2.3 Downsampling

The downsampling operation of an image consists of discarding all the samples that are not elements of a given lattice [Vaidyanathan 92]. Therefore,

$$y_D[\mathbf{n}] = x[\mathbf{D}\mathbf{n}], \qquad \text{for every } \mathbf{n} = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \in \mathcal{Z}^2. \tag{2.10}$$

For the quincunx downsampling, relation 2.10 becomes

$$y_D[n_1, n_2] = x[n_2 + n_1, n_2 - n_1], \tag{2.11}$$

Fourier domain equivalent of relation 2.10 is

$$Y_D(\boldsymbol{\omega}) = \frac{1}{|\det \mathbf{D}|} \sum_{k \in \mathcal{N}(\mathbf{D}^T)} X(\mathbf{D}^{-T}(\boldsymbol{\omega} - 2\pi\mathbf{k})), \tag{2.12}$$

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 $\boldsymbol{\omega} = [\omega_1 \;\; \omega_2]^T$. A spectrum of the decimated signal is obtained by adding $M = |\det \mathbf{D}|$ copies of $X(\mathbf{D}^{-T}\boldsymbol{\omega})$ which is a stretched input signal's spectrum. Each stretched copy is shifted for the corresponding $2\pi\mathbf{k}$. For the quincunx case

$$Y_D(\boldsymbol{\omega}) = \frac{1}{2}[X(\begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix}) + X(\begin{bmatrix} 0.5 & 0.5 \\ -0.5 & 0.5 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 - 2\pi \end{bmatrix})]. \tag{2.13}$$

It the z-domain,

$$Y_D(\mathbf{z}) = \frac{1}{|\det \mathbf{D}|} \sum_{k \in \mathcal{N}(\mathbf{D}^T)} X\left(\mathbf{z}^{\mathbf{D}^{-1}} \circ e^{-j2\pi\mathbf{k}^T\mathbf{D}^{-1}}\right), \tag{2.14}$$

where $\mathcal{N}(\mathbf{D}^T)$ is a set of integer vectors in the fundamental parallelepiped $FPD(\mathbf{D}^T)$.

For the quincunx case

$$\mathbf{D} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \qquad \mathbf{D}^{-1} = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}, \tag{2.15}$$

$$Y_D(\mathbf{z}) = \frac{1}{2}\left[X\left(\mathbf{z}^{\mathbf{D}^{-1}} \circ e^{-j2\pi\mathbf{k}_1^T\mathbf{D}^{-1}}\right) + X\left(\mathbf{z}^{\mathbf{D}^{-1}} \circ e^{-j2\pi\mathbf{k}_2^T\mathbf{D}^{-1}}\right)\right], \tag{2.16}$$

where

$$\mathbf{k}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \qquad \mathbf{k}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2.17}$$

Therefore,

$$Y_D(\mathbf{z}) = \frac{1}{2}\left[X\left(\mathbf{z}^{\mathbf{D}^{-1}} \circ \begin{bmatrix} 1 \\ 1 \end{bmatrix}\right) + X\left(\mathbf{z}^{\mathbf{D}^{-1}} \circ \begin{bmatrix} -1 \\ -1 \end{bmatrix}\right)\right], \tag{2.18a}$$

$$Y_D(\mathbf{z}) = \frac{1}{2}\left[X\left(\mathbf{z}^{\mathbf{D}^{-1}}\right) + X\left(-\mathbf{z}^{\mathbf{D}^{-1}}\right)\right], \tag{2.18b}$$

or written in a different way

$$Y_D(z_1, z_2) = \frac{1}{2}\left[X\left(z_1^{\frac{1}{2}} z_2^{\frac{1}{2}}, z_1^{-\frac{1}{2}} z_2^{\frac{1}{2}}\right) + X\left(-z_1^{\frac{1}{2}} z_2^{\frac{1}{2}}, -z_1^{-\frac{1}{2}} z_2^{\frac{1}{2}}\right)\right]. \tag{2.19}$$

2.2.4 Upsampling

Upsampling an image corresponds to placing the image pixels on the lattice points and setting all the other integer valued pixels outside the lattice to zero [Vaidyanathan 92]:

$$y_E[\mathbf{n}] = \begin{cases} x[\mathbf{D}^{-1}\mathbf{n}] & \text{if } \mathbf{n} \in LAT(\mathbf{D}), \\ 0 & \text{otherwise.} \end{cases} \tag{2.20}$$

For the quincunx case

$$y_E[n_1, n_2] = \begin{cases} x[\frac{1}{2}(n_1 - n_2), \frac{1}{2}(n_1 + n_2)] & \text{if } \mathbf{n} \in LAT(\mathbf{D}), \\ 0 & \text{otherwise.} \end{cases} \tag{2.21}$$

Equation 2.20 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:

$$Y_E(\boldsymbol{\omega}) = X(\mathbf{D}^T\boldsymbol{\omega}), \tag{2.22}$$

or expressed in the z-domain:

$$Y_E(\mathbf{z}) = X(\mathbf{z}^{\mathbf{D}}). \tag{2.23}$$

Given a quincunx dilation matrix, equation 2.23 becomes

$$Y_E(z_1, z_2) = X(\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}^{\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}}) = X(\begin{bmatrix} z_1 \\ z_2 \end{bmatrix}^{\begin{bmatrix} 1 \\ -1 \end{bmatrix}}, \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}^{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}) = X(z_1 z_2^{-1}, z_1 z_2) \tag{2.24}$$

Quincunx Dowsampling Followed by Upsampling

For the case of quincunx sampling, the result of downsampling followed by upsampling can be obtained by using equations 2.19 and 2.24:

$$Y(z_1, z_2) = \frac{1}{2}\left[X(z_1, z_2) + X(-z_1, -z_2)\right], \tag{2.25}$$

or written in the Fourier domain

$$Y(\omega_1, \omega_2) = \frac{1}{2}\left[X(\omega_1, \omega_2) + X(\omega_1 + \pi, \omega_2 + \pi)\right]. \tag{2.26}$$

2.3 Two-dimensional Filter Banks

2.3.1 Perfect Reconstruction Filter Banks

A filter bank splits a signal into a number of frequency bands. For a two-channel filter bank, such as the one shown in Figure 2.2, the spectrum of the input image is split in two parts, low-pass and high-pass frequency band.

In the two-channel filter bank the index 0 denotes low-pass filters and index 1 denotes high-pass filters. Hence, H_0 and G_0 are analysis and synthesis low-pass filters and H_1 and G_1 are analysis and synthesis high-pass filters respectively.

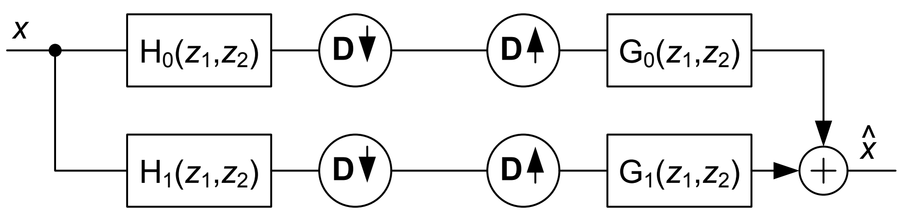


Figure 2.2: Two-channel critically sampled quincunx filter bank. The dilation matrix $\mathbf{D} = \mathbf{D}_q$. The number of channels is defined with $|\det(\mathbf{D}_q)| = 2$.

An important requirement imposed on filter banks is to enable perfect reconstruction (PR) of the analyzed signal [Strang 96], i.e. the output has to be an exact copy of the input.

Downsampling and upsampling operators are elementary building blocks of a critically sampled filter bank. A critically sampled filter bank does not introduce redundancy in the signal description while at the same time preserves the perfect reconstruction property. In the case of quincunx sampling, the subsampling by a factor of two will be performed based on the checkerboard quincunx lattice. In general, the critically sampled filter bank will have exactly $|\det(D)|$ channels which split a signal into $|\det(D)|$ appropriate frequency bands. For example, the separable sampling matrix from equation 2.7 will lead to the 4-channel critically sampled filter bank as shown in Figure 2.3.

The analysis and synthesis filters are selected in such a way that the PR requirement is satifsied [Viscito 91]. The output of the quincunx filter bank from Figure 2.2 can be expressed using equation 2.25 as

$$\begin{aligned}\hat{X}(\mathbf{z}) &= \frac{1}{2}[H_0(\mathbf{z})G_0(\mathbf{z}) + H_1(\mathbf{z})G_1(\mathbf{z})]X(\mathbf{z}) \\ &+ \frac{1}{2}[H_0(-\mathbf{z})G_0(\mathbf{z}) + H_1(-\mathbf{z})G_1(\mathbf{z})]X(-\mathbf{z}).\end{aligned} \tag{2.27}$$

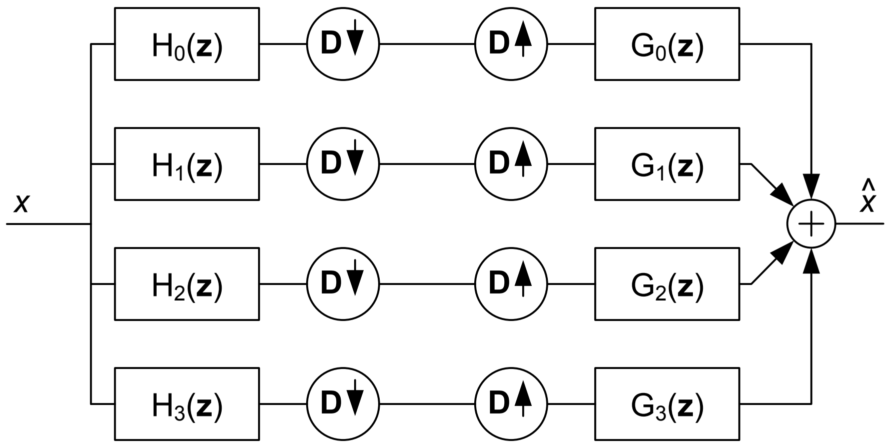


Figure 2.3: Four-channel critically sampled filter bank. The number of channels is defined with $|\det(\mathbf{D})| = 4$.

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

$$H_0(-\mathbf{z})G_0(\mathbf{z}) + H_1(-\mathbf{z})G_1(\mathbf{z}) = 0 \tag{2.28a}$$

$$H_0(\mathbf{z})G_0(\mathbf{z}) + H_1(\mathbf{z})G_1(\mathbf{z}) = 2, \tag{2.28b}$$

or stated in the Fourier domain:

$$H_0(\boldsymbol{\omega} + \boldsymbol{\pi})G_0(\boldsymbol{\omega}) + H_1(\boldsymbol{\omega} + \boldsymbol{\pi})G_1(\boldsymbol{\omega}) = 0 \tag{2.29a}$$

$$H_0(\boldsymbol{\omega})G_0(\boldsymbol{\omega}) + H_1(\boldsymbol{\omega})G_1(\boldsymbol{\omega}) = 2, \tag{2.29b}$$

where $\boldsymbol{\pi} = [\pi\ \pi]^T$.

The filter bank structures in Figures 2.2 and 2.3 with filtering followed by downsampling are far from being computationally efficient. A more efficient structure can be obtained by performing downsampling prior to filtering. This is the well-known polyphase filter bank implementation [Vaidyanathan 92]. Also, on the synthesis side performing upsampling after filtering will lead to more efficient filter bank structures. The exchange of filters and sampling rate converters is known as *First and Second Noble identities* [Vaidyanathan 92].

2.3.2 The Lifting Scheme

The lifting scheme is an efficient tool for building PR filter banks. It was proposed by Sweldens [Sweldens 95] and it became an important tool for building biorthogonal wavelet filter banks [Sweldens 96a] and more generally a tool for building second generation wavelets [Sweldens 97].

The lifting scheme leads to efficient filter bank implementations. It relies on the polyphase decomposition of the input signal. The signal/image at the filter bank input is being split into $M = |\det(\mathbf{D})|$ phases. For the quincunx case, samples from the first quincunx phase will go in the first channel and samples from the second quincunx phase will go in the second channel.

As shown in Figure 2.5, the polyphase decomposition is followed by the prediction step. The filter P predicts the values of pixels from the second phase based on the pixels from the first phase. The prediction error is calculated with the difference operator leading to the output of the high-pass channel. The update step which follows creates the low-pass output of the filter bank.

The reconstruction is readily obtainable. The synthesis filter bank is constructed using the same steps but in the opposite order from that on the analysis side and with the opposite signs. This way, the operations from the analysis side have been reversed and cancelled on the synthesis side.

The analyisis of the lifting scheme is usually performed in the polyphase domain, i.e. every filter is expressed as a sum of its polyhpase components.

The analysis polyphase matrix of the filter bank from Figure 2.5 [Kovacevic 00] is:

$$\mathbf{H}_p(\mathbf{z}) = \begin{bmatrix} H_{0e}(\mathbf{z}) & H_{0o}(\mathbf{z}) \\ H_{1e}(\mathbf{z}) & H_{1o}(\mathbf{z}) \end{bmatrix} = \begin{bmatrix} 1 & U(\mathbf{z}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -P(\mathbf{z}) & 1 \end{bmatrix} = \begin{bmatrix} 1 - U(\mathbf{z})P(\mathbf{z}) & U(\mathbf{z}) \\ -P(\mathbf{z}) & 1 \end{bmatrix}, \tag{2.30}$$

which can be written as

$$H_0(\mathbf{z}) = 1 - P(\mathbf{z}^{\mathbf{D}})U(\mathbf{z}^{\mathbf{D}}) + \mathbf{z}^{-\mathbf{t}}U(\mathbf{z}^{\mathbf{D}}), \tag{2.31a}$$

$$H_1(\mathbf{z}) = -P(\mathbf{z}^{\mathbf{D}}) + \mathbf{z}^{-\mathbf{t}}, \tag{2.31b}$$

where $t = [1 \;\; 0]^T$. The synthesis polyphase matrix follows directly from analysis polyphase matrix:

$$\mathbf{G}_p(\mathbf{z}) = \begin{bmatrix} G_{0e}(\mathbf{z}) & G_{0o}(\mathbf{z}) \\ G_{1e}(\mathbf{z}) & G_{1o}(\mathbf{z}) \end{bmatrix} = \mathbf{H}_p^{*-1}(\mathbf{z}) = \begin{bmatrix} 1 & P^*(\mathbf{z}) \\ -U^*(\mathbf{z}) & 1 - U^*(\mathbf{z})P^*(\mathbf{z}) \end{bmatrix}. \tag{2.32}$$

Therefore,

$$G_0(\mathbf{z}) = 1 + \mathbf{z}^{\mathbf{t}}P^*(\mathbf{z}^{\mathbf{D}}), \tag{2.33a}$$

$$G_1(\mathbf{z}) = -U^*(\mathbf{z}^{\mathbf{D}}) + \mathbf{z}^{\mathbf{t}}(1 - P^*(\mathbf{z}^{\mathbf{D}})U^*(\mathbf{z}^{\mathbf{D}})). \tag{2.33b}$$

2.3.3 Quincunx Interpolating Filter Banks

It this section we introduce filters that will be used in the rest of the thesis. They are proposed by Kovacevic and Sweldens. In [Kovacevic 00] a method is proposed, which allows to build perfect reconstruction filter banks for any lattice and in any dimension. The construction is based on the lifting scheme. We use these results to build the quincunx filter bank shown in Figure 2.5.

The proposed construction leads to interpolating synthesis low-pass filter. Interpolating means that the impulse response of the filter has zero values in all locations of its first phase except at the origin where it has value of 1. Therefore, applying an interpolating filter on a single image pixel leaves the very pixel unchanged and influences the positions that do not belong to the phase to which the pixel itself belongs.

The filters used to build the predict and update lifting steps are called Neville filters. They are based on the polynomial interpolation. To understand the role of Neville filters, let us consider a polynomial of degree lower than N and the Neville filter P of degree N or higher. Filtering the polynomial with the Neville filter P gives the samples of the same polynomial but sampled on the lattice shifted by $\boldsymbol{\tau}$. Following the notation from [Kovacevic 00], we define π to be a multivariate polynomial, and $\pi(\mathcal{Z}^2)$ the sequence formed by evaluating the polynomial on the lattice $\mathcal{Z}^2$. Then, applying the Neville filter P on the sampled polynomial yields

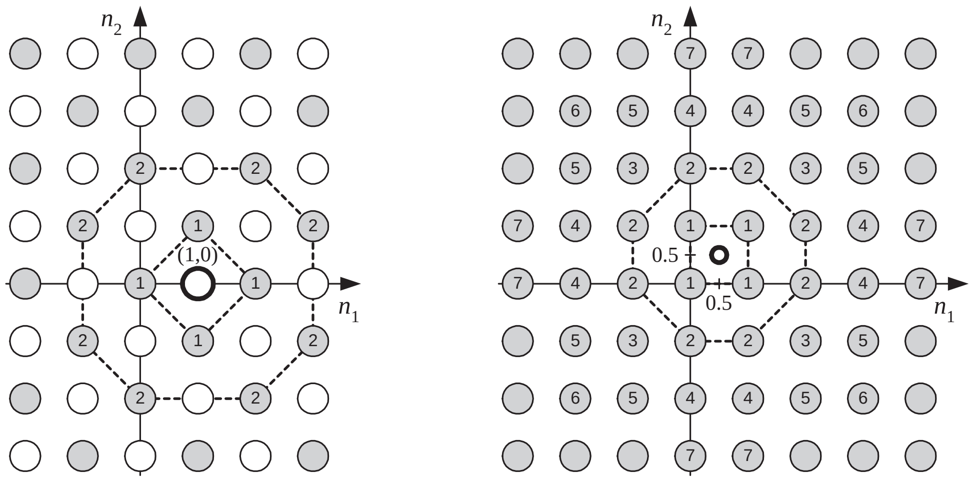
$$P\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \boldsymbol{\tau}) \tag{2.34}$$

where $\boldsymbol{\tau} \in \mathcal{R}^2$.

The construction of such filters relies on solving the interpolation problem. The solution of the interpolation problem in multiple dimensions is given by de Boor and Ron in [de Boor 90, de Boor 92]. In [Kovacevic 00] the authors give Neville filter coefficients for the quincunx case obtained by using de Boor-Ron algorithm. The predict filter coefficients are shown in Table 2.1. Higher order prediction filters use more coefficients for the interpolation. The coefficients are grouped into rings with same values.

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 2.1: Quincunx Neville 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 shown in Figure 2.4.



(a) Interpolating rings on the quincunx lattice.

(b) Interpolating rings in the sampled domain.

Figure 2.4: Rings for the quincunx interpolation. Samples are marked with the numbers of rings they belong to. 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 (gray circles).

Vanishing Moments

An analysis filter bank has $\tilde{N}$ vanishing moments if its high-pass channel annihilates polynomial sequences of degree lower than $\tilde{N}$. Also, a synthesis filter bank has N vanishing moments if its synthesis high-pass channel annihilates polynomial sequences of degree lower than N. The vanishing moments of the analysis filter bank are called dual (DM), and the vanishing moments of the synthesis filter bank are called primal (PM).

In order to have $\tilde{N}$ dual vanishing moments, P has to be a Neville filter of the order $\tilde{N}$ and shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$ [Kovacevic 00]. In that case the prediction of a sampled polynomial will be perfect, and the resulting high-pass output, calculated as a prediction error will be equal to zero.

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 $\boldsymbol{\tau}$ [Kovacevic 00]. Consequently, $2U$ has to be a Neville filter of the order N and shift $-\boldsymbol{\tau}$ [Kovacevic 00]. The simplest choice is to make

$$U = U_N = \frac{1}{2} P_N^*, \tag{2.35}$$

where P_N is a Neville filter of the order N and shift $\boldsymbol{\tau}$ while U_N per se in not a Neville filter.

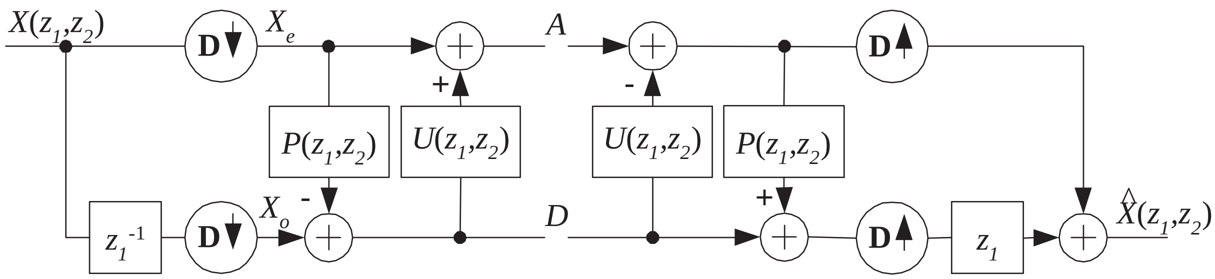


Figure 2.5: Quincunx interpolating filter bank. Downsampling and upsampling operators are defined with a quincunx dilation matrix, $\mathbf{D} = \mathbf{D}_q$.

It is important to note that the prediction filter by itself defines the number of dual vanishing moments and the update filter influences only the number of primal vanishing moments as long as $\tilde{N} \geq N$ [Kovacevic 00].

For example, the filter bank with 2 dual vanishing moments and 2 primal vanishing moments will be constructed using the prediction filter P_2 and the update filter $U_2 = 1/2P_2^*$ (see Table 2.1):

$$P_2(z_1, z_2) = \frac{1}{4}(1 + z_1^{-1} + z_2^{-1} + z_1^{-1}z_2^{-1}), \tag{2.36}$$

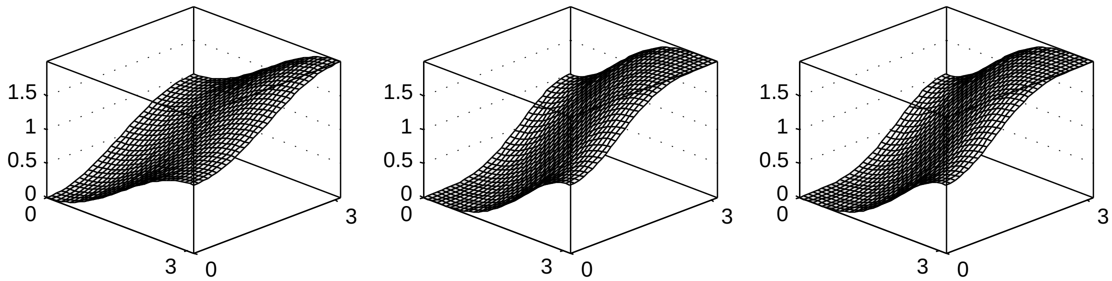
$$U_2(z_1, z_2) = \frac{1}{2}P_2^*(z_1, z_2) = \frac{1}{8}(1 + z_1 + z_2 + z_1 z_2). \tag{2.37}$$

In order to have more dual vanishing moments, e.g. 4, one should use the P_4 prediction filter:

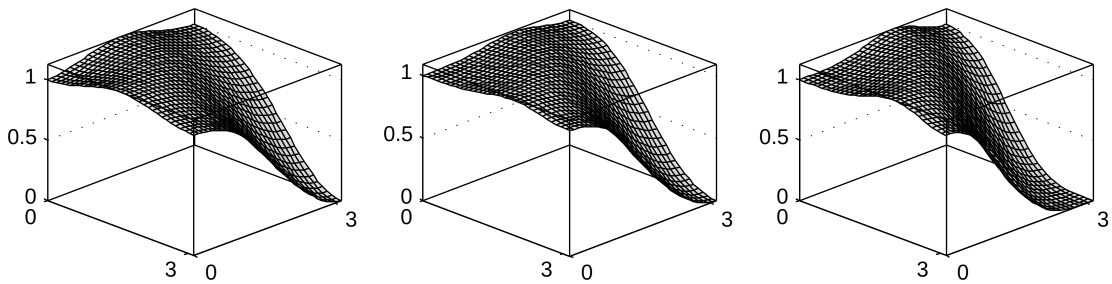
$$\begin{aligned} P_4(z_1, z_2) =& \frac{10}{32}(1 + z_1^{-1} + z_2^{-1} + z_1^{-1}z_2^{-1}) \\ &- \frac{1}{32}(z_1^{-2} + z_2^{-2} + z_1^{-2}z_2^{-1} + z_1^{-1}z_2^{-2} + z_1 + z_2 + z_1 z_2^{-1} + z_1^{-1}z_2), \end{aligned} \tag{2.38}$$

$$U_2(z_1, z_2) = \frac{1}{2}P_2^*(z_1, z_2) = \frac{1}{8}(1 + z_1 + z_2 + z_1 z_2). \tag{2.39}$$

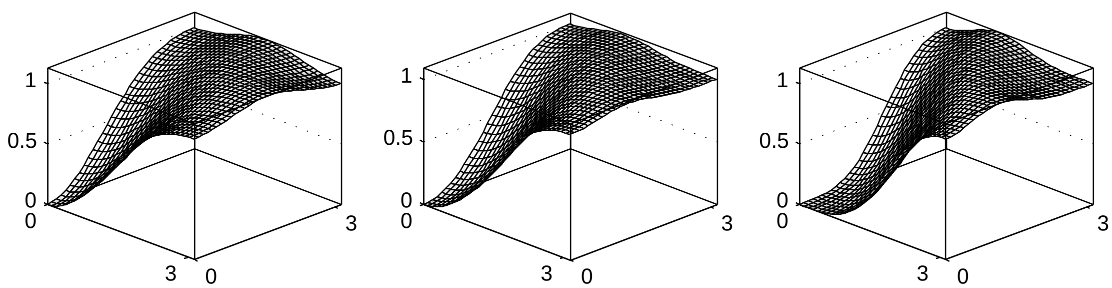
The analysis and synthesis filter for various combinations of primal and dual vanishing moments are shown in Figure 2.6.



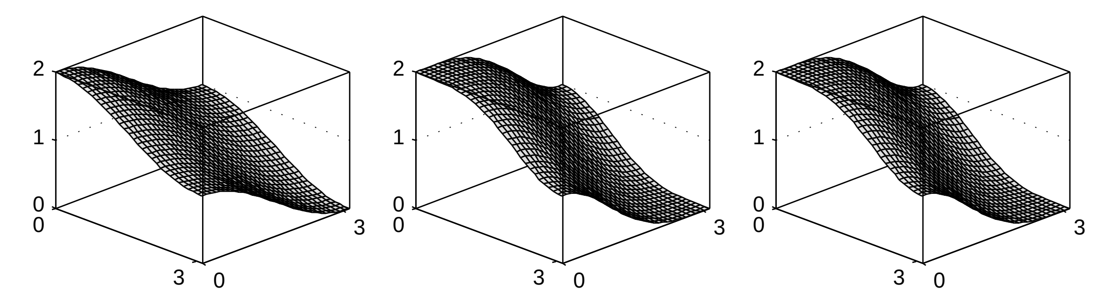
(a) Analysis HP for P_2U_2. (b) Analysis HP for P_4U_2. (c) Analysis HP for P_4U_4.



(d) Analysis LP for P_2U_2. (e) Analysis LP for P_4U_2. (f) Analysis LP for P_4U_4.



(g) Synthesis HP for P_2U_2. (h) Synthesis HP for P_4U_2. (i) Synthesis HP for P_4U_4.



(j) Synthesis LP for P_2U_2. (k) Synthesis LP for P_4U_2. (l) Synthesis LP for P_4U_4.

Figure 2.6: Magnitude frequency responses for analysis and synthesis filter banks with spatial frequencies ω_1 and ω_2 in the range from 0 to π. Every figure column represents one combination of P and U filters.

2.4 Wavelets and Filter Banks

2.4.1 Multiresolution Analysis

In this section we give a brief overview of the multiresolution analysis (MRA) by generalizing its concepts in two dimensions. For a detailed treatment of multiresolution analysis see [Vetterli 95].

The MRA of $L_2(\mathbf{R}^2)$ is based on a sequence of embedded closed subspaces

$$\{0\} \rightarrow \ldots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \ldots \rightarrow L^2(\mathbf{R}^2) \tag{2.40}$$

satisfying the following properties:

Scale invariance

$$f(\mathbf{t}) \in V_j \Longleftrightarrow f(\mathbf{D}\mathbf{t}) \in V_{j-1}. \tag{2.41}$$

or stated in another way

$$f(\mathbf{t}) \in V_m \Longleftrightarrow f(\mathbf{D}^m\mathbf{t}) \in V_0. \tag{2.42}$$

Shift invariance

$$f(\mathbf{t}) \in V_0 \Longrightarrow f(\mathbf{t}-\mathbf{n}) \in V_0, \text{ for all } \mathbf{n} \in \mathbf{Z}^2. \tag{2.43}$$

Existence of a basis

There exists a function $\varphi(\mathbf{t})$ in V_0 such that

$$\{\varphi(\mathbf{t}-\mathbf{n}) \mid \mathbf{n} \in \mathbf{Z}^2\} \tag{2.44}$$

is an orthonormal basis for V_0. The function $\varphi(\mathbf{t})$ is called a scaling function. From the above properties, an orthonormal basis for V_{-j} is

$$\{M^{j/2}\varphi(\mathbf{D}^j\mathbf{t}-\mathbf{n}) \mid \mathbf{n} \in \mathbf{Z}^2\}, \tag{2.45}$$

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

There are additional requirements imposed on the dilation matrix $\mathbf{D}$. Firsty, it needs to have integer entries. Secondly, its eigenvalues λ_1 and λ_2 must satisfy

$$|\lambda_m| > 1, \tag{2.46}$$

in order to have dilation in both dimensions. Therefore, the approximation will get finer in every direction as j decreases.

For the case of quincunx sampling $|\lambda_1| = |\lambda_2| = \sqrt{2}$. For separable sampling $|\lambda_1| = |\lambda_2| = 2$. So, the advantage of quincunx sampling is that its scale change is more gradual, by a factor $\sqrt{2}$, twice smaller than for the separable case.

The number of wavelets is obtained as $|\det(\mathbf{D})| - 1$. So, for the quincunx case there will be one mother wavelets while for the separable case there will be three mother wavelets.

Based on the embedding of spaces and the scaling property the *dilation equation* or a *two scale equation* [Vetterli 95] for two-dimensional case can be written as

$$\varphi(\mathbf{t}) = \sum_{\mathbf{n}\in\mathbb{Z}^2} g_0[\mathbf{n}]\sqrt{M}\varphi(\mathbf{D}\mathbf{t} - \mathbf{n}). \tag{2.47}$$

Complementary to approximation spaces V_j, there are detail spaces W_j such that

$$V_j \bigoplus W_j = V_{j-1}. \tag{2.48}$$

Wavelets as basis function for those detail spaces can be expressed in terms of scaling function using refinement relation

$$\psi(\mathbf{t}) = \sum_{\mathbf{n}\in\mathbb{Z}^2} g_1[\mathbf{n}]\sqrt{M}\varphi(\mathbf{D}\mathbf{t} - \mathbf{n}). \tag{2.49}$$

The coefficients $g_0[\mathbf{n}]$ and $g_1[\mathbf{n}]$ correspond to the low-pass and high-pass filters of the synthesis filter bank [Vetterli 95].

2.4.2 Biorthogonality

In the thesis we will consider biorthogonal wavelet filter banks. In the case of biorthogonality there also exist dual scaling function $\tilde{\varphi}$ and M dual wavelet functions $\tilde{\psi}_i$. Dual functions will be the ones on the analysis side and the primal functions are the ones from the synthesis side. Dual and primal scaling functions are orthogonal to their integer translates:

$$\langle \varphi(\mathbf{t}), \tilde{\varphi}(\mathbf{t} - \mathbf{n}) \rangle = \delta[\mathbf{n}]. \tag{2.50}$$

The same holds for all scales:

$$\langle \varphi(\mathbf{D}^j\mathbf{t} - \mathbf{n}), \tilde{\varphi}(\mathbf{D}^j\mathbf{t} - \mathbf{n}') \rangle = \langle \varphi_{-j,\mathbf{n}}(\mathbf{t}), \tilde{\varphi}_{-j,\mathbf{n}'}(\mathbf{t}) \rangle = M^{-j}\delta[\mathbf{n} - \mathbf{n}']. \tag{2.51}$$

Also, the M primal and dual wavelets are orthogonal to each other and their integer translates.

$$\begin{aligned} \langle \psi_i(\mathbf{D}^j\mathbf{t} - \mathbf{n}), \tilde{\psi}_{i'}(\mathbf{D}^j\mathbf{t} - \mathbf{n}') \rangle &= \langle \psi_{i;-j,\mathbf{n}}(\mathbf{t}), \tilde{\psi}_{i';-j,\mathbf{n}'}(\mathbf{t}) \rangle \\ &= M^{-j}\delta[i - i']\delta[\mathbf{n} - \mathbf{n}']. \end{aligned} \tag{2.52}$$

Similarly, the scaling function is orthogonal to each of the M dual wavelets:

$$\langle \varphi(\mathbf{t}), \tilde{\psi}_i(\mathbf{t}-\mathbf{n})\rangle = 0. \tag{2.53}$$

wavelets and dual wavelets are orthogonal across scales [Kovacevic 92, Vetterli 95]

$$\begin{aligned} \langle \psi_i(\mathbf{D}^j\mathbf{t}-\mathbf{n}), \tilde{\psi}_{i'}(\mathbf{D}^{j'}\mathbf{t}-\mathbf{n}')\rangle &= \langle \psi_{i;-j,\mathbf{n}}(\mathbf{t}), \tilde{\psi}_{i';-j',\mathbf{n}'}(\mathbf{t})\rangle \\ &= M^{-j-j'}\delta[i-i']\delta[j-j']\delta[\mathbf{n}-\mathbf{n}']. \end{aligned} \tag{2.54}$$

The image representation can be obtained using wavelet functions from all scales as

$$x(\mathbf{t}) = \sum_{j\in\mathbb{Z}}\sum_{\mathbf{n}\in\mathbb{Z}^2}\sum_{i=1}^{M-1}\langle\tilde{\psi}_{i;j,\mathbf{n}}, x\rangle\psi_{i;j,\mathbf{n}}(\mathbf{t}) \tag{2.55}$$

where for the quincunx case the innermost summation over all wavelets does not exist since there is only one wavelet function. In practice, there is only a finite number of scales used so an image representation is obtained by using a number od wavelet functions and the coarsest scale function

$$x(\mathbf{t}) = \sum_{j=1}^{J}\sum_{\mathbf{n}\in\mathbb{Z}^2}\sum_{i=1}^{M-1}\langle\tilde{\psi}_{i;j,\mathbf{n}}, x\rangle\psi_{i;j,\mathbf{n}}(\mathbf{t}) + \sum_{\mathbf{n}\in\mathbb{Z}^2}\langle\tilde{\varphi}_{J,\mathbf{n}}, x\rangle\varphi_{J,\mathbf{n}}(\mathbf{t}) \tag{2.56}$$

2.4.3 Iterated Quincunx Filter Banks

The construction of wavelets and scaling functions can be obtained by using the filter bank structure iterated on the low-pass channel as shown in Figure 2.8(a). The scaling function is connected with the impulse response of the low-pass channel, and the wavelet function is connected with the impulse response of the high-pass channel. In the z-domain the equivalent low-pass and high-pass filters after N iterations are:

$$G_0^{(N)}(\mathbf{z}) = \prod_{i=0}^{N-1} G_0(\mathbf{z}^{\mathbf{D}^i}), \tag{2.57}$$

$$G_1^{(N)}(\mathbf{z}) = G_1(\mathbf{z}^{\mathbf{D}^{N-1}})\prod_{i=0}^{N-2} G_0(\mathbf{z}^{\mathbf{D}^i}). \tag{2.58}$$

Frequency response magnitudes of the corresponding filters are shown in Figures 2.9, 2.10, 2.11, and 2.12.

Continuous-time functions associated with these discrete filters' impulse responses are [Kovacevic 95]:

$$\varphi^{(N)}(\mathbf{t}) = 2^{\frac{N}{2}} g_0^{(N)}[\mathbf{n}], \qquad (\mathbf{D}^T)^N\mathbf{t} \in \mathbf{n} + [0,1)^2, \tag{2.59}$$

$$\psi^{(N)}(\mathbf{t}) = 2^{\frac{N}{2}} g_1^{(N)}[\mathbf{n}], \qquad (\mathbf{D}^T)^N \mathbf{t} \in \mathbf{n} + [0,1)^2. \tag{2.60}$$

Underlying scaling and wavelet functions are obtained as

$$\varphi(\mathbf{t}) = \lim_{N\to\infty} \varphi^{(N)}(\mathbf{t}), \tag{2.61}$$

$$\psi(\mathbf{t}) = \lim_{N\to\infty} \psi^{(N)}(\mathbf{t}). \tag{2.62}$$

Necessary (but not sufficient) conditions for the existence of the limit is a zero of a low-pass filter at aliasing frequencies, i.e.

$$G_0(-1,-1) = 0. \tag{2.63}$$

In case of a biorthogonal filter bank the dual scale and wavelet functions $\tilde{\psi}$, $\tilde{\varphi}$ are obtained in the same way by using iterated analysis filers $H_0^{(N)}$ and $H_1^{(N)}$ (see Figure 2.7).

Wavelet Series From Iterated Filter Banks

The famous *Mallat's algorithm* [Mallat 89b] allows to calculate wavelet series of a continuous-time signal in an efficient way by using an iterated discrete-time filter bank structure. In this way, the wavelet series coefficients are calculated without calculating continuous-time integrals. The only assumption made is that the analyzed continuous time image belongs to V_0

$$x(\mathbf{t}) \in V_0 \tag{2.64}$$

A discrete signal that is an input to a filter bank should represent coefficients of the projection onto V_0, i.e

$$\langle \tilde{\varphi}(\mathbf{t}-\mathbf{n}), x(\mathbf{t}) \rangle \tag{2.65}$$

The coefficients of the projections onto V_1, W_1, V_2, W_2... will be obtained iteratively by following the filter bank structure as shown on Figure 2.7(a). Approximation coefficients $a^{(j)}$ shown in the figure represent coefficients of projection of the analyzed signal onto V_j since

$$a^{(j)}[\mathbf{n}] = \langle \tilde{\varphi}_{j,\mathbf{n}}, x(\mathbf{t}) \rangle. \tag{2.66}$$

Detail coefficients $d^{(j)}$ represent coefficients of projection of the analyzed signal onto W_j:

$$d^{(j)}[\mathbf{n}] = \langle \tilde{\psi}_{j,\mathbf{n}}, x(\mathbf{t}) \rangle. \tag{2.67}$$

To further simplify the calculation of the wavelet series, the discrete-time input signal to the filter bank will simply be the sampled continuous-time signal. This is possible if the resolution corresponding to V_0 is fine enough when compared to the continuous-time input signal [Vetterli 95]. Then, the projection coefficients onto V_0 can be approximated as

$$\langle \tilde{\varphi}(\mathbf{t}-\mathbf{n}), x(\mathbf{t}) \rangle \simeq x(\mathbf{n}) \tag{2.68}$$

since the integral of $\tilde{\varphi}(\mathbf{t}) = 1$.

2.4.4 Vanisihing Moments

Moments of the primal and dual wavelet and scaling functions are defined as

$$m_p = \int \mathbf{t}^p \varphi(\mathbf{t}) d\mathbf{t} \qquad \tilde{m}_p = \int \mathbf{t}^p \tilde{\varphi}(\mathbf{t}) d\mathbf{t} \tag{2.69a}$$

$$n_p = \int \mathbf{t}^p \psi(\mathbf{t}) d\mathbf{t} \qquad \tilde{n}_p = \int \mathbf{t}^p \tilde{\psi}(\mathbf{t}) d\mathbf{t} \tag{2.69b}$$

where $p \in \mathbf{N}$. The scaling functions are normalized in order to have

$$m_0 = \int \varphi(\mathbf{t}) d\mathbf{t} = 1 \tag{2.70a}$$

$$\tilde{m}_0 = \int \tilde{\varphi}(\mathbf{t}) d\mathbf{t} = 1 \tag{2.70b}$$

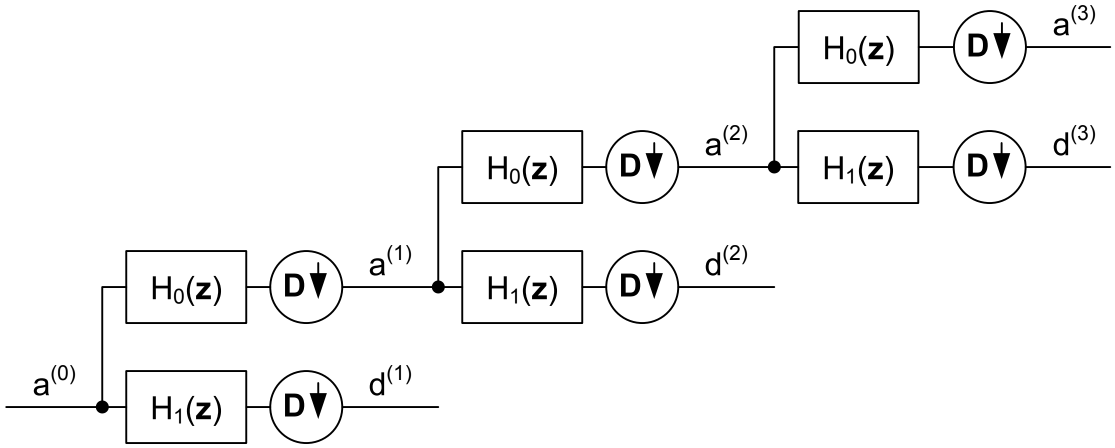
If the following equation holds

$$\int \mathbf{t}^p \tilde{\psi}(\mathbf{t}) d\mathbf{t} = 0 \text{ for } |p| < \tilde{N} \tag{2.71}$$

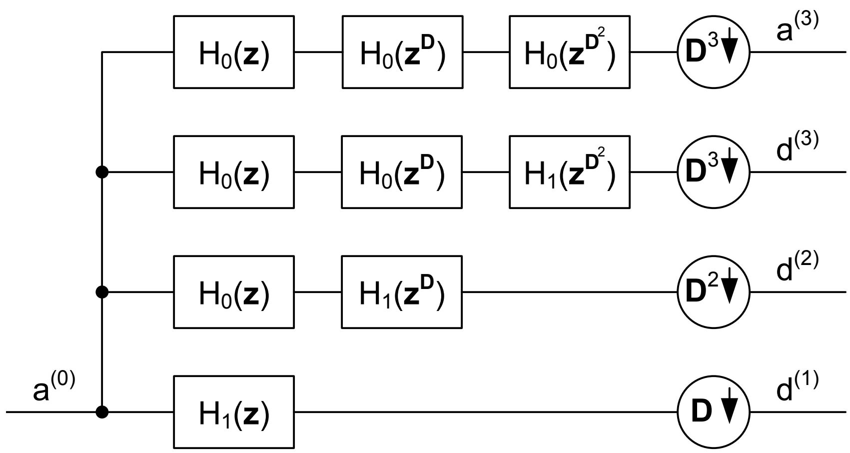
then it is said that a wavelet has $\tilde{N}$ vanishing moments. This is equivalent to the primal scaling function being able to reproduce polynomials of degree lower than $\tilde{N}$. In the same manner, if the primal wavelet has N vanishing moments then a dual scaling function can reproduce polynomials up to degree N [Burrus 97]. In signal and image processing applications it is in general desirable to have more vanishing moments. Usually, the primal vanishing moments are more important than dual vanising moments since they are related to the smoothness of the reconstruction functions.

These requirements on wavelets can be easily translated to filters present in refinement relations leading to vanishing moments of discrete filters as stated in section 2.3.3.

The limit wavelet and scale function obtained for different choices of primal and dual vanishing moments are shown in Figure 2.13.



(a)



(b)

Figure 2.7: (a) Analysis filter bank iterated on the low-pass filter channel results in a wavelet series expansion. (b) Equivalent structure obtained by using the interchange of downsampling and filtering called *First Noble Identity*.

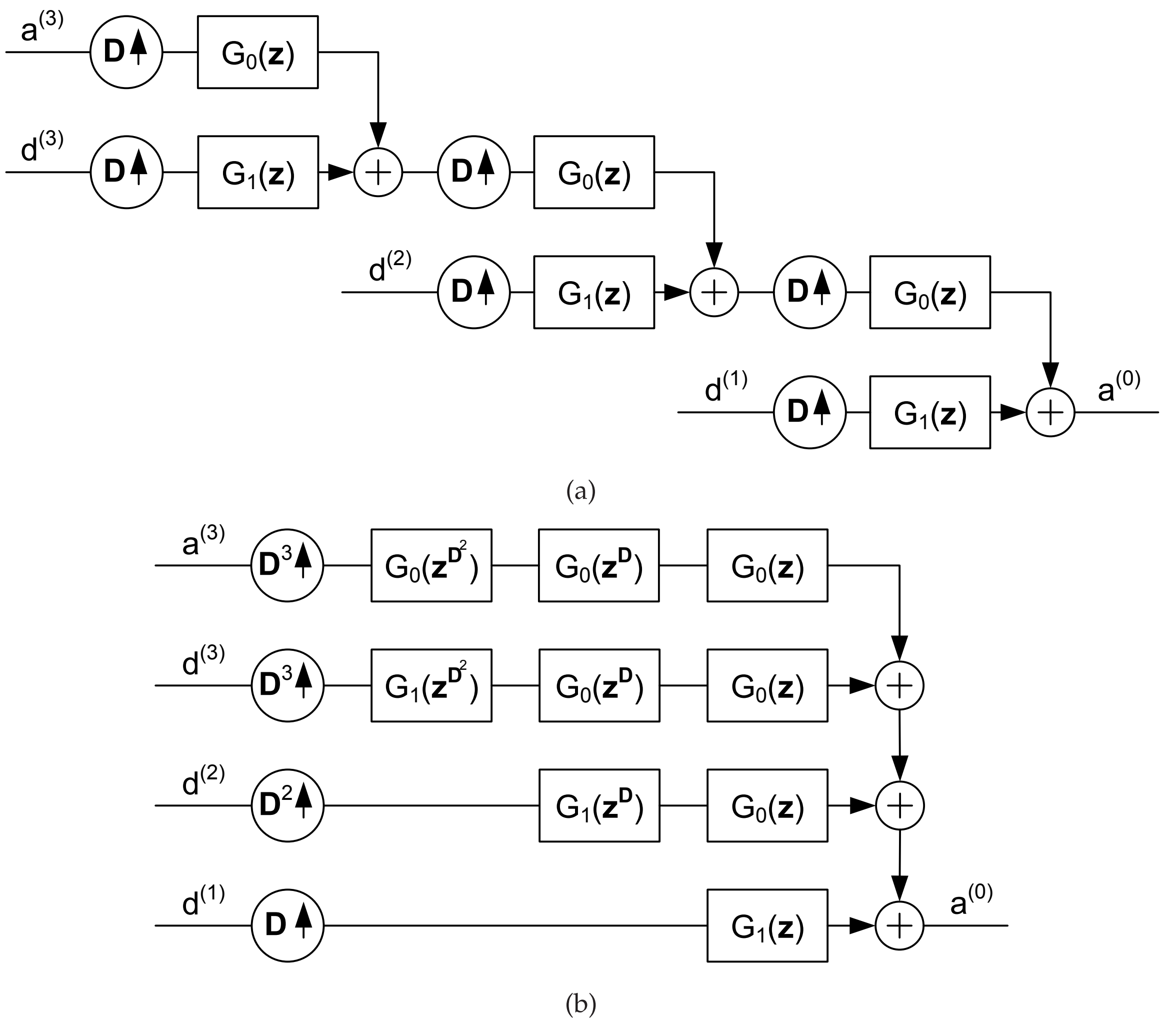
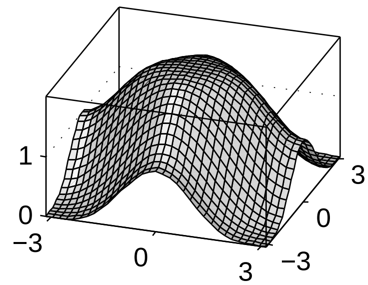
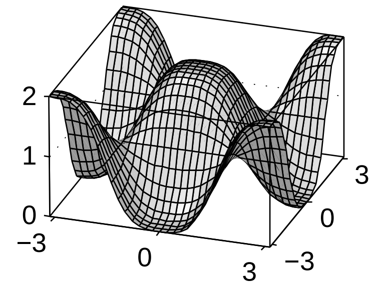


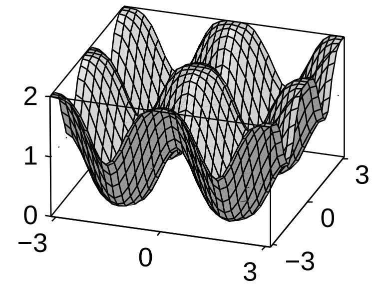
Figure 2.8: (a) Synthesis filter bank iterated on the low-pass filter channel. (b) Equivalent synthesis structure obtained by using the interchange of upsampling and filtering called *Second Noble Identity*.



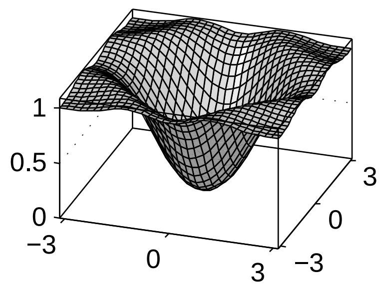
(a) $G_0(\mathbf{z})$



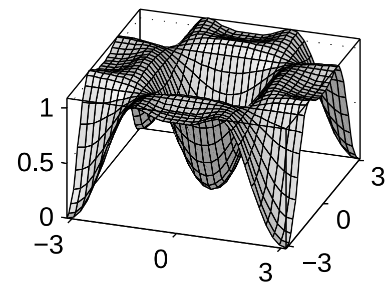
(b) $G_0(\mathbf{z}^{\mathbf{D}})$



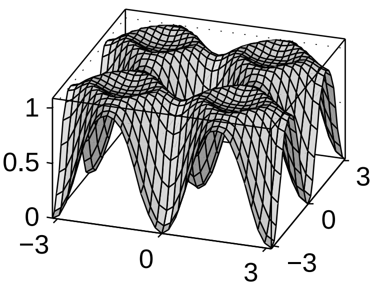
(c) $G_0(\mathbf{z}^{\mathbf{D}^2})$



(d) $G_1(\mathbf{z})$

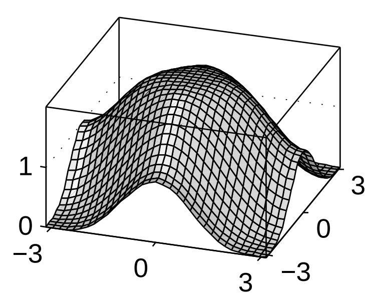


(e) $G_1(\mathbf{z}^{\mathbf{D}})$

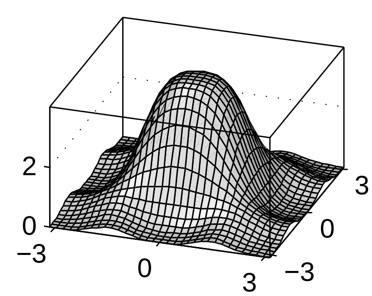


(f) $G_1(\mathbf{z}^{\mathbf{D}^2})$

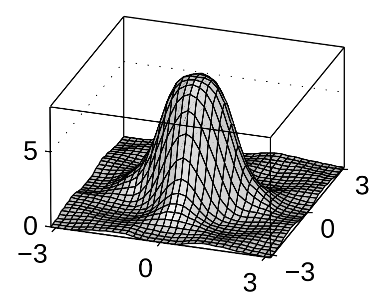
Figure 2.9: Frequency response magnitudes of the upsampled low-pass and high-pass synthesis filters obtained with P_4 and U_2 lifting steps with spatial frequencies ω_1 and ω_2 in the range from $-\pi$ to π.



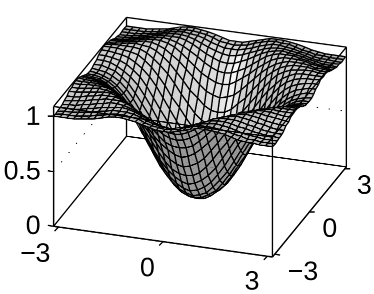
(a) $G_0^{(1)}(\mathbf{z})$



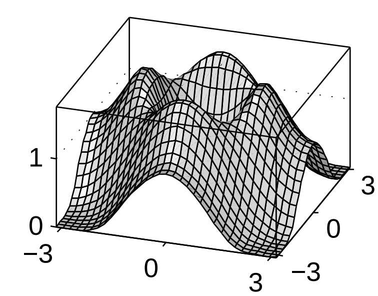
(b) $G_0^{(2)}(\mathbf{z})$



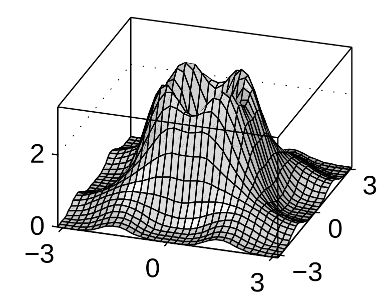
(c) $G_0^{(3)}(\mathbf{z})$



(d) $G_1^{(1)}(\mathbf{z})$

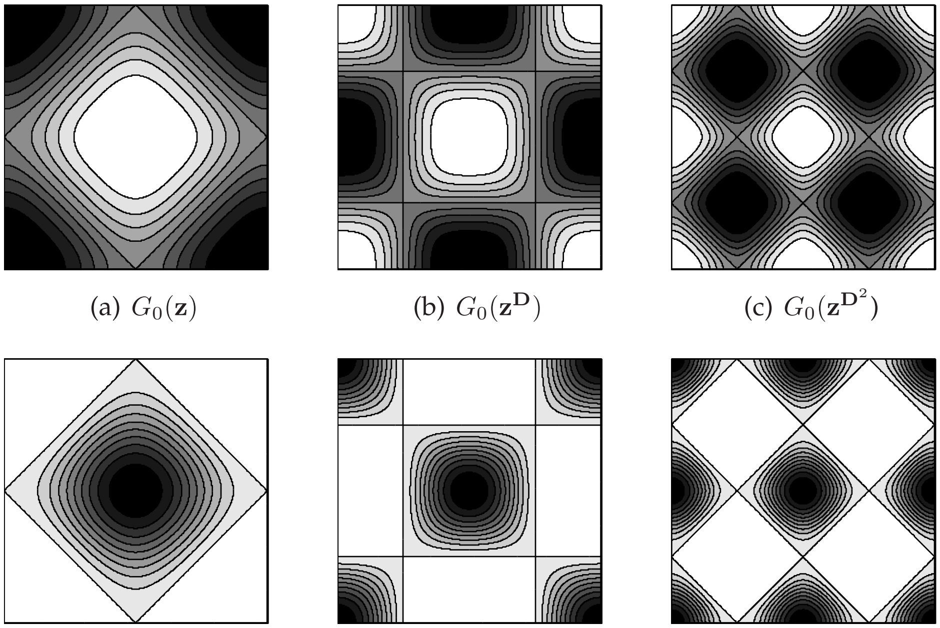


(e) $G_1^{(2)}(\mathbf{z})$



(f) $G_1^{(3)}(\mathbf{z})$

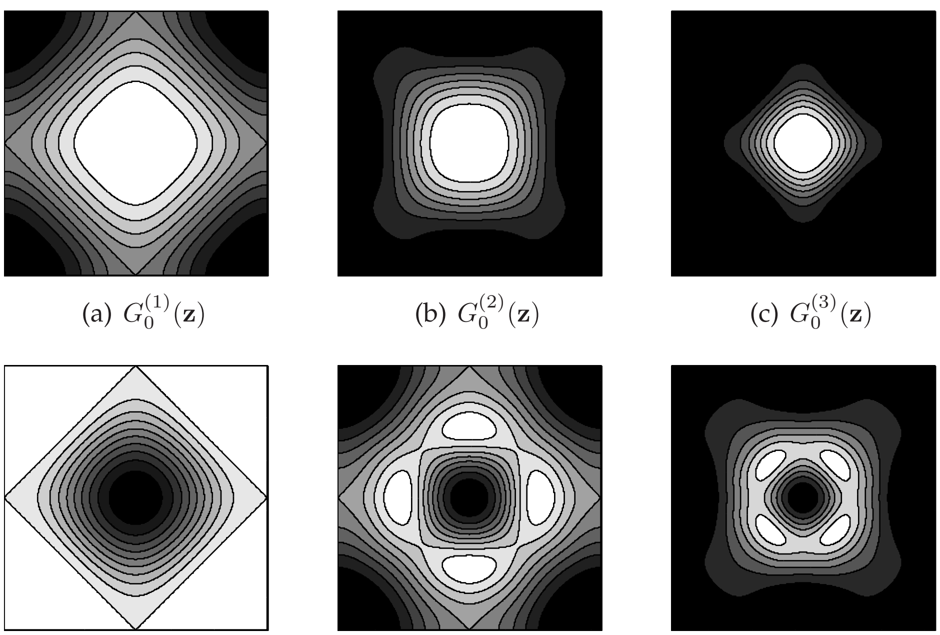
Figure 2.10: Frequency response magnitudes of the iterated low-pass and high-pass synthesis filters obtained with P_4 and U_2 lifting steps.



(a) $G_0(\mathbf{z})$ (b) $G_0(\mathbf{z}^{\mathbf{D}})$ (c) $G_0(\mathbf{z}^{\mathbf{D}^2})$

(d) $G_1(\mathbf{z})$ (e) $G_1(\mathbf{z}^{\mathbf{D}})$ (f) $G_1(\mathbf{z}^{\mathbf{D}^2})$

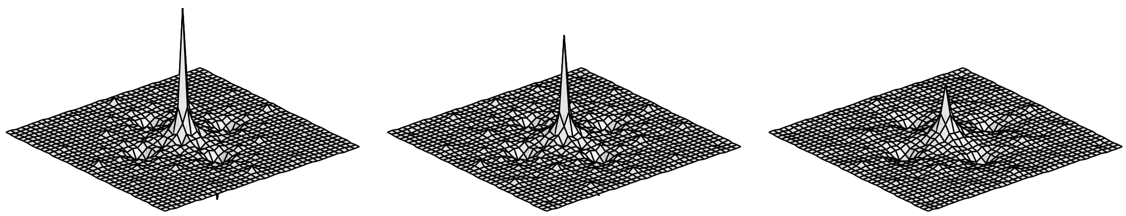
Figure 2.11: Contour plots for frequency response magnitudes of the upsampled low-pass and high-pass synthesis filters from Figure 2.9.



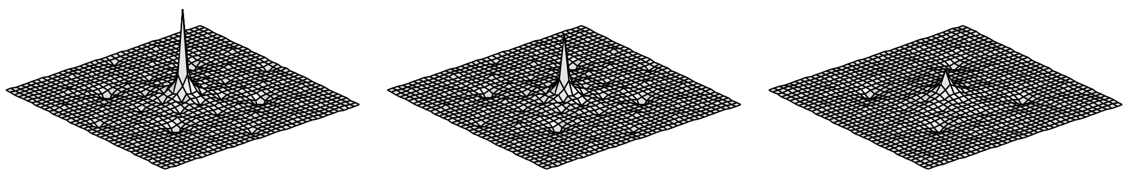
(a) $G_0^{(1)}(\mathbf{z})$ (b) $G_0^{(2)}(\mathbf{z})$ (c) $G_0^{(3)}(\mathbf{z})$

(d) $G_1^{(1)}(\mathbf{z})$ (e) $G_1^{(2)}(\mathbf{z})$ (f) $G_1^{(3)}(\mathbf{z})$

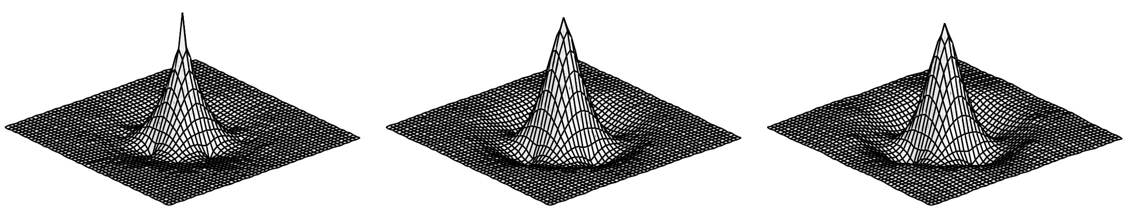
Figure 2.12: Contour plots for frequency response magnitudes of the iterated low-pass and high-pass synthesis filters from Figure 2.10.



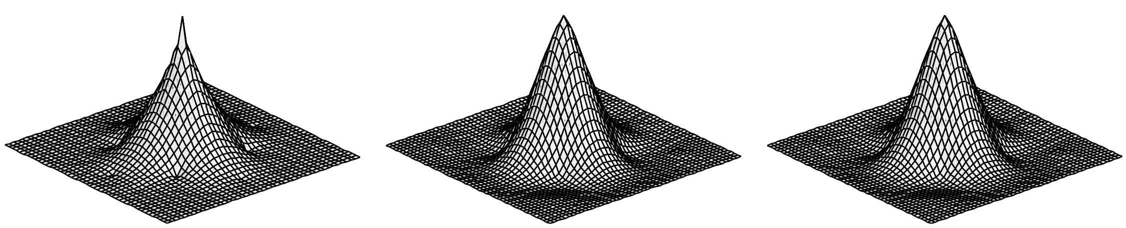
(a) Analysis wavelet for P_2U_2. (b) Analysis wavelet for P_4U_2. (c) Analysis wavelet for P_4U_4.



(d) Analysis scale for P_2U_2. (e) Analysis scale for P_4U_2. (f) Analysis scale for P_4U_4.



(g) Synthesis wavelet for P_2U_2. (h) Synthesis wavelet for P_4U_2. (i) Synthesis wavelet for P_4U_4.



(j) Synthesis scale for P_2U_2. (k) Synthesis scale for P_4U_2. (l) Synthesis scale for P_4U_4.

Figure 2.13: Limit wavelet and scale functions obtained after seven iterations on the low-pass channel. Every column represents a different combination of predict (P) and update (U) filters leading to different number of dual and primal vanishing moments.

Chapter 3

Adaptive Lifting Scheme Structure

3.1 Introduction

As stated in Section 1.2, the goal is to create an adaptive wavelet transform based on the well-known wavelet family. In this way we would like to keep the good properties of the known transform, such as the polynomial approximation property, and introduce additional improvements by adapting the transform locally. In order to build adaptive wavelets, we use interpolating quincunx wavelets described in the previous chapter.

The lifting scheme structure enables us to introduce the adaptation in the filter bank structure while still retaining the necessary perfect reconstruction property of the filter bank.

In this chapter we present two filter bank structures that will allow the filter bank to become locally adaptive while still retaining the number of primal and dual vanishing moments. The first structure called PPaU has already been proposed in [Seršić 02, Vrankić 03]. The novel second structure called PUPa introduces significant improvements in the adaptive filter bank structure.

3.2 The PPaU Structure

The adaptive lifting scheme structure that has been proposed in [Vrankić 03] is shown in Figure 3.1. It consists of a fixed prediction step that guarantees $\tilde{N}$ vanishing moments of a dual wavelet. The fixed prediction step is followed by an adaptive prediction step. The analysis filter bank structure ends with the fixed update step that gives N vanishing moments to the primal wavelet. The predict and update filters are given is Section 2.3.3.

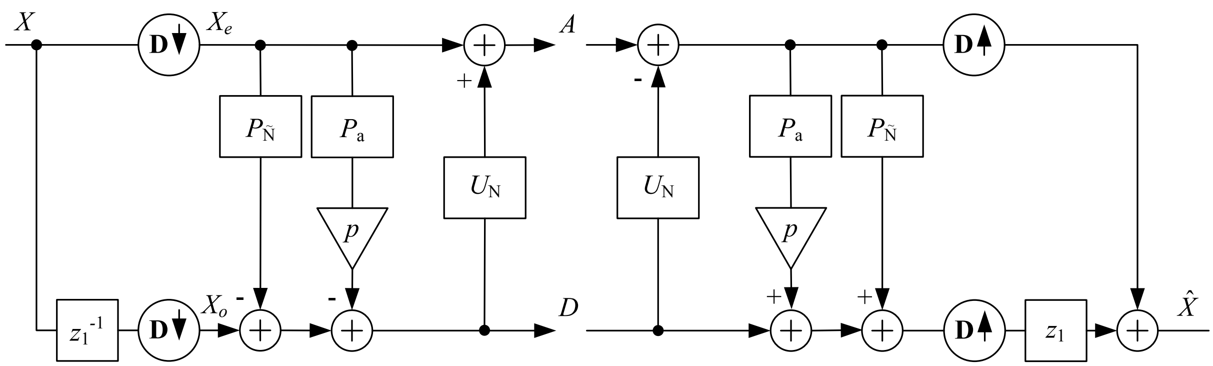


Figure 3.1: PPaU analysis and synthesis adaptive filter bank structure.

3.2.1 Dual Vanishing Moments are Preserved

The analysis filter bank structure is shown in Figure 3.2. The dual vanishing moments condition requires that

$$(\downarrow \mathbf{D})H_1\pi = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}, \tag{3.1}$$

where $\Pi_{\tilde{N}}$ denotes the space of all polynomial sequences of total degree strictly less than $\tilde{N}$.

Downsampled output of the analysis high pass filter can be expressed in terms of the polyphase components, each polyphase component affecting one sublattice which leads to

$$(\downarrow \mathbf{D})H_1\pi(\mathcal{Z}^2) = H_{1e}\pi(\mathbf{D}\mathcal{Z}^2) + H_{1o}\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.2}$$

Using equation 2.30

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

the DM condition becomes:

$$-P\pi(\mathbf{D}\mathcal{Z}^2) + \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}, \tag{3.3}$$

which yields

$$P\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.4}$$

Equation 3.4 can be expressed in the downsampled domain as

$$P\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.5}$$

Now, introducing the overall prediction filter $P = P_{\tilde{N}} + pP_a$ that consists of the fixed branch and the adaptive branch we obtain

$$\begin{aligned} (P_{\tilde{N}} + pP_a)\pi(\mathcal{Z}^2) &= \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}), \\ P_{\tilde{N}}\pi(\mathcal{Z}^2) + pP_a\pi(\mathcal{Z}^2) &= \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}). \end{aligned} \tag{3.6}$$

Since $P_{\tilde{N}}$ is a Neville filter of order $\tilde{N}$ and shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$

$$P_{\tilde{N}}\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.7}$$

which gives

$$\pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) + pP_a\pi(\mathcal{Z}^2) = \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}), \tag{3.8}$$

which further simplifies the DM condition to

$$pP_a\pi(\mathcal{Z}^2) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.9}$$

So, in order to keep $\tilde{N}$ dual vanishing moments introduced with $P_{\tilde{N}}$, filter P_a must cancel polynomials of degree lower than $\tilde{N}$.

Our choice is to create

$$P_a = P_{\tilde{R}} - P_{\tilde{S}}, \tag{3.10}$$

where $P_{\tilde{R}}$ and $P_{\tilde{S}}$ are Neville filters of shift $\boldsymbol{\tau} = \mathbf{D}^{-1}\mathbf{t}$ and order $\tilde{R} \geq \tilde{N}$ and $\tilde{S} \geq \tilde{N}$ respectively (see Figure 3.2). Applying this filter section to a polynomial sequence $\pi \in \Pi_{\tilde{N}}$ gives

$$p(P_{\tilde{R}} - P_{\tilde{S}})\pi(\mathcal{Z}^2) = p(\pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t}) - \pi(\mathcal{Z}^2 + \mathbf{D}^{-1}\mathbf{t})) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.11}$$

Therefore, with the choice of the P_a filter from equation 3.10, the $\tilde{N}$ dual vanishing moments are guaranteed no matter what values the p parameter may take.

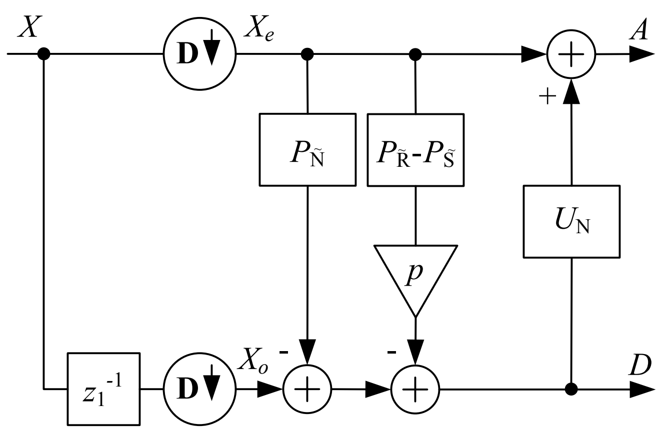


Figure 3.2: Analysis filter bank.

In [Vrankić 03] a more general structure has been proposed with more than one adaptive branch. However, more than one adaptive branch was not practical in constructing the adaptation algorithm.

We define the filter adaptive prediction filter $P_{\tilde{N}+}$ composed of Neville filters as

$$P_{\tilde{N}+} = P_{\tilde{N}} + p(P_{\tilde{R}} - P_{\tilde{S}}) \quad \text{for } \tilde{R} \geq \tilde{N} \text{ and } \tilde{S} \geq \tilde{N}. \tag{3.12}$$

3.2.2 Primal Vanishing Moments are Preserved

Now, we will show that a number of primal vanishing moments N that is fixed with the update filter is not influenced by the adaptive prediction branch as long as

$$\tilde{N} \geq N. \tag{3.13}$$

The synthesis filter bank is shown in Figure 3.3.

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

$$G_1(\uparrow \mathbf{D})\pi = 0 \quad \text{for } \pi \in \Pi_N, \tag{3.14}$$

where Π_N denotes the space of all polynomial sequences of total degree strictly less than N.

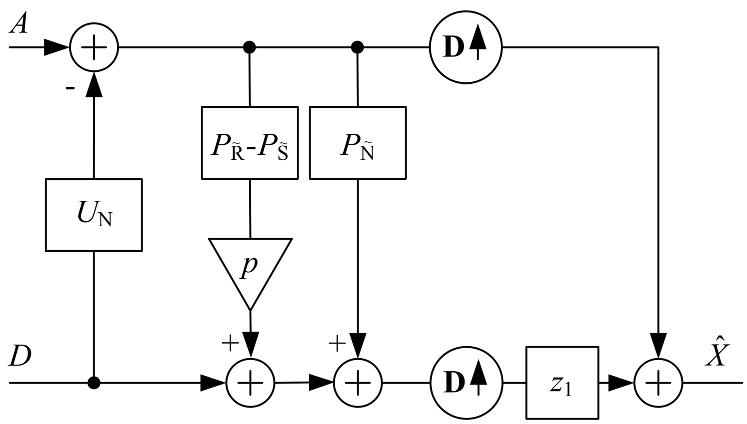


Figure 3.3: Synthesis filter bank.

Using equation 2.32

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

the PM condition can be stated in terms of polyphase components:

$$\begin{aligned} G_1(\uparrow \mathbf{D})\pi(\mathcal{Z}^2) &= G_{1e}\pi(\mathbf{D}\mathcal{Z}^2) + G_{1o}\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \\ &= -U_N^*\pi(\mathbf{D}\mathcal{Z}^2) + (1 - U_N^* P_{\tilde{N}+}^*)\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \\ &= -U_N^*\pi(\mathbf{D}\mathcal{Z}^2) + \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) - U_N^* P_{\tilde{N}+}^*\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_N. \end{aligned} \tag{3.15}$$

If filter $P_{\tilde{N}+}$ is such that

$$\begin{aligned} \tilde{N} &\geq N, \\ \tilde{R} &\geq N, \\ \tilde{S} &\geq N, \end{aligned} \tag{3.16}$$

then

$$
\begin{aligned}
P^*_{\tilde{N}+}\pi(\mathbf{D}\mathcal{Z}^2+\mathbf{t}) &= (P^*_{\tilde{N}} + p(P^*_{\tilde{R}} - P^*_{\tilde{S}}))\pi(\mathbf{D}\mathcal{Z}^2+\mathbf{t}) \\
&= P^*_{\tilde{N}+}\pi(\mathbf{D}\mathcal{Z}^2+\mathbf{t}) + \underbrace{p(P^*_{\tilde{R}} - P^*_{\tilde{S}})\pi(\mathbf{D}\mathcal{Z}^2+\mathbf{t})}_{=0} \\
&= \pi(\mathbf{D}\mathcal{Z}^2),
\end{aligned}
\tag{3.17}
$$

since $P^*_{\tilde{N}}$ is a Neville filter of order $\tilde{N}$ and shift $-\boldsymbol{\tau}$, which equals $-\mathbf{D}^{-1}\mathbf{t}$. Therefore, the PM condition becomes

$$
2U^*_N\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2+\mathbf{t}) \quad \text{for } \pi \in \Pi_N, \tag{3.18}
$$

which is always true since $2U^*_N = P_N$.

3.3 The PUPa Structure

The proposed novel adaptive structure is shown in Figure 3.4. The analysis filter bank consists of three stages. After the polyphase decomposition the fixed quincunx Neville predict filter of order $\tilde{N}$ is applied. This stages results in $\tilde{N}$ vanishing moments of a dual wavelet. Next, the update filter U_N is applied which results in N vanishing moments of a primal wavelet. As stated before, $N \leq \tilde{N}$.

The adaptation is introduced in the final lifting stage with the pP_a filter. The big advantage of such a structure compared to the PPaU structure is that the adaptation in one decomposition level of the iterated filter bank structure influenced only that very decomposition level and it does not influence next adaptation levels. In other words, iterated analysis low-pass filter $H_0^{(k)}$ remains the same, only the high-pass filters $H_1^{(k)}$ change.

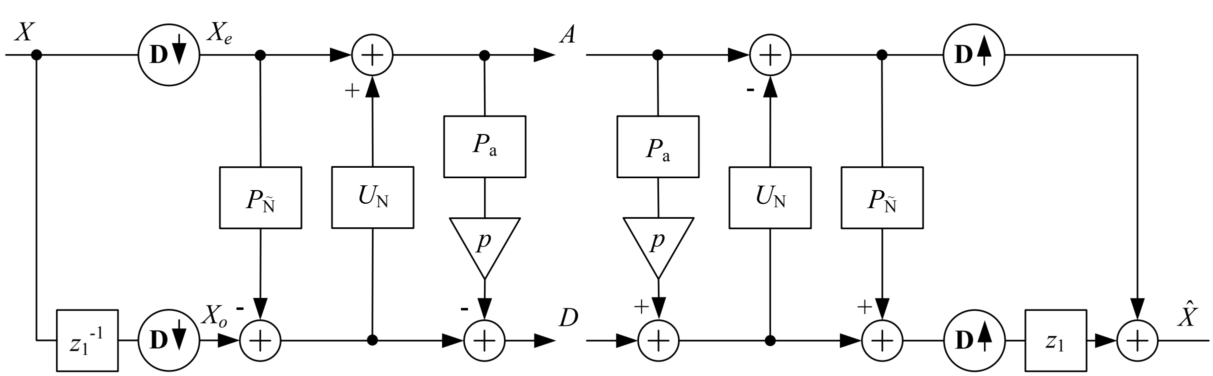


Figure 3.4: PUPa analysis and synthesis adaptive filter bank structure.

3.3.1 Dual Vanishing Moments are Preserved

The polyphase matrix of the analysis filter bank from Figure 3.5 is

$$\begin{aligned}\mathbf{H}_p &= \begin{bmatrix} 1 & 0 \\ -pP_a & 1 \end{bmatrix} \begin{bmatrix} 1 & U_N \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -P_{\tilde{N}} & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 - P_{\tilde{N}} U_N & U_N \\ -pP_a(1 - P_{\tilde{N}} U_N) - P_{\tilde{N}} & -pP_a U_N + 1 \end{bmatrix}\end{aligned} \tag{3.19}$$

Therefore, the polyphase components of the analysis filters are

$$\begin{aligned} H_{1e} &= -pP_a(1 - P_{\tilde{N}} U_N) - P_{\tilde{N}} \\ H_{1o} &= -pP_a U_N + 1 \end{aligned} \tag{3.20}$$

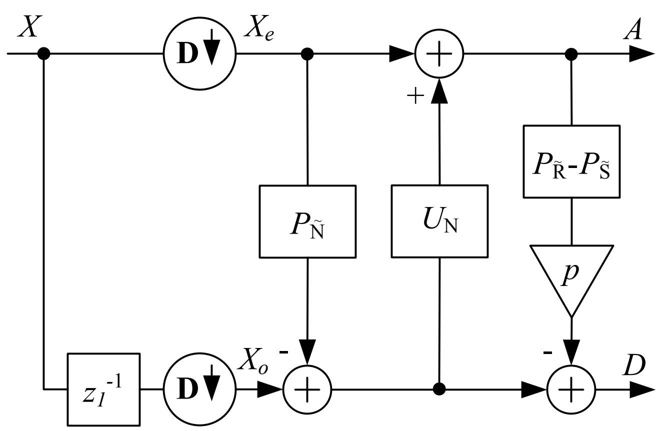


Figure 3.5: Analysis filter bank.

Using the DM condition

$$H_{1e}\pi(\mathbf{D}\mathcal{Z}^2) + H_{1o}\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.21}$$

gives

$$(-pP_a(1 - P_{\tilde{N}} U_N) - P_{\tilde{N}})\pi(\mathbf{D}\mathcal{Z}^2) + (-pP_a U_N + 1)\pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.22}$$

Taking the P_a filter from equation 3.10

$$P_a = P_{\tilde{R}} - P_{\tilde{S}} \text{ where } \tilde{R} \geq \tilde{N} \text{ and } \tilde{S} \geq \tilde{N},$$

for which

$$P_a \pi(\mathcal{Z}^2) = 0 \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.23}$$

the expression 3.22 becomes

$$P_{\tilde{N}}\pi(\mathbf{D}\mathcal{Z}^2) = \pi(\mathbf{D}\mathcal{Z}^2 + \mathbf{t}) \quad \text{for } \pi \in \Pi_{\tilde{N}}. \tag{3.24}$$

which is always true.

Therefore, with the choice of the P_a filter from equation 3.10 the $\tilde{N}$ dual vanishing moments are guaranteed no matter what values the parameter p may take!

3.3.2 Primal Vanishing Moments are Preserved

The synthesis polyphase matrix of the filter bank from Figure 3.6 can be obtained from the analysis polyphase matrix 3.19 as

$$\mathbf{G}_p = \begin{bmatrix} -pP_aU_N^* + 1 & pP_a(1 - P_{\tilde{N}}^*U_N^*) + P_{\tilde{N}}^* \\ -U_N^* & 1 - P_{\tilde{N}}^*U_N^* \end{bmatrix} \tag{3.25}$$

Therefore,

$$\begin{aligned} G_{1e} &= -U_N^* \\ G_{1o} &= 1 - P_{\tilde{N}}^*U_N^* \end{aligned} \tag{3.26}$$

which is the same as for the PPaU structure so we conclude that N primal vanishing moments are preserved as long as

$$\tilde{N} \geq N. \tag{3.27}$$

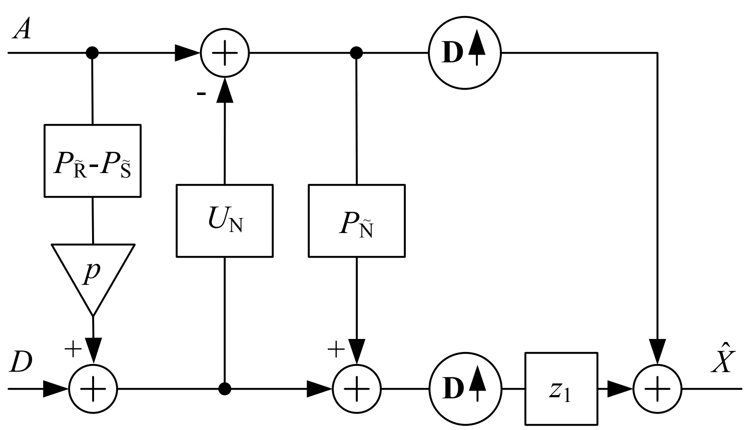


Figure 3.6: Synthesis filter bank.

This result is obvious since the adaptive predict step does not influence the high-pass filter on the synthesis side, as shown in Figure 3.6.

3.4 Changing the p Parameter

This section investigates the influence of the value of the parameter p on the resulting filters and wavelet and scale functions. The smallest support P_2 and U_2 filters shown in figures 3.7 and 3.8 will be used.

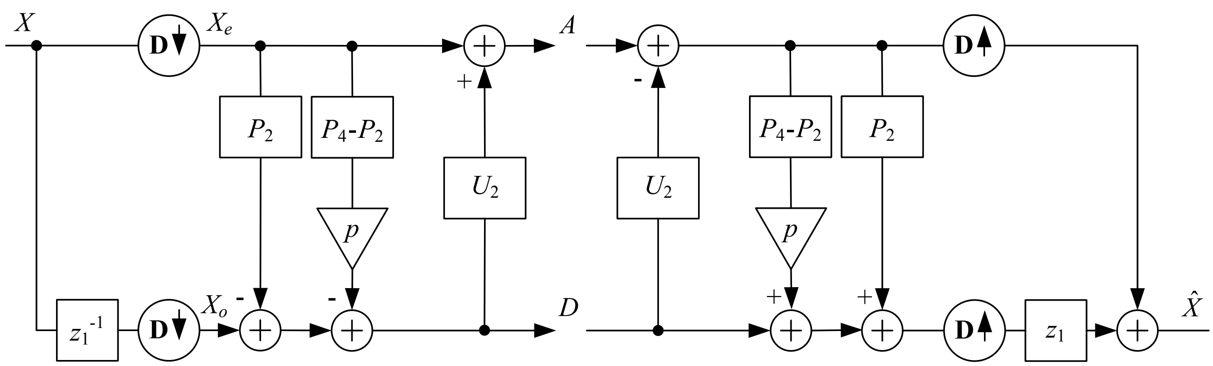


Figure 3.7: PPaU analyisis and synthesis adaptive filter bank structure.

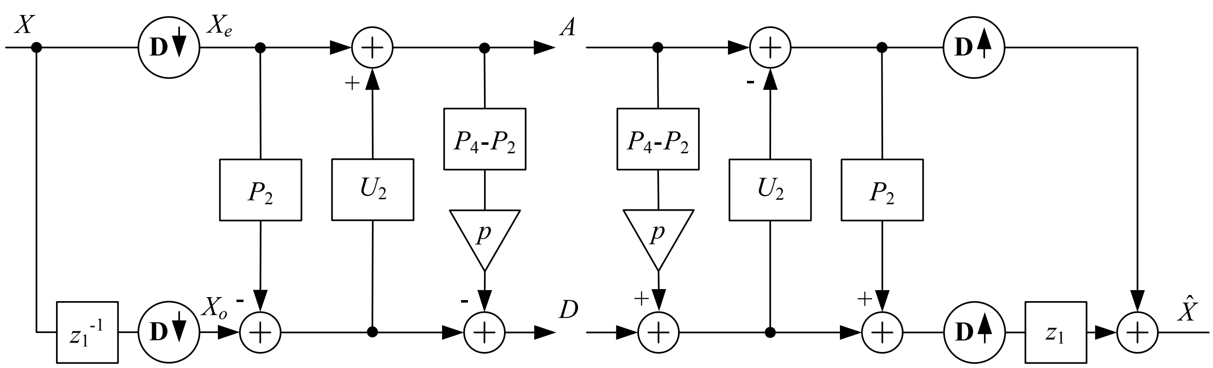
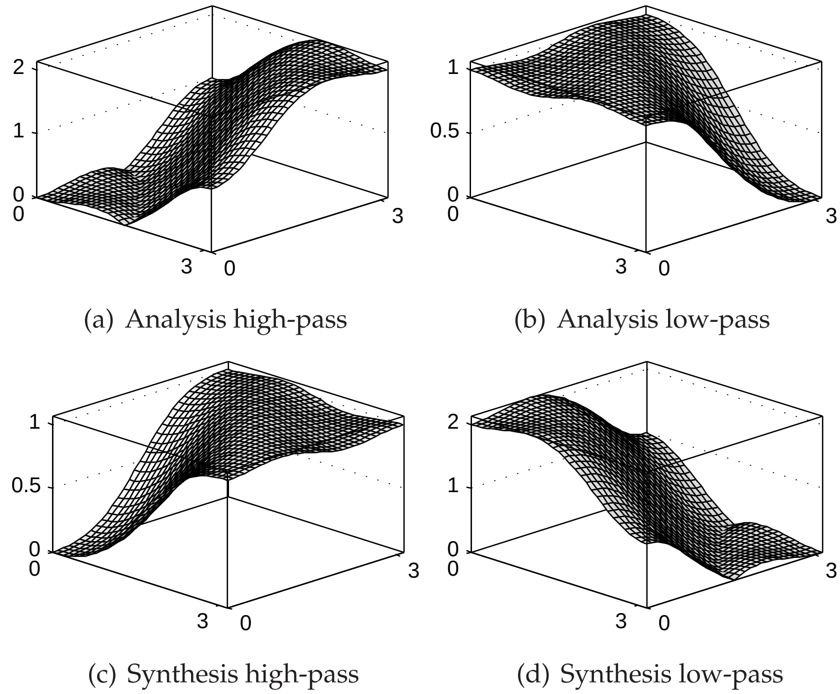


Figure 3.8: PUPa analyisis and synthesis adaptive filter bank structure.

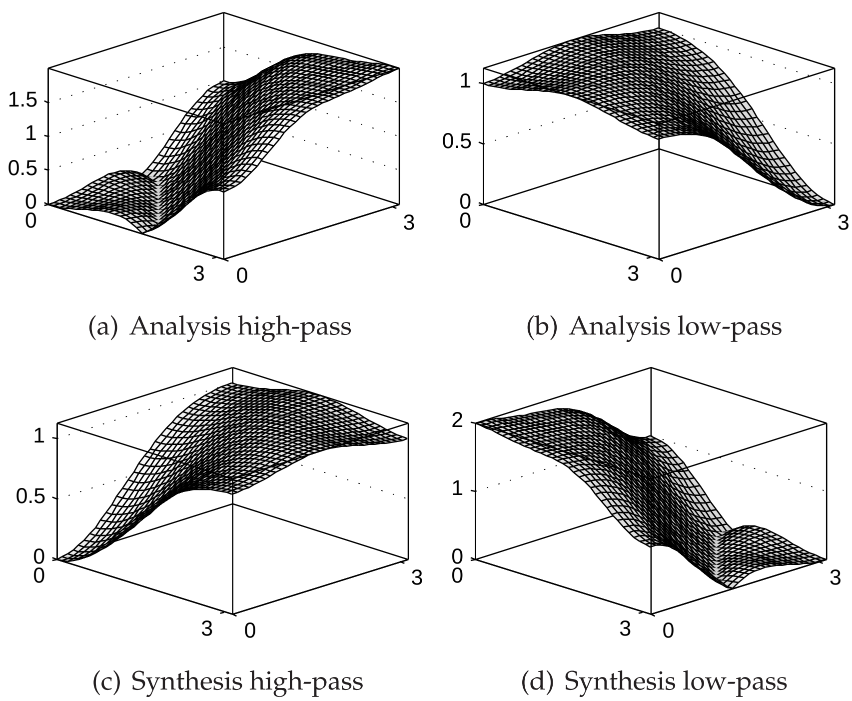
3.4.1 Resulting Filters

The magnitude frequency responses of the resulting filters for $p = 2$ for PPaU and PUPa structures are shown in Figures 3.9 and 3.10 respectively. The magnitude frequency responses of the analysis high-pass filters for different values of the parameter p are shown in Figures 3.11 and 3.12. It is important to note that with the increase of the parameter p the zero ditch appears in the resulting magnitude response of the analysis high-pass filter. As shown in Figure 3.12, the zero ditch grows with the parameter p. Therefore, by changing the value of the parameter p, it is possible to cancel some frequency components present in the input image.



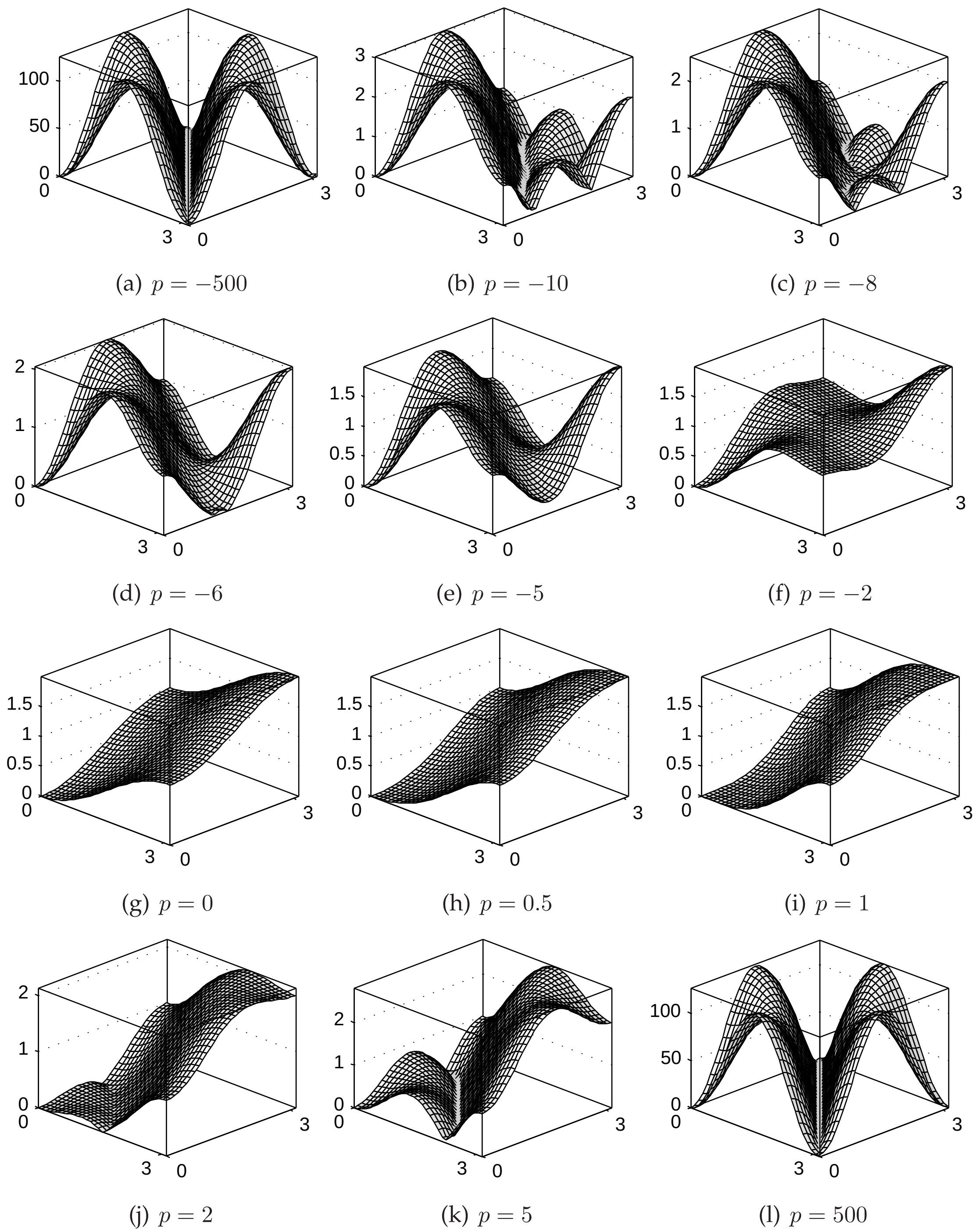
(a) Analysis high-pass
(b) Analysis low-pass
(c) Synthesis high-pass
(d) Synthesis low-pass

Figure 3.9: The PPaU structure with $p = 2$. Magnitude frequency response for analysis and synthesis filters with spatial frequencies ω_1 and ω_2 in the range from 0 to π.



(a) Analysis high-pass
(b) Analysis low-pass
(c) Synthesis high-pass
(d) Synthesis low-pass

Figure 3.10: The PUPa structure with $p = 2$. Magnitude frequency response for analysis and synthesis filters.



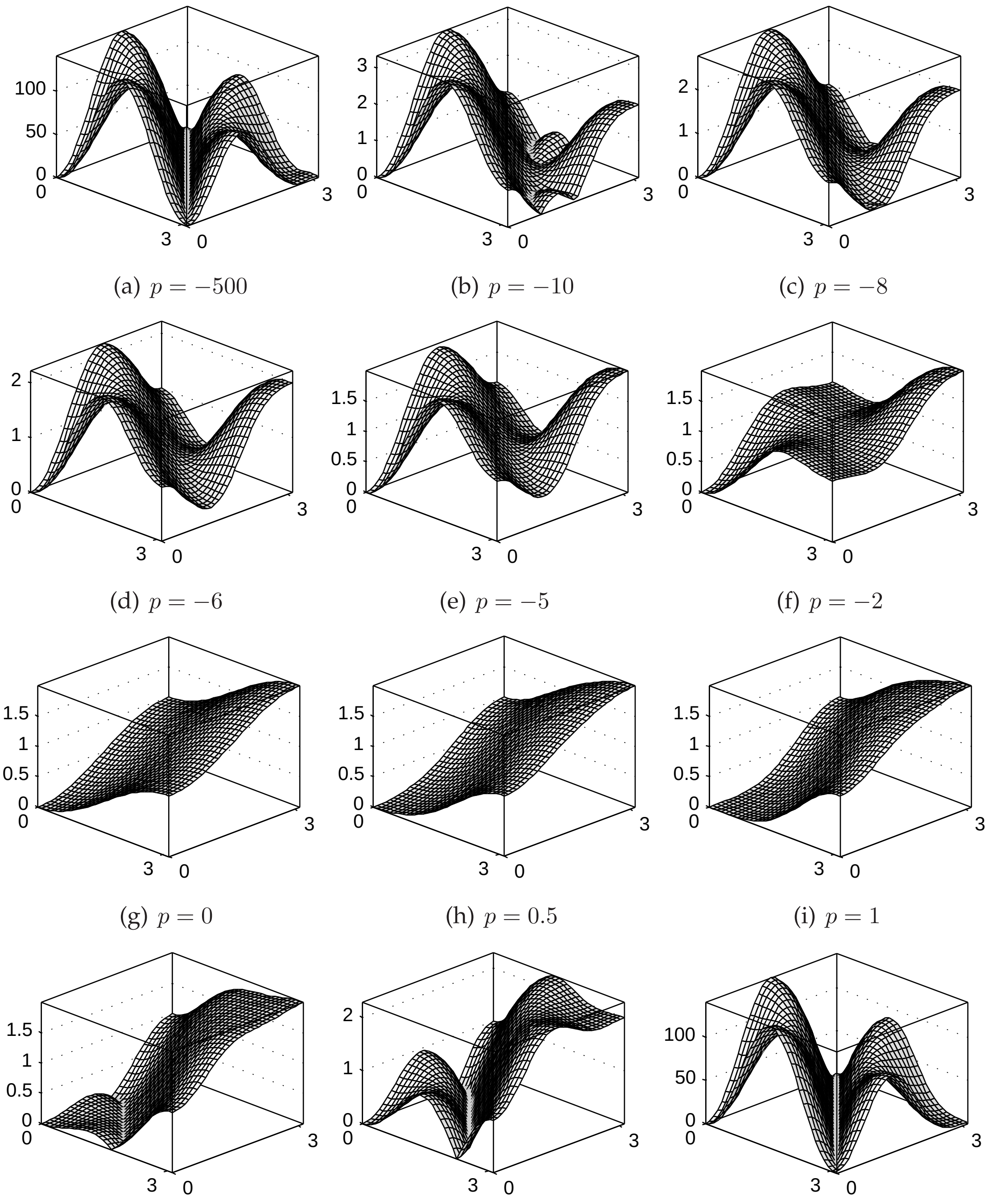
(a) $p = -500$ (b) $p = -10$ (c) $p = -8$

(d) $p = -6$ (e) $p = -5$ (f) $p = -2$

(g) $p = 0$ (h) $p = 0.5$ (i) $p = 1$

(j) $p = 2$ (k) $p = 5$ (l) $p = 500$

Figure 3.11: The PPaU structure. Analysis high-pass filter resulting for different values of p.



(a) $p = -500$ (b) $p = -10$ (c) $p = -8$

(d) $p = -6$ (e) $p = -5$ (f) $p = -2$

(g) $p = 0$ (h) $p = 0.5$ (i) $p = 1$

(j) $p = 2$ (k) $p = 5$ (l) $p = 500$

Figure 3.12: The PUPa structure. Analysis high-pass filter resulting for different values of p.

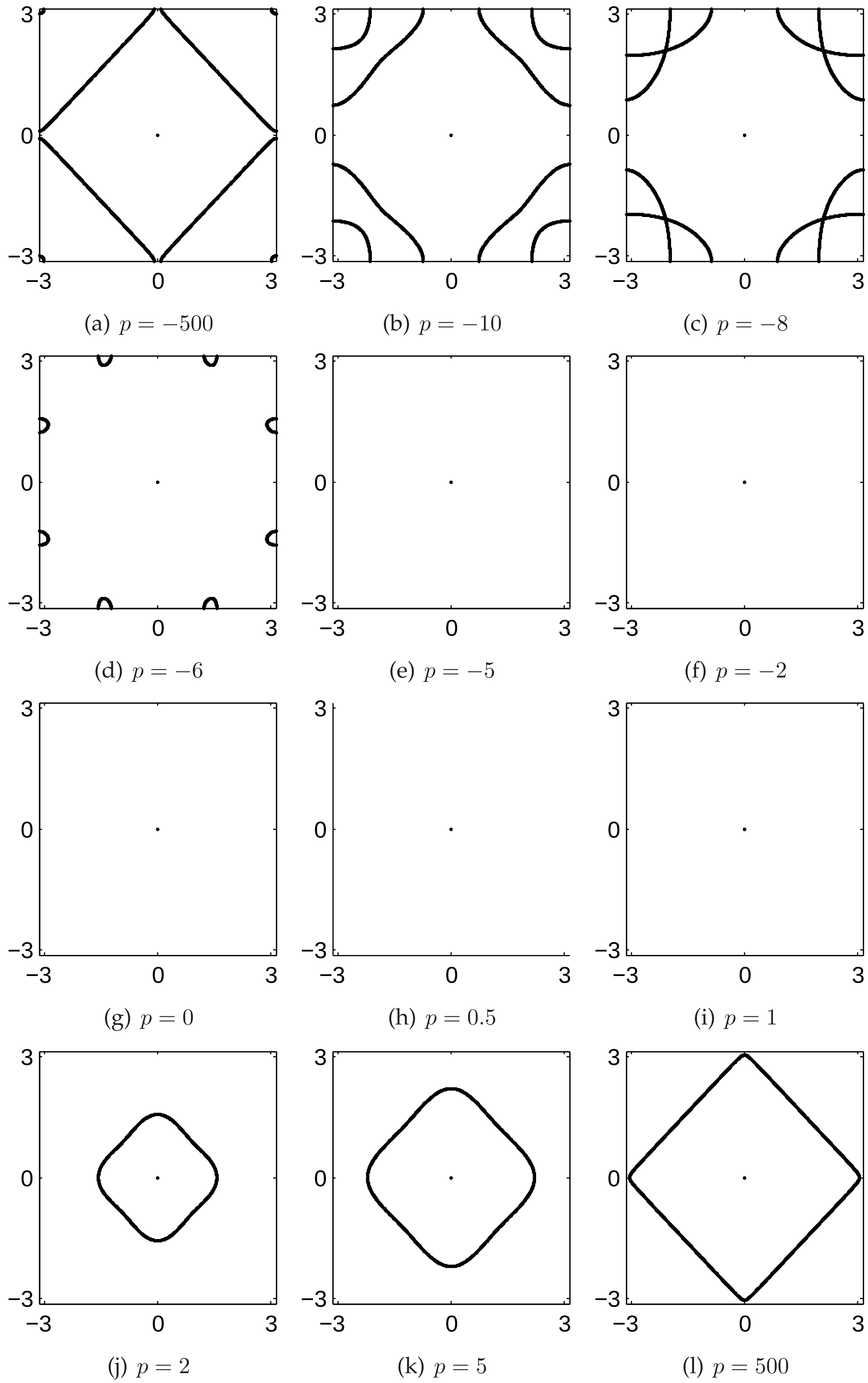
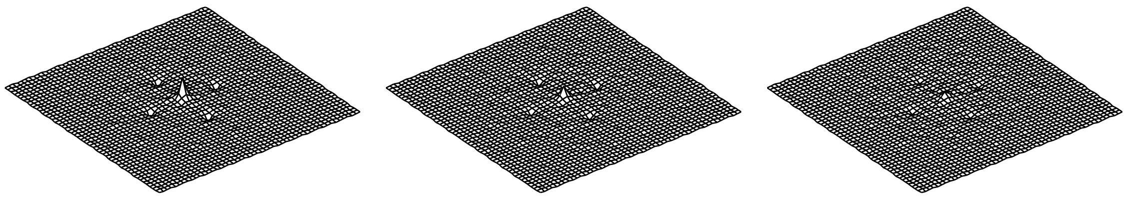


Figure 3.13: The PPaU structure. Zero locations of the magnitude frequency responses of the analysis high-pass filter for different values of p.

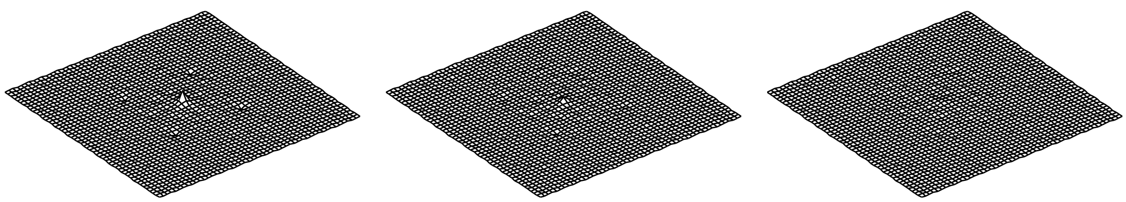
3.4.2 Limit Wavelet and Scale Functions

The limit wavelet and scale functions are obtained by setting a single value of the parameter p in all decomposition levels and calculating the impulse response of the 7-level iterated filter bank. The analysis and synthesis limit wavelet and scale functions for different values of the parameter p for the PPaU and PUPa structures are shown in Figures 3.14 and 3.15 respectively. In Figures 3.16 and 3.17, the analysis limit wavelet functions are shown for a wider range of the values of p. The synthesis limit wavelet functions are shown in Figures 3.18 and 3.19. The limit functions are regular for smaller values of the parameter p. However, for absolute values of p greater than approximately 5, the limit functions become increasingly irregular. Therefore, for a limited range of p values, the "well-behaved" analysis and synthesis functions are obtained.

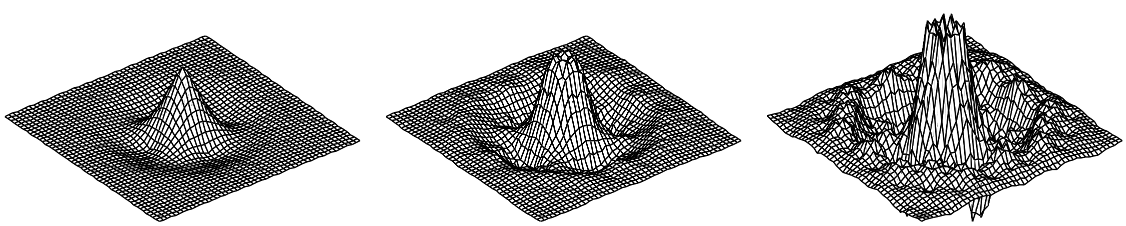
Comparing the wavelet figures for the PUPa and the PPaU structure, it can be seen that synthesis wavelet function behave similarly when the parameter p grows, they become increasingly irregular. On the other hand, the analysis wavelet functions behave differently. For the PPaU structure, the analysis wavelet functions change significantly with the increase of parameter p. Yet, for the PUPa structure, the analysis wavelet functions look similar for the whole range of the p values. The reason lies in the fact that the analysis low-pass filter of the iterated filter bank does not change so the iterated analysis high-pass filter changes only because of the value of the p parameter it the last level of the filter bank cascade.



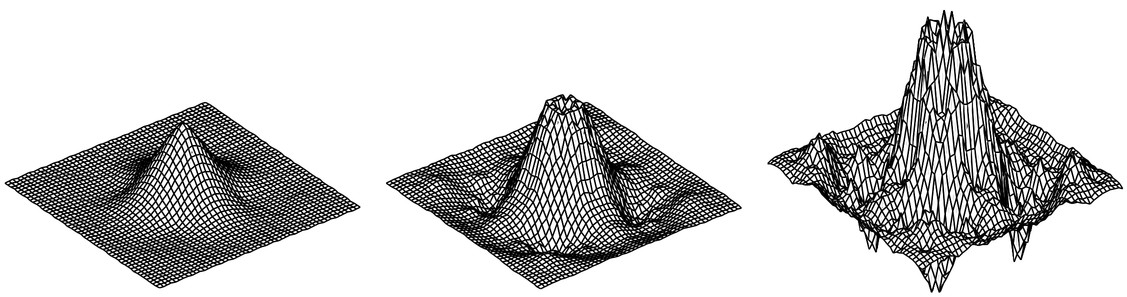
(a) Analysis wavelet for $p = 1$ (b) Analysis wavelet for $p = 3$ (c) Analysis wavelet for $p = 5$



(d) Analysis scale for $p = 1$ (e) Analysis scale for $p = 3$ (f) Analysis scale for $p = 5$

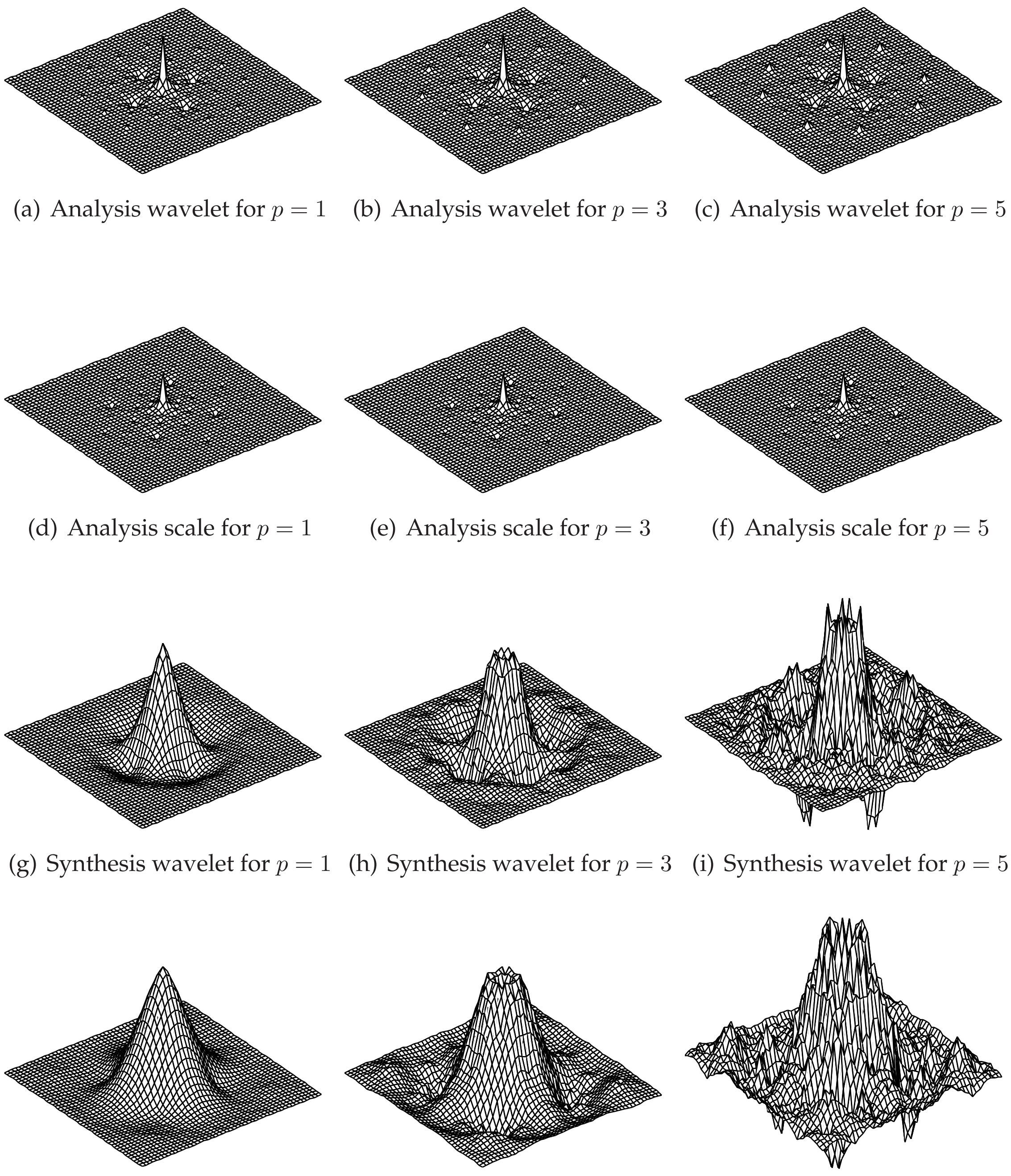


(g) Synthesis wavelet for $p = 1$ (h) Synthesis wavelet for $p = 3$ (i) Synthesis wavelet for $p = 5$



(j) Synthesis scale for $p = 1$ (k) Synthesis scale for $p = 3$ (l) Synthesis scale for $p = 5$

Figure 3.14: The PPaU structure. Analysis and synthesis limit wavelet and scale functions obtained as impulse responses of the high-pass and low-pass filters after 7 iterations of the FB structure.



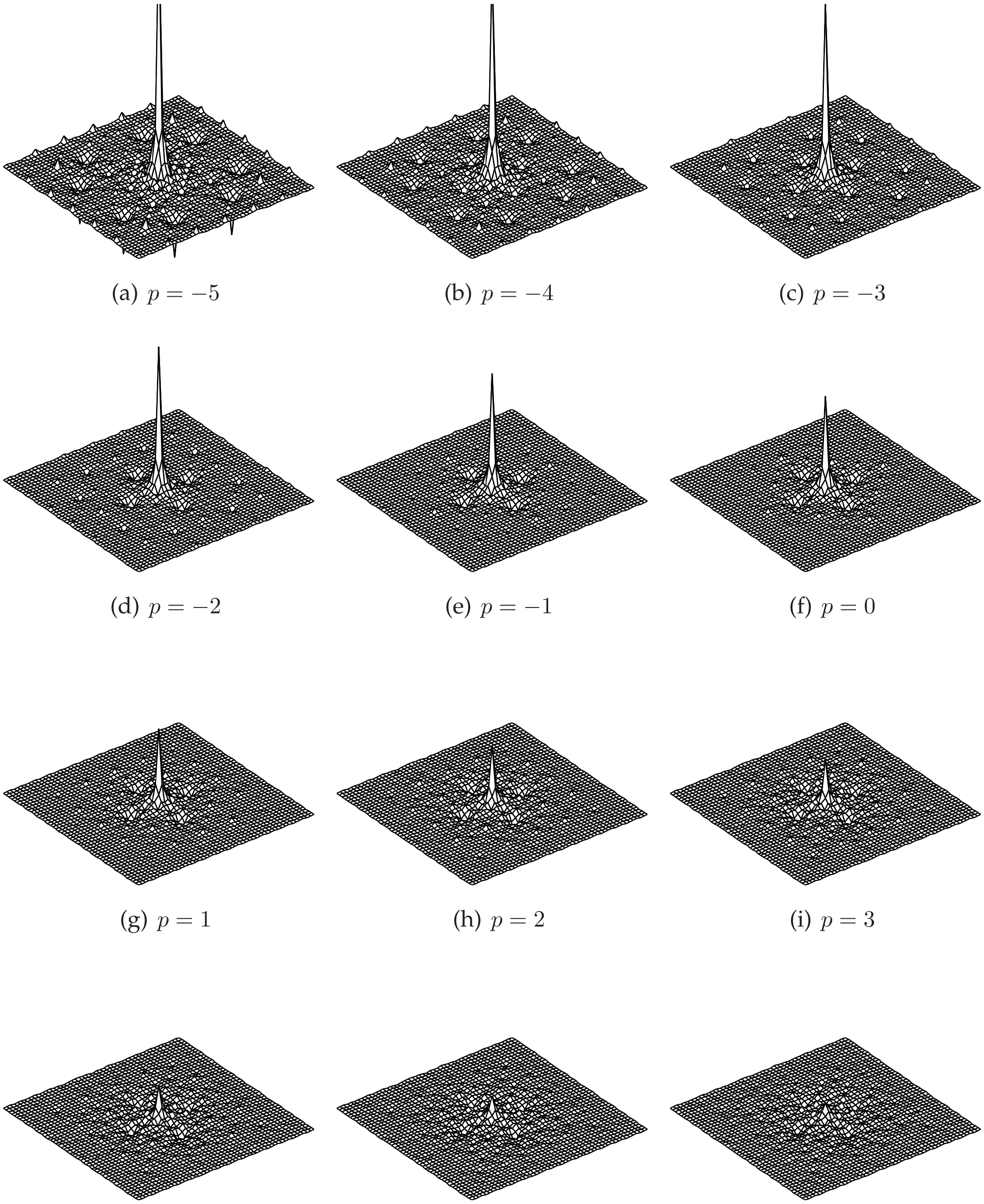
(a) Analysis wavelet for $p = 1$ (b) Analysis wavelet for $p = 3$ (c) Analysis wavelet for $p = 5$

(d) Analysis scale for $p = 1$ (e) Analysis scale for $p = 3$ (f) Analysis scale for $p = 5$

(g) Synthesis wavelet for $p = 1$ (h) Synthesis wavelet for $p = 3$ (i) Synthesis wavelet for $p = 5$

(j) Synthesis scale for $p = 1$ (k) Synthesis scale for $p = 3$ (l) Synthesis scale for $p = 5$

Figure 3.15: The PUPa structure. Analysis and synthesis limit wavelet and scale functions obtained as impulse responses of the high-pass and low-pass filters after 7 iterations of the FB structure.



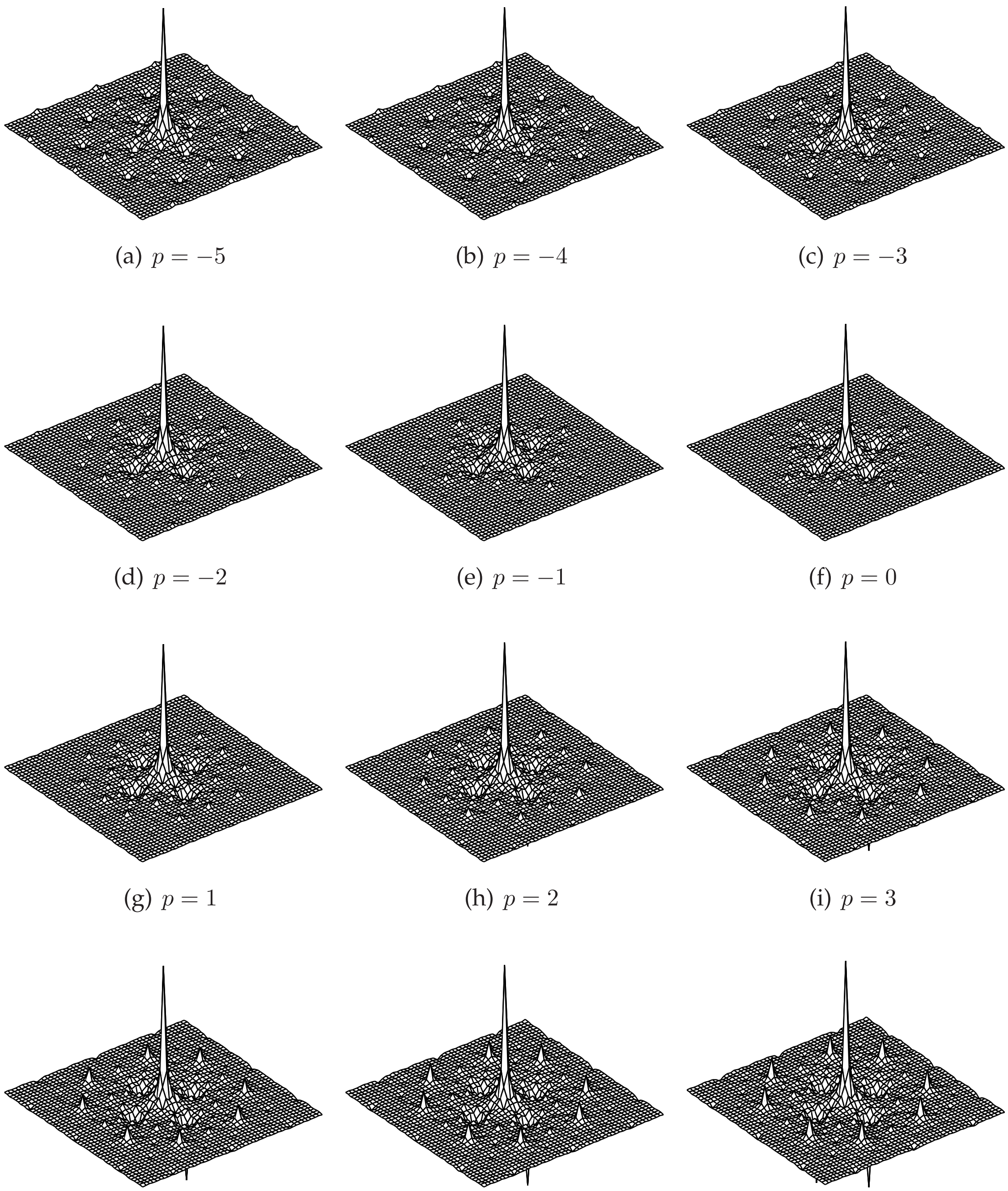
(a) $p = -5$ (b) $p = -4$ (c) $p = -3$

(d) $p = -2$ (e) $p = -1$ (f) $p = 0$

(g) $p = 1$ (h) $p = 2$ (i) $p = 3$

(j) $p = 4$ (k) $p = 5$ (l) $p = 6$

Figure 3.16: The PPaU structure. Analysis limit wavelet functions obtained as impulse responses of the high-pass filter after 7 iterations of the FB structure.



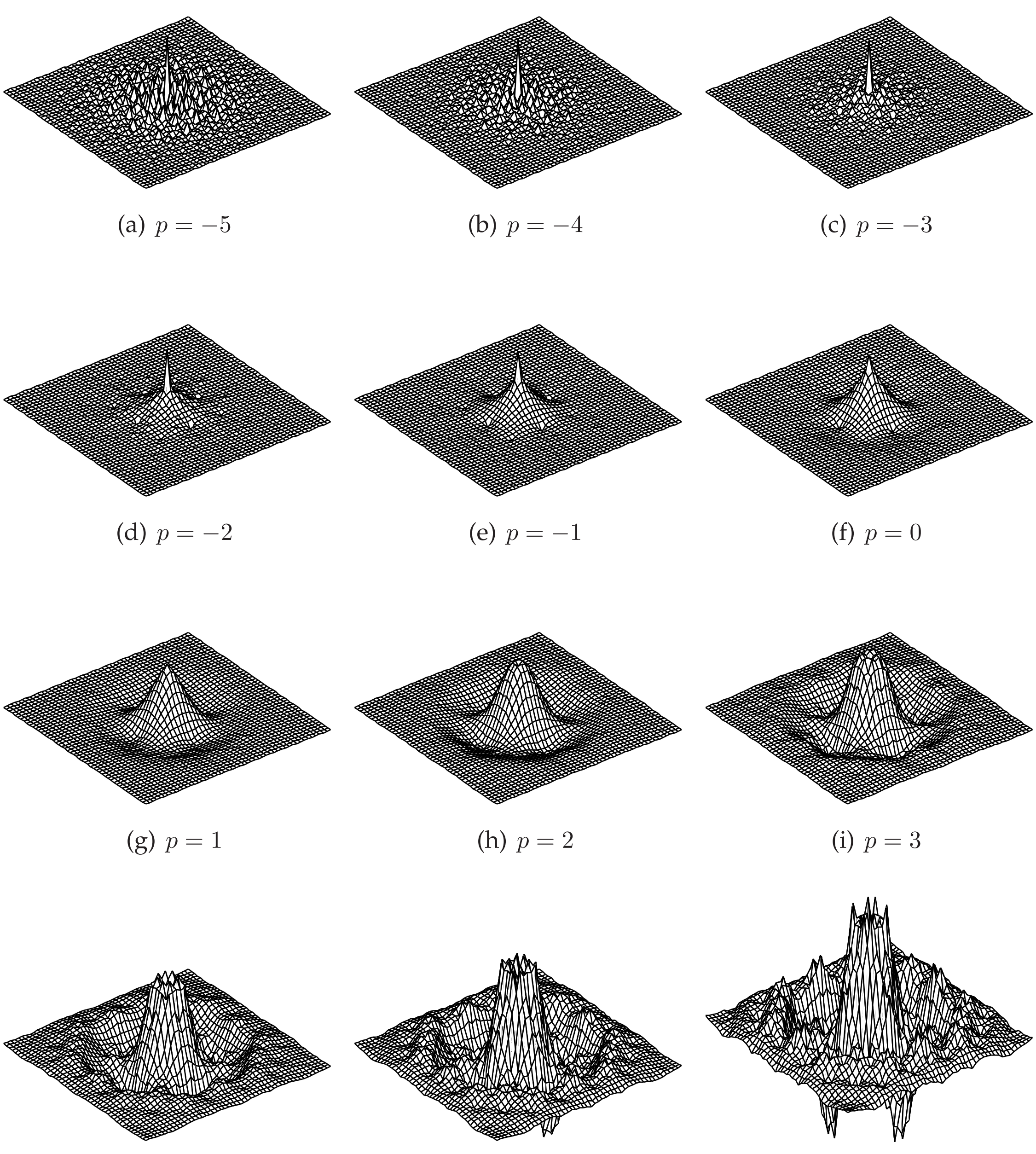
(a) $p = -5$ (b) $p = -4$ (c) $p = -3$

(d) $p = -2$ (e) $p = -1$ (f) $p = 0$

(g) $p = 1$ (h) $p = 2$ (i) $p = 3$

(j) $p = 4$ (k) $p = 5$ (l) $p = 6$

Figure 3.17: The PUPa structure. Analysis limit wavelet functions obtained as impulse responses of the high-pass filter after 7 iterations of the FB structure.



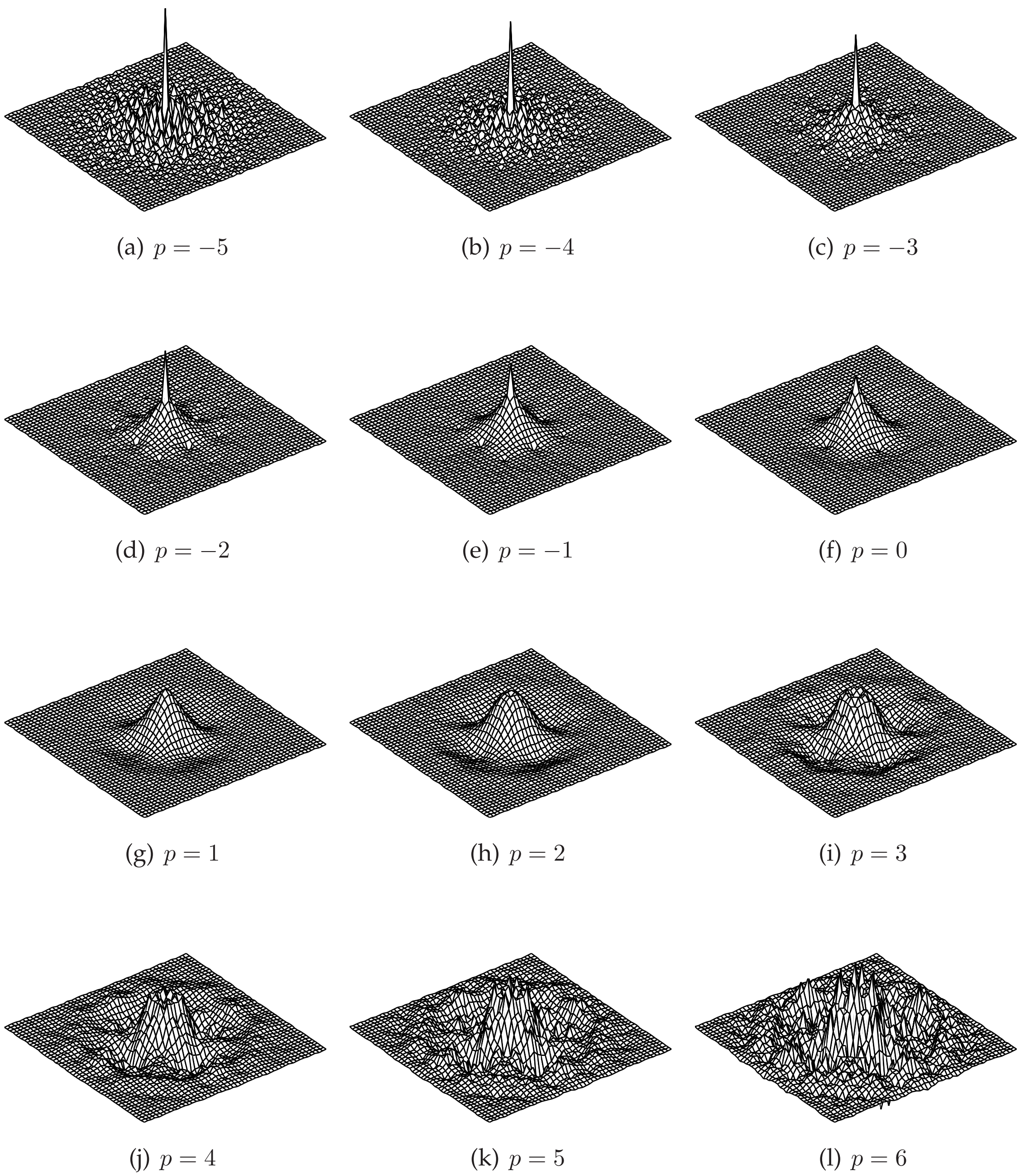
(a) $p = -5$ (b) $p = -4$ (c) $p = -3$

(d) $p = -2$ (e) $p = -1$ (f) $p = 0$

(g) $p = 1$ (h) $p = 2$ (i) $p = 3$

(j) $p = 4$ (k) $p = 5$ (l) $p = 6$

Figure 3.18: The PPaU structure. Synthesis limit wavelet functions obtained as impulse responses of the high-pass filter after 7 iterations of the FB structure.



(a) $p = -5$ (b) $p = -4$ (c) $p = -3$

(d) $p = -2$ (e) $p = -1$ (f) $p = 0$

(g) $p = 1$ (h) $p = 2$ (i) $p = 3$

(j) $p = 4$ (k) $p = 5$ (l) $p = 6$

Figure 3.19: The PUPa structure. Synthesis limit wavelet functions obtained as impulse responses of the high-pass filter after 7 iterations of the FB structure.

Chapter 4

Adapting the Filter Bank Parameters

4.1 Introduction

In the previous chapter filter banks that can tune their properties while still keeping a number of vanishing moments were presented. In this chapter, we introduce the adaptation methods that will be used to tune the parameter p of the filter bank. The goal of the adaptation is to minimize the energy of the detail coefficients.

In this chapter we use the undecimated filter bank structure for two reasons. Firstly, we will use our decomposition for the image denoising. It was shown that for denoising purposes the undecimated wavelet filter bank outperforms a decimated filer bank [Coifman 95]. Secondly, the undecimated filter bank structure leads to simpler expressions and the results obtained for the undecimated filter bank will be straightforwardly applicable to the decimated filter bank structure.

4.2 Problem Framework

The undecimated analysis filter bank based on the PPaU structure that was presented in Section 3.2 is shown in Figure 4.2 and its simplified equivalent is shown in Figure 4.3. The adaptation of the parameter p will first be derived for the PPaU structure and then the results will be generalized for the PUPa structure from section 3.3.

4.2.1 Notation

For simplicity of notation we will use linear indexing of the image pixels so that the two-dimensional indexes (j, k) will be replaced with a single index i. Therefore, instead of indexing a pixel of the image x as $x(j, k)$ we will use x_i with such i that

$$i = (k - 1) \cdot J + j, \tag{4.1}$$

where J is the number of rows of the image.

In the following sections a pixel of a filtered image (obtained by convolution) will be represented as a weighted sum of a number of pixels from a predefined configuration shown in Figure 4.1.

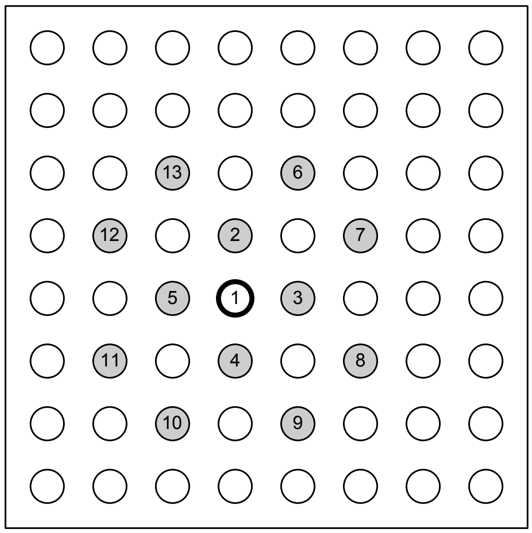


Figure 4.1: The numbering scheme based on a central pixel surrounded with rings of pixels on the quincunx lattice. Only the first two rings are marked.

We introduce relative indexing x_{in} that is based on enumeration of pixels starting from a single pixel x_i. The pixel x_{in} is the one that is located at the position marked with n of the constellation from Figure 4.1 centered at the the pixel x_i.

4.2.2 Input Image Corrupted With Additive Noise

We will consider the case when the clean input image x_0 is corrupted with additive zero mean Gaussian noise w, so the processed image is

$$x = x_0 + w. \tag{4.2}$$

The original image x_0 will be treated as deterministic and the samples of the noise will be treated as independent and identically distributed (IID) random variables.

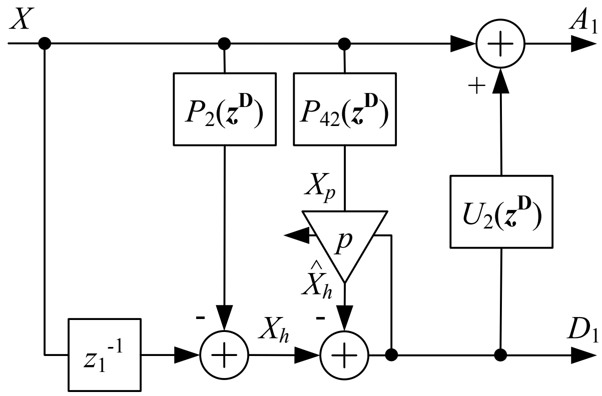


Figure 4.2: The first decomposition level of the adaptive filter bank where $P_{42} = P_4 - P_2$.

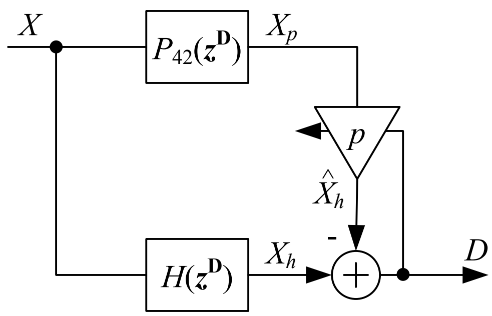


Figure 4.3: Simplified adaptive prediction stage.

The x_p represents the image at the output of the $P_{42} = P_4 - P_2$ filter. The corresponding impulse responses are

$$p_2 = \frac{1}{4}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{4.3}$$

$$p_4 = \frac{1}{32}\begin{pmatrix} 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & 10 & 0 & -1 \\ 0 & 10 & 0 & 10 & 0 \\ -1 & 0 & 10 & 0 & -1 \\ 0 & -1 & 0 & -1 & 0 \end{pmatrix} \tag{4.4}$$

so the impulse response of the filter $P_{42} = P_4 - P_2$ is

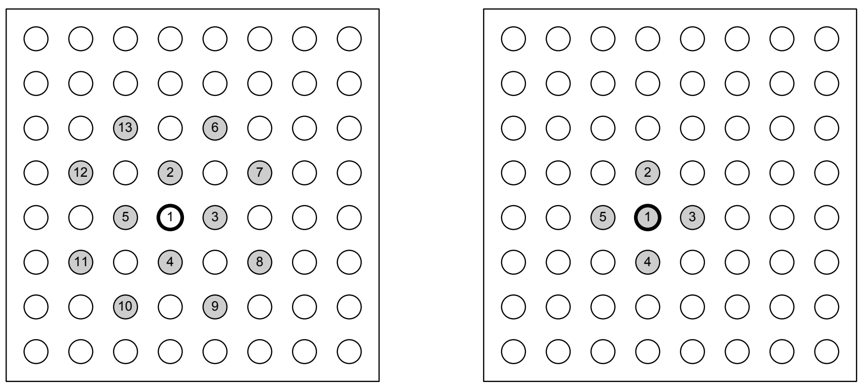
$$p_{42} = \frac{1}{32}\begin{pmatrix} 0 & -1 & 0 & -1 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & 2 & 0 & 2 & 0 \\ -1 & 0 & 2 & 0 & -1 \\ 0 & -1 & 0 & -1 & 0 \end{pmatrix}. \tag{4.5}$$

The image resulting after the fixed prediction branch is denoted as x_h. It is obtained by convolution with the high-pass filter

$$h = \begin{pmatrix} 0 & -\frac{1}{4} & 0 \\ -\frac{1}{4} & 1 & -\frac{1}{4} \\ 0 & -\frac{1}{4} & 0 \end{pmatrix} \tag{4.6}$$

Therefore, a pixel x_{hi}, as shown in Figure 4.4(b), will be obtained as a weighted sum of 5 pixels from the input image x as

$$x_{hi} = x_{i1} - \frac{1}{4}(x_{i2} + x_{i3} + x_{i4} + x_{i5}) \tag{4.7}$$



(a) Support in x for obtaining x_{pi}.

(b) Support in x for obtaining x_{hi}.

Figure 4.4: Filter supports. Circles marked with gray represent pixels in x that are used for obtaining x_{pi} at position 1 (marked with bold circle).

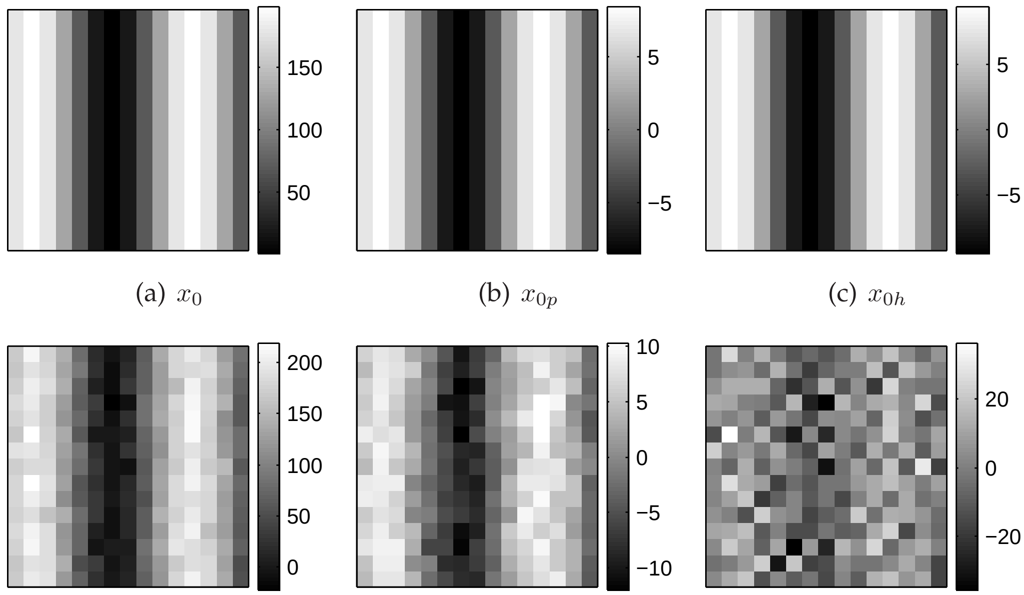
The x_p and h_p can be expressed in terms of the original input image x_0 and the zero mean Gaussian noise w as

$$\begin{aligned} x_p &= (x_0 + w) * p_{42} = x_{0p} + w_p \\ x_h &= (x_0 + w) * h = x_{0h} + w_h \end{aligned} \tag{4.8}$$

where p_{42} is the impulse response of the quincunx-upsampled filter P_{42}. The signals x_{0p} and x_{0h}, which come from filtering the original image will represent a deterministic components of x_p and x_h respectively.

Since $x_p = x * p_{42}$, variance of x_p for any pixel can be obtained as

$$Var(x_{pi}) = Var(w_{pi}) = \sigma_w^2 \sum_n p_{42n}^2 = \sigma_w^2 \left(4\left(\frac{2}{32}\right)^2 + 8\left(\frac{-1}{32}\right)^2\right) = \frac{6}{256}\sigma_w^2 = 0.0234\sigma_w^2 \tag{4.9}$$



(a) x_0 (b) x_{0p} (c) x_{0h}

(d) x for $\sigma_w = 10$ (e) x_p for $\sigma_w = 10$ (f) x_h for $\sigma_w = 10$

Figure 4.5: Original image x_0 is a spatial sine wave with $T = 10$. The input image x (second row) is obtained by adding white Gaussian noise with $\sigma_w = 10$ to the original image, $x = x_0 + w$. $x_p = x * p_{42}$ and $x_h = x * h$.

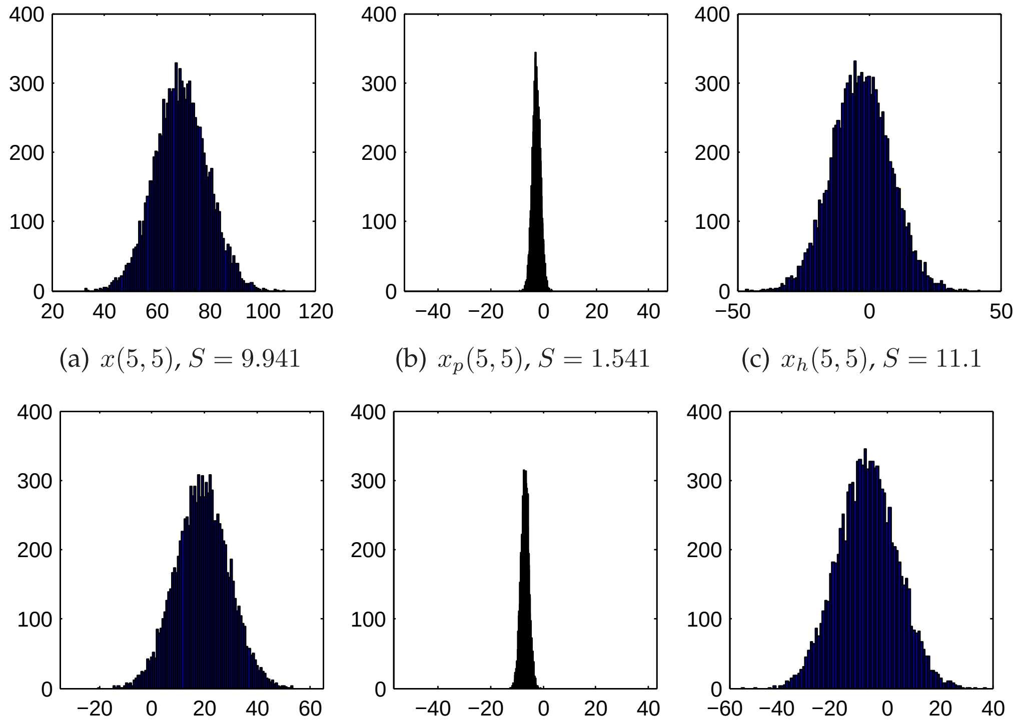
Similarly, we obtain

$$Var(x_{hi}) = Var(w_{hi}) = \sigma_w^2 \sum_n h_n^2 = \sigma_w^2 \left(1 + 4\left(-\frac{1}{4}\right)^2\right) = 1.25\sigma_w^2 \tag{4.10}$$

where σ_w^2 is the variance of the additive Guassian noise present in the input image. Therefore,

$$\begin{aligned} \sigma_{x_p} &= 0.153\sigma_w \\ \sigma_{x_h} &= 1.118\sigma_w. \end{aligned} \tag{4.11}$$

The variance of x_h is much higher than the variance of x_p. This result is verified in Figure 4.6.



(a) $x(5,5)$, $S = 9.941$ (b) $x_p(5,5)$, $S = 1.541$ (c) $x_h(5,5)$, $S = 11.1$

(d) $x(8,8)$, $S = 10.02$ (e) $x_p(8,8)$, $S = 1.51$ (f) $x_h(8,8)$, $S = 11.26$

Figure 4.6: Histograms of values of a single pixel in x, x_p and x_h from Figure 4.5. Histograms are based on 10000 realizations of white Gaussian noise with $\sigma_w = 10$. Histograms have the same scale on the x-axis for easier comparison.

4.3 Minimizing the Energy of Detail Coefficients

The image x_h represents detail coefficients resulting from dual wavelet with two vanishing moments as stated in Section 2.3.3. Our goal is to minimize those detail coefficients by changing the value of the p parameter. The reason for minimizing detail coefficients is to obtain a more compact image representation. We have chosen to minimize the energy of detail coefficients since it leads to the well-known least squares solution. The adaptation will not necessary be performed over the whole image. The parts of the image for which the energy of details obtained with a fixed predictor is negligible will remain unchanged.

4.3.1 Least Squares Solution for the Parameter p

The filters used in the prediction branch are Neville filters of degree $\tilde{N}$. Hence, if the input to the filter bank is a polynomial of degree lower than $\tilde{N}$, the prediction will be perfect, leading to zero detail coefficients. On the contrary, the prediction will not be perfect for the components of the input signal that are not polynomials (or that are polynomials of higher degree), and the prediction error will be represented with wavelet detail coefficients. Our goal is to improve the prediction by additionally capturing those non-polynomial components. It would lead to a smaller detail coefficients and finally to a more compact image representation.

The prediction error for the i-th pixel is defined as

$$e_i = x_{hi} - \hat{x}_{hi} \tag{4.12}$$

where

$$\hat{x}_{hi} = p \cdot x_p \tag{4.13}$$

The adaptation of the parameter p will be performed for every pixel of the analyzed image based on a given neighborhood as shown in Figure 4.7. A given pixel and its neighborhood form a region on which the value of the parameter p will be calculated. If a pixel is indexed with r then the appropriate region will be denoted as $\mathcal{R}_r$. For every region $\mathcal{R}_r$ the adaptation algorithm will set the value of the parameter p_r in order to minimize the sum of squared errors. So, the function

$$F(p_r) = \sum_{i \in \mathcal{R}_r} e_i^2 = \sum_{i \in \mathcal{R}_r} (x_{hi} - p_r \cdot x_{pi})^2 \tag{4.14}$$

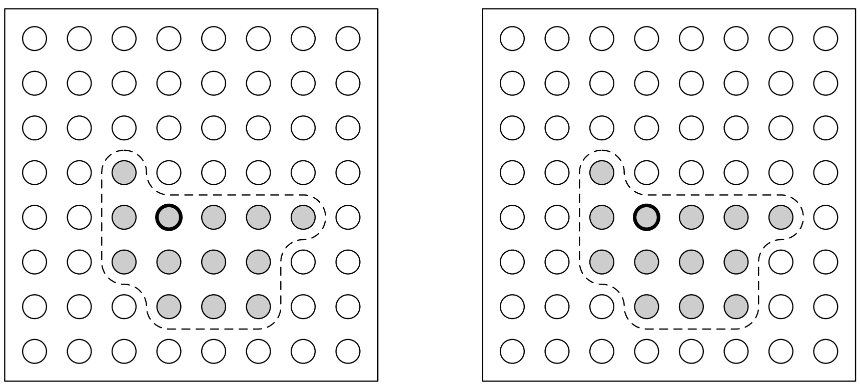
needs to be minimized. Calculating the least squares (LS) solution leads to minimizing the energy of detail coefficients. The optimal parameter p_r is the one for which the derivative of $F(p_r)$ equals zero.

$$
\begin{aligned}
\frac{\partial F(p_r)}{\partial p_r} = \sum_{i \in \mathcal{R}_r} 2(x_{hi} - p_r \cdot x_{pi})(-x_{pi}) = 0 \\
\sum_{i \in \mathcal{R}_r} (x_{hi} - p_r \cdot x_{pi}) x_{pi} = 0 \\
\sum x_{hi} \cdot x_{pi} - \sum p_r \cdot y_i^2 = 0 \\
p_r \sum y_i^2 = \sum x_{hi} \cdot x_{pi}
\end{aligned}
\tag{4.15}
$$

which gives

$$
p_r = \frac{\sum_{i \in \mathcal{R}_r} x_{hi} \cdot x_{pi}}{\sum_{i \in \mathcal{R}_r} x_{pi}^2}
\tag{4.16}
$$

For simplicity of notation, the index r will be omitted in the following equations.



(a) Region in x_p.

(b) Region in x_h.

Figure 4.7: Value of the parameter p for a pixel represented with a bold-line circle will be calculated based on the neighborhood of pixels marked gray.

4.3.2 Parameters Affecting the Estimation of p

As stated in equation 4.16, the parameter p is calculated as a quotient of the two functions of region elements in x_p and x_h. As stated in equation 4.8 every x_{pi} has its deterministic component x_{0pi} and a random component w_{pi}. The random variable w_{pi} is a function of 12 independent random variables of the additive noise present in the input image from the constellation given in Figure 4.7(a):

$$
\begin{aligned}
w_{pi} = (w * p_{42})_i =& \frac{2}{32}(w_{i2} + w_{i3} + w_{i4} + w_{i5}) \\
&- \frac{1}{32}(w_{i6} + w_{i7} + w_{i8} + w_{i9} + w_{i10} + w_{i11} + w_{i12} + w_{i13}).
\end{aligned}
\tag{4.17}
$$

In a similar manner x_{hi} consists of a deterministic component x_{0hi} and a random variable w_{hi} where w_{hi} is a weighted sum of 5 independent random variables

$$w_{hi} = (w * h)_i = w_{i1} - \frac{1}{4}(w_{i2} + w_{i3} + w_{i4} + w_{i5}) \tag{4.18}$$

Therefore, the w_{pi} and w_{hi} are dependent random variables, and the summations in numerator and denominator of p are also dependent random variables. There are various factors influencing the statistical properties of the estimated p.

Input noise. Statistical properties of the input noise w directly influence the properties of the p estimator.

Shape of the region. Various shapes of the region on which the LS solution is being calculated result in different constellations of the input noise samples included in the calculation and therefore different statistical properties of the calculated p parameters.

Value of the original image. The value of the original image x_0 significantly affects the distribution of the calculated p parameter. In general, the value of the original image is not known in denoising applications. Figure 4.8 represents histograms of the calculated p parameters on the region of the same size but on different parts of the analyzed image, including different values of the original image.

4.3.3 The Estimate of p is Biased

As it can be seen from Figure 4.8 the calculated values of p are biased. The bias of p is defined as

$$bias(p) = E[p - \hat{p}] = p - E\left[\frac{\sum_i x_{hi} \cdot x_{pi}}{\sum_i x_{pi}^2}\right] \tag{4.19}$$

As shown in Figure 4.10 the estimation bias depends on the variance of the input noise, the values of the original image and the size (and shape) of the region on which the LS solution is being calculated. If one would know the estimation bias than an unbiased estimator of p could be calculated as

$$\hat{p}_u = \hat{p} + bias(p) \tag{4.20}$$

Unfortunately, there is no closed form expression for the expectation of the estimated p and the value of the original image is unknown in image denoising applications so the unbiased estimator is not available.

4.4 Solution for the Biasness Problem

The true value of the parameter p is obtained as a function of the original image pixels only, i.e.

$$p = \frac{\sum_i x_{0hi} \cdot x_{0pi}}{\sum_i x_{0pi}^2} = \frac{q(x_0)}{r(x_0)}, \tag{4.21}$$

where q represents the numerator and r represents the denominator. The estimate of p is obtained as a function of the image pixels corrupted with noise

$$\hat{p} = \frac{\sum_i x_{hi} \cdot x_{pi}}{\sum_i x_{pi}^2} = \frac{\hat{q}(x_0, w)}{\hat{r}(x_0, w)} \tag{4.22}$$

In the following sections we show that the biases of estimating q and r have closed form expressions and *do not depend on the values of the original image*. Therefore, we will introduce an improved estimator $\hat{p}_c$ which is obtained as

$$\hat{p}_c = \frac{\hat{q}(x_0, w) + bias(q)}{\hat{r}(x_0, w) + bias(r)} = \frac{\hat{q}_c}{\hat{r}_c} \tag{4.23}$$

and which is quite different from the unbiased estimator $\hat{p}_u$ but, as it will be shown, for a class of input images, it gives less biased values of the parameter p than the basic LS estimator given in 4.16.

4.4.1 Bias of the Numerator

Bias of the numerator q is

$$bias(q) = E[q - \hat{q}] = q - E[\hat{q}] \tag{4.24}$$

where the expectation of $\hat{q}$

$$\begin{aligned}
E[\hat{q}] = E\left[\sum x_{hi}x_{pi}\right] &= E\left[\sum (x_{0hi} + w_{hi})(x_{0pi} + w_{pi})\right] \\
&= E\left[\sum x_{0hi}x_{0pi} + x_{0hi}w_{pi} + w_{hi}x_{0pi} + w_{hi}w_{pi}\right] \\
&= \sum \left(x_{0hi}x_{0pi} + x_{0hi}E[w_{pi}] + E[w_{hi}]x_{0pi} + E\left[w_{hi}w_{pi}\right]\right) \\
&= \sum \left(x_{0hi}x_{0pi} + E\left[w_{hi}w_{pi}\right]\right) \\
&= q + \sum E\left[w_{hi}w_{pi}\right]
\end{aligned} \tag{4.25}$$

since $E[w_{pi}] = 0$ and $E[w_{hi}] = 0$. It follows that

$$bias(q) = q - E[\hat{q}] = -\sum_i E\left[w_{hi}w_{pi}\right] \tag{4.26}$$

Since w_{in} are independent and identically distributed random variables with zero mean we have

$$E(w_{im}w_{in}) = \begin{cases} 0 & \text{for } m \neq n \\ \sigma_w^2 & \text{for } m = n \end{cases} \tag{4.27}$$

so by restating equations 4.18 and 4.17

$$\begin{aligned} w_{hi} = (w * h)_i =& w_{i1} - \frac{1}{4}(w_{i2} + w_{i3} + w_{i4} + w_{i5}) \\ w_{pi} = (w * p_{42})_i =& \frac{2}{32}(w_{i2} + w_{i3} + w_{i4} + w_{i5}) \\ & - \frac{1}{32}(w_{i6} + w_{i7} + w_{i8} + w_{i9} + w_{i10} + w_{i11} + w_{i12} + w_{i13}) \end{aligned}$$

we can write

$$\begin{aligned} E\left[w_{hi}w_{pi}\right] &= E\left[-\frac{1}{4}\cdot\frac{2}{32}(w_{i2}^2 + w_{i3}^2 + w_{i4}^2 + w_{i5}^2)\right] \\ &= -\frac{1}{4}\cdot\frac{2}{32}\cdot 4E[w^2] = -\frac{1}{16}\sigma_w^2 \end{aligned} \tag{4.28}$$

Therefore, the summation over the whole region gives

$$\sum_i E\left[w_{hi}w_{pi}\right] = N \cdot E\left[w_{hi}w_{pi}\right] = -\frac{N\sigma_w^2}{16} \tag{4.29}$$

where N is the size of the region. By using equation 4.26 we obtain the bias of q:

$$bias(q) = -\sum_i E\left[w_{hi}w_{pi}\right] = \frac{N\sigma_w^2}{16} \tag{4.30}$$

It can be seen that the estimation of the numerator of p is biased but the value of $bias(q)$ in neither dependent on the value of the original signal x_0 nor the shape of the region. The bias depends only on the size of the region on which the estimation is based and the variance of the additive zero-mean noise present in the image.

4.4.2 Bias of the Denominator

Bias of the denominator r is

$$bias(r) = E[r - \hat{r}] = r - E[\hat{r}] \tag{4.31}$$

where the expectation of $\hat{r}$

$$\begin{aligned} E[\hat{r}] = E\left[\sum x_{pi}^2\right] &= E\left[\sum (x_{0pi} + w_{pi})^2\right] \\ &= E\left[\sum x_{0pi}^2 + 2x_{0pi}w_{pi} + w_{pi}^2\right] \\ &= \sum (x_{0pi}^2 + E[w_{pi}^2]) \\ &= r + \sum E[w_{pi}^2] \end{aligned} \tag{4.32}$$

hence

$$bias(r) = r - E[\hat{r}] = -\sum_i E[w_{pi}^2] \tag{4.33}$$

Since from $E[w_i] = 0$ it follows that $E[w_{pi}] = 0$, by using equation 4.9 we have

$$E[w_{pi}^2] = \sigma_{w_{pi}}^2 = \frac{6}{256}\sigma_w^2 \tag{4.34}$$

The denominator's bias can now be expressed as

$$bias(r) = -\sum_i E[w_{pi}^2] = -\sum_i \frac{6}{256}\sigma_w^2 = -\frac{6N}{256}\sigma_w^2 \tag{4.35}$$

Again, as for the denominator, the estimation of the numerator of p is biased but the value of the bias is neither dependent on the value of the original signal x_0 nor the shape of the region but rather on the size of the region on which the estimation is based and the variance of the additive zero-mean noise present in the image.

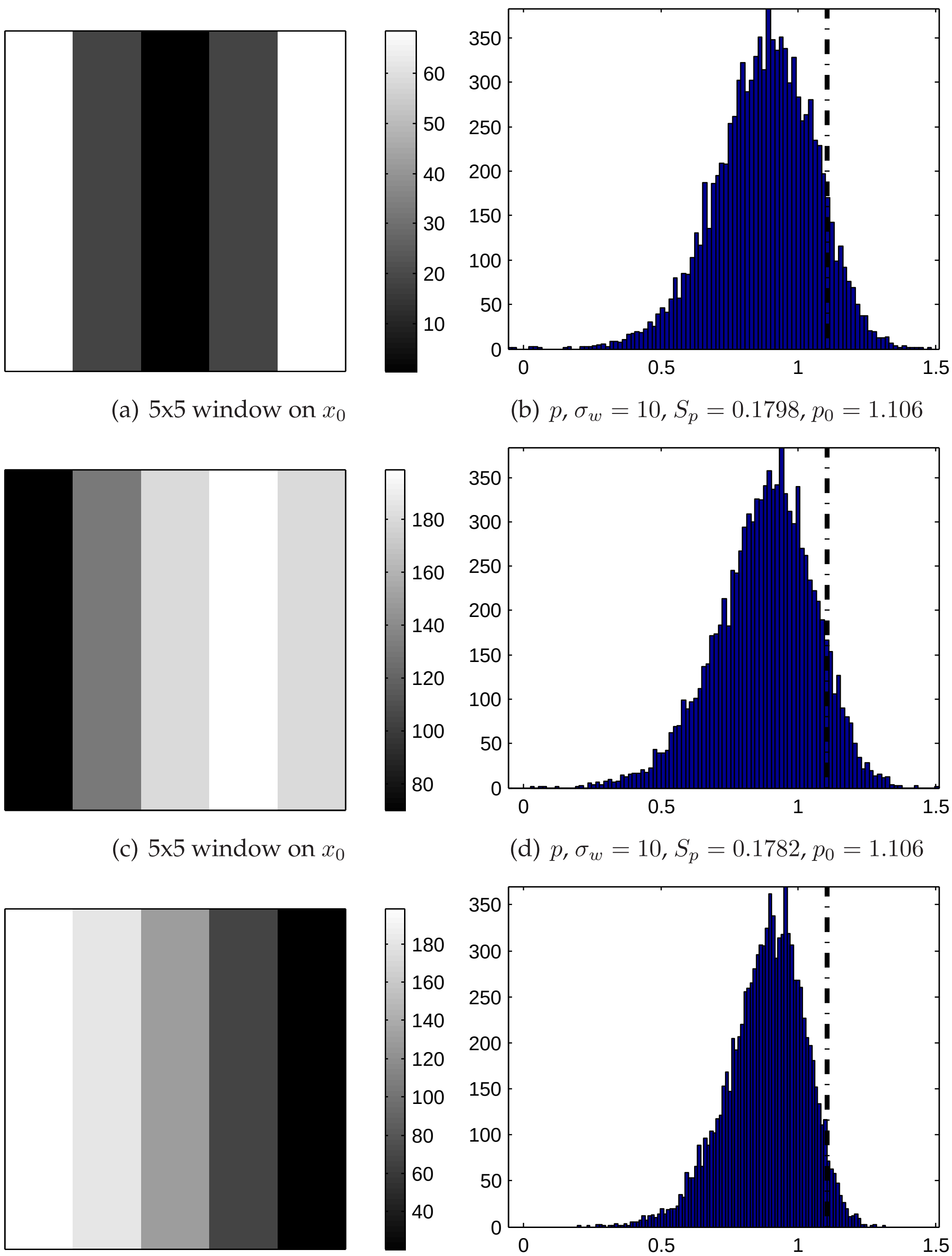
4.4.3 Improved Estimation Formula

Using the relations in 4.30 and 4.35 for the bias of the numerator and denominator respectively, we introduce an improved estimator of the parameter p as

$$\hat{p}_c = \frac{\left(\sum_i x_{hi} \cdot x_{pi}\right) + \frac{N}{16}\sigma_w^2}{\left(\sum_i x_{pi}^2\right) - \frac{6N}{256}\sigma_w^2} \tag{4.36}$$

The $\hat{p}_c$ is not the same as the unbiased estimator $\hat{p}_u$ from equation 4.20 since correcting the bias of numerator and denominator separately is not the same as correcting the overall bias. In practice the $\hat{p}_c$ estimator gives values that are closer to the true value of the parameter p then the ones obtained by using the $\hat{p}$ estimator from equation 4.16. This is true as long as the distribution of corrected denominator has negligible density around 0 (as shown in Figures 4.12 and 4.13). In the case of sine-wave images, distribution of denominator comes around zero for higher values of the sine-wave period T (as shown in Figures 4.14, 4.15 and 4.11), which correspond to low-frequency components present in the image. However, for those low-frequency components the adaptation will not be needed since the signal will already be well suppressed in the high-pass channel by the fixed prediction stage.

Therefore, the adaptation of the filter bank parameters by using the improved estimator $\hat{p}_c$ will not be performed for the whole image. The adaptation will be performed only in the regions which result in higher values of the estimator's denominator. In the regions where the the estimator's denominator is close to zero the adaptation will not be necessary since the signal is already well suppressed in the high-pass channel.



(a) 5x5 window on x_0

(b) p, $\sigma_w = 10$, $S_p = 0.1798$, $p_0 = 1.106$

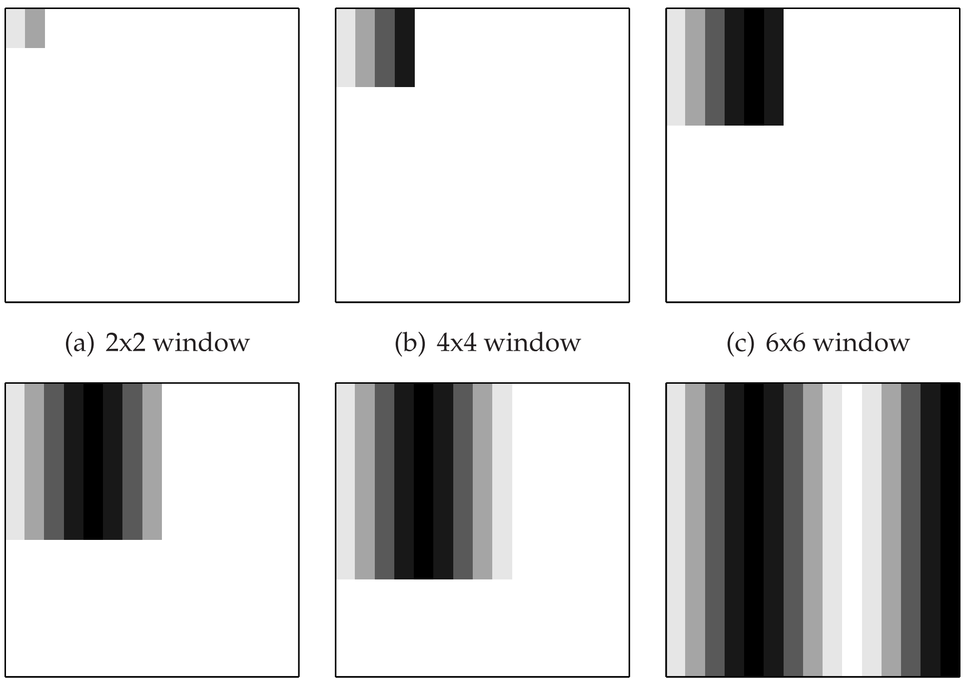
(c) 5x5 window on x_0

(d) p, $\sigma_w = 10$, $S_p = 0.1782$, $p_0 = 1.106$

(e) 5x5 window on x_0

(f) p, $\sigma_w = 10$, $S_p = 0.1415$, $p_0 = 1.106$

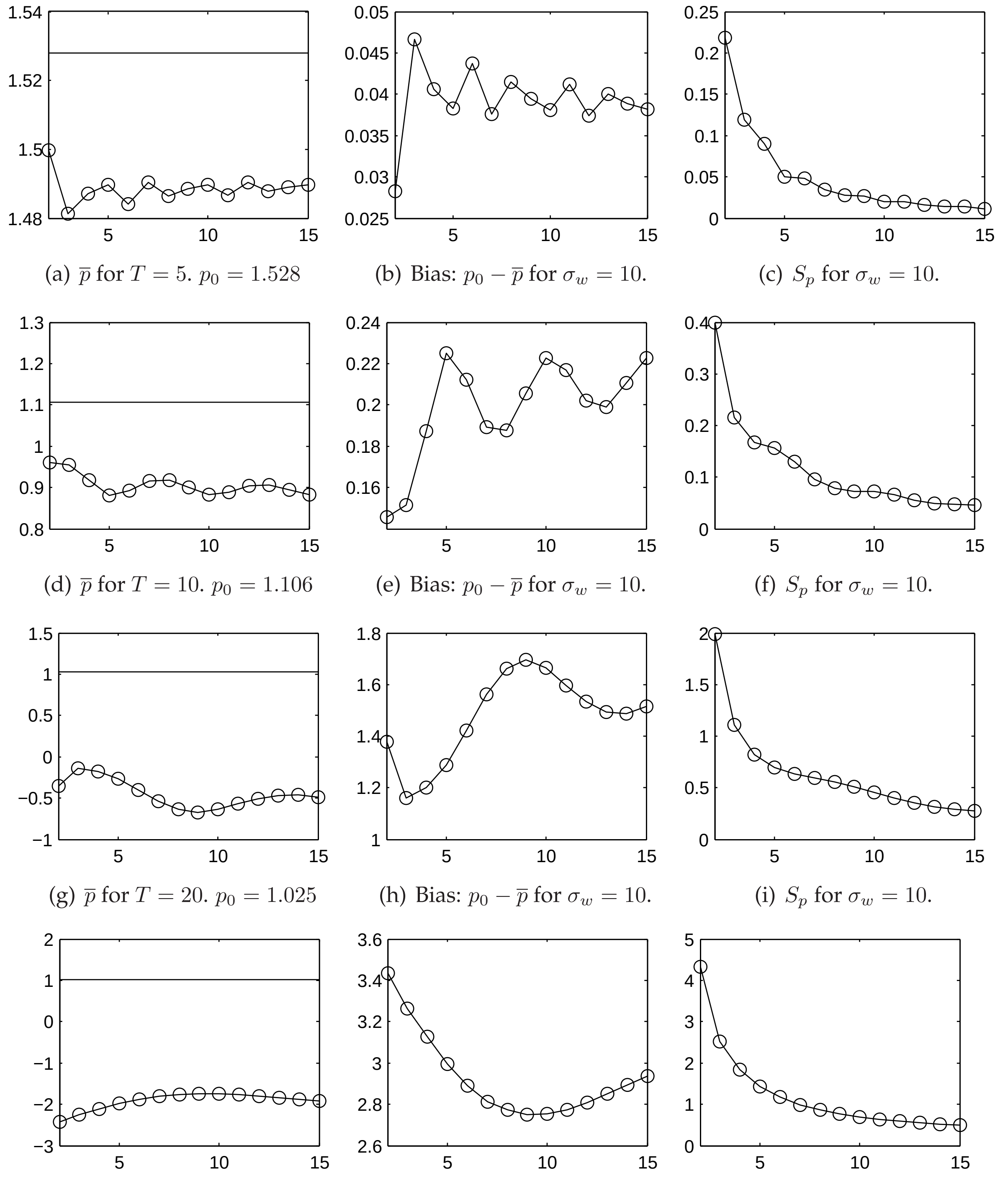
Figure 4.8: Histograms of p obtained for different windows in x_p and x_h from Figure 4.5. Histograms are based on 10000 realizations of white Gaussian noise with $\sigma_w = 10$. True value of p is represented with the vertical dashed line.



(a) 2x2 window (b) 4x4 window (c) 6x6 window

(d) 8x8 window (e) 10x10 window (f) 15x15 window

Figure 4.9: Different window sizes (regions) over the original sine-wave image.



(a) $\overline{p}$ for $T = 5$. $p_0 = 1.528$

(b) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.

(c) S_p for $\sigma_w = 10$.

(d) $\overline{p}$ for $T = 10$. $p_0 = 1.106$

(e) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.

(f) S_p for $\sigma_w = 10$.

(g) $\overline{p}$ for $T = 20$. $p_0 = 1.025$

(h) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.

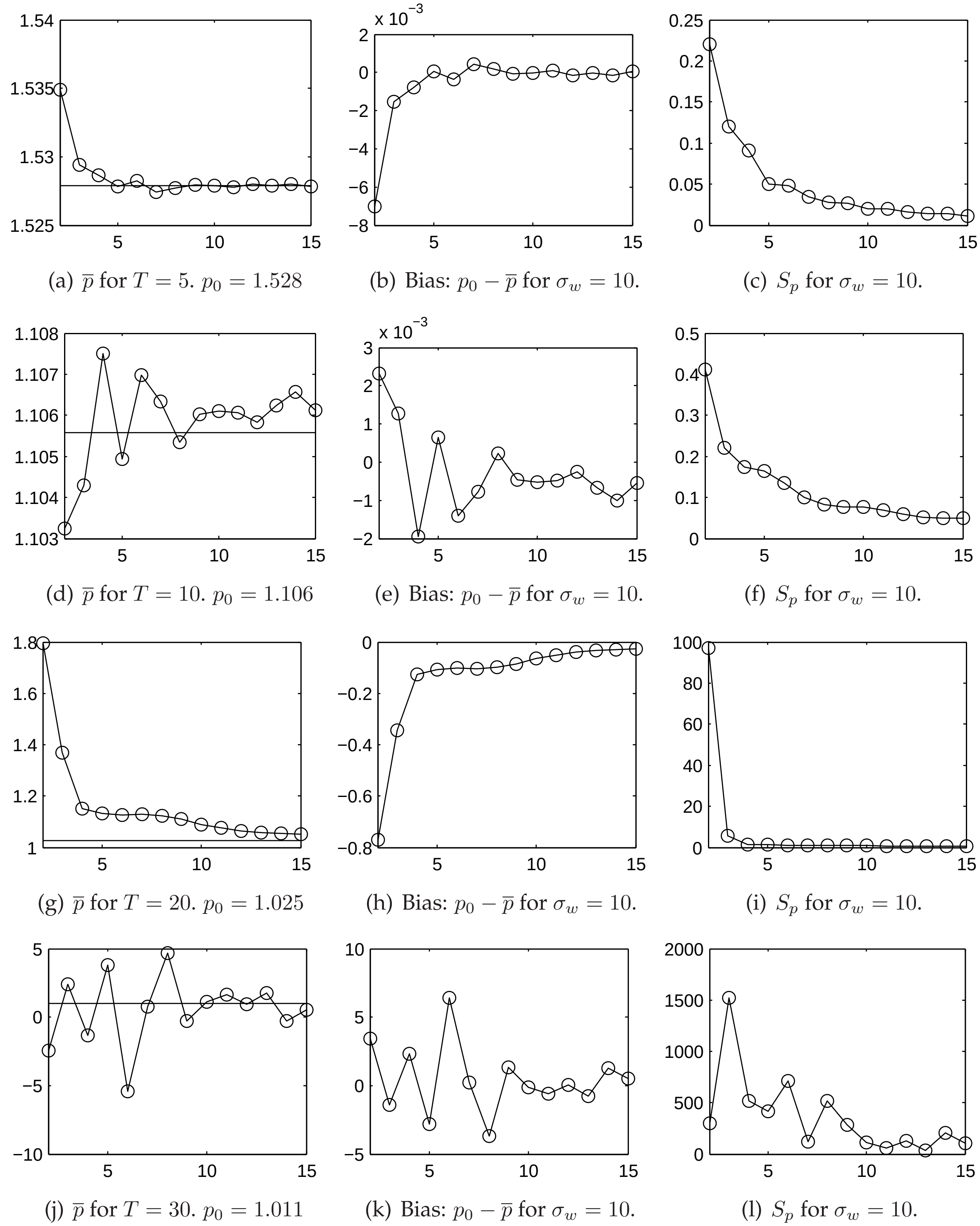
(i) S_p for $\sigma_w = 10$.

(j) $\overline{p}$ for $T = 30$. $p_0 = 1.011$

(k) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.

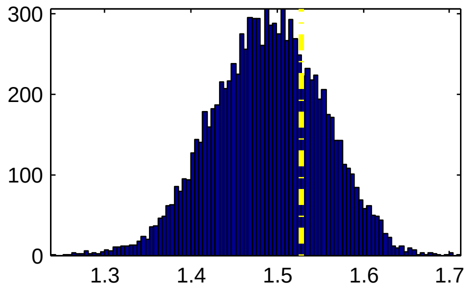
(l) S_p for $\sigma_w = 10$.

Figure 4.10: Bias of p for different window sizes and different sine wave images. The true value of p is shown with a solid horizontal line in figures of the first column. Every row stands for one picture type with a sine wave of different period T. Window size is defined as $n \times n$ where n is the number on the abscissa.

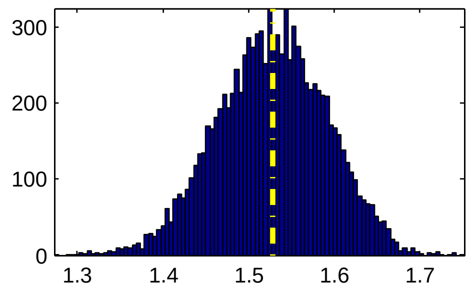


(a) $\overline{p}$ for $T = 5$. $p_0 = 1.528$
(b) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.
(c) S_p for $\sigma_w = 10$.
(d) $\overline{p}$ for $T = 10$. $p_0 = 1.106$
(e) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.
(f) S_p for $\sigma_w = 10$.
(g) $\overline{p}$ for $T = 20$. $p_0 = 1.025$
(h) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.
(i) S_p for $\sigma_w = 10$.
(j) $\overline{p}$ for $T = 30$. $p_0 = 1.011$
(k) Bias: $p_0 - \overline{p}$ for $\sigma_w = 10$.
(l) S_p for $\sigma_w = 10$.

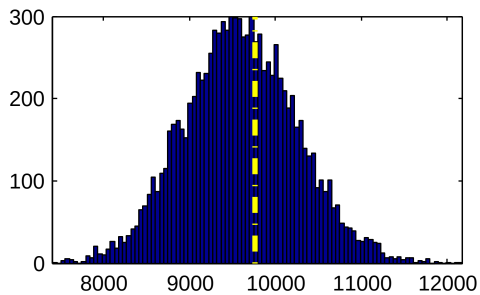
Figure 4.11: Bias of the improved p for different window sizes and different sine wave images. Every row stands for one picture type with a sine wave of different period T. The window size is defined as $n \times n$ where n is the number on the abscissa.



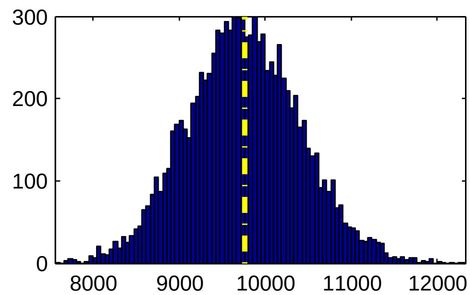
(a) Biased $\hat{p}$, $p = 1.528$, bias $= 0.03885$



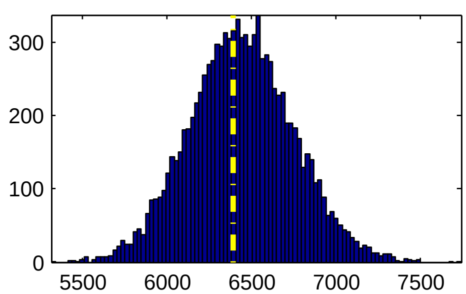
(b) Improved $\hat{p}_c$, $p = 1.528$, bias $= 0.0006281$



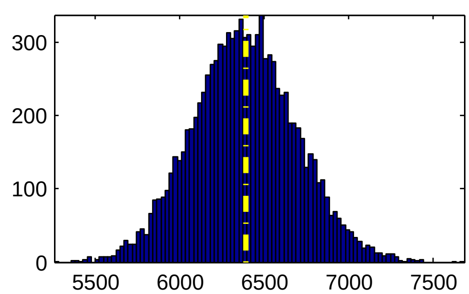
(c) Biased $\hat{q}$, $q = 9766$, bias $= 167.8$



(d) Unbiased $\hat{q}_c$, $q = 9766$, bias $= 11.57$

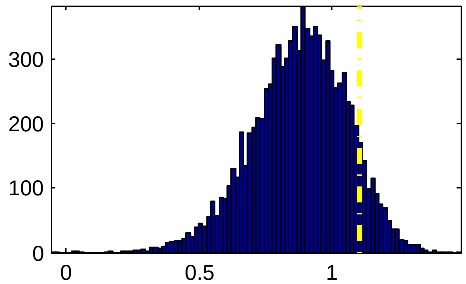


(e) Biased $\hat{r}$, $r = 6392$, bias $= -52.22$

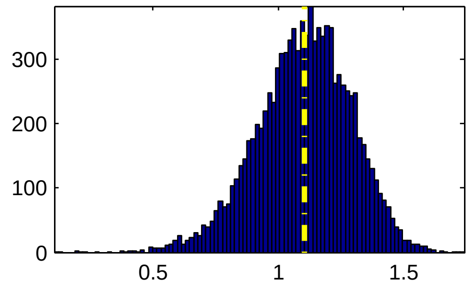


(f) Unbiased $\hat{r}_c$, $r = 6392$, bias $= 6.369$

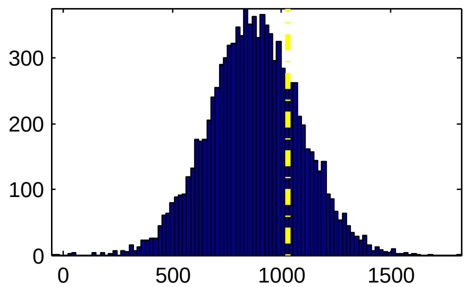
Figure 4.12: Results on calculating the parameter p. First column: biased p and its numerator and denominator. Second column: improved p obtained with unbiased numerator and denominator. Region size: 5×5. Image: sine-wave with $T = 5$. Input noise is white Gaussian zero-mean noise with $\sigma_w = 10$.



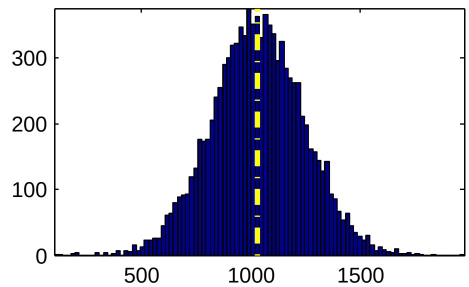
(a) Biased $\hat{p}$, $p = 1.106$, bias $= 0.2273$



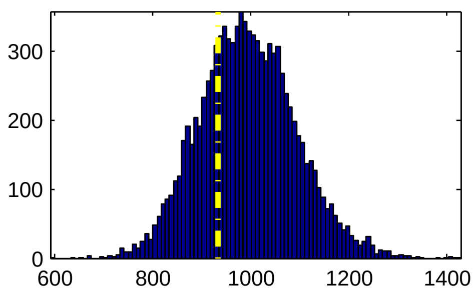
(b) Improved $\hat{p}_c$, $p = 1.106$, bias $= 0.003061$



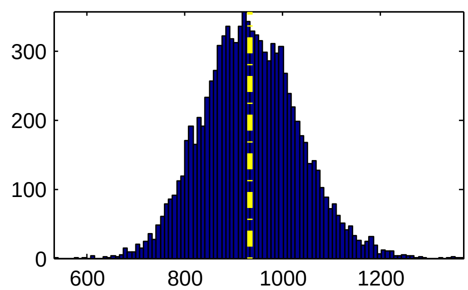
(c) Biased $\hat{q}$, $q = 1031$, bias $= 154.8$



(d) Unbiased $\hat{q}_c$, $q = 1031$, bias $= -1.421$

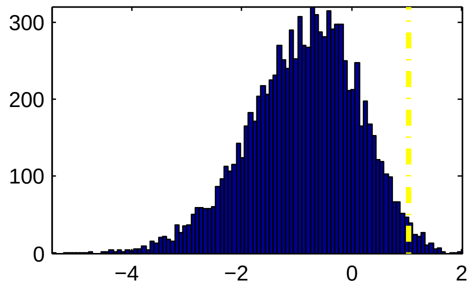


(e) Biased $\hat{r}$, $r = 932.5$, bias $= -61.57$

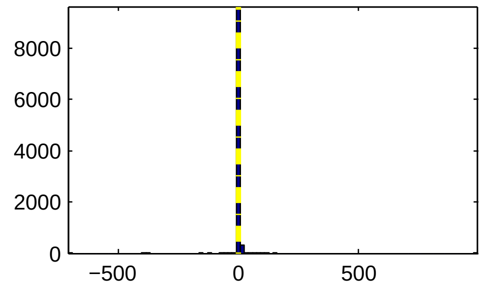


(f) Unbiased $\hat{r}_c$, $r = 932.5$, bias $= -2.977$

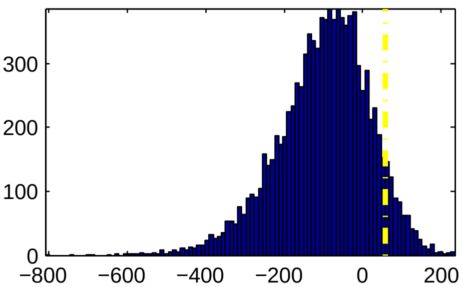
Figure 4.13: Results on calculating the parameter p. First column: biased p and its numerator and denominator. Second column: improved p obtained with unbiased numerator and denominator. Region size: 5×5. Image: sine-wave with $T = 10$. Input noise is white Gaussian zero-mean noise with $\sigma_w = 10$.



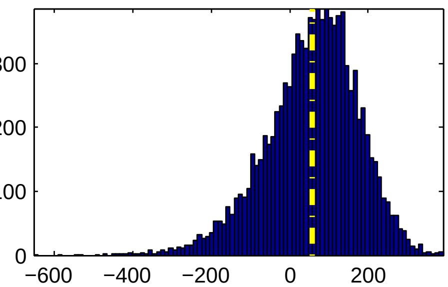
(a) Biased $\hat{p}$, $p = 1.025$, bias $= 1.904$



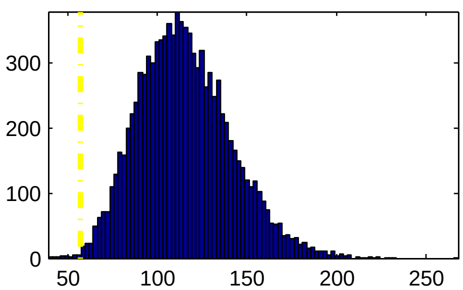
(b) Improved $\hat{p}_c$, $p = 1.025$, bias $= -0.5285$



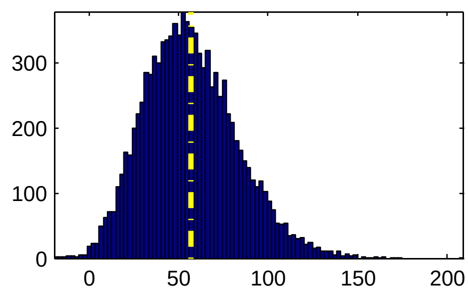
(c) Biased $\hat{q}$, $q = 58.42$, bias $= 158.4$



(d) Unbiased $\hat{q}_c$, $q = 58.42$, bias $= 2.189$

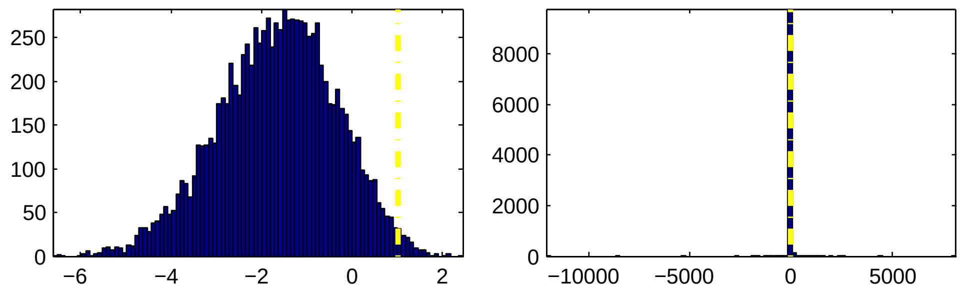


(e) Biased $\hat{r}$, $r = 56.99$, bias $= -58.61$

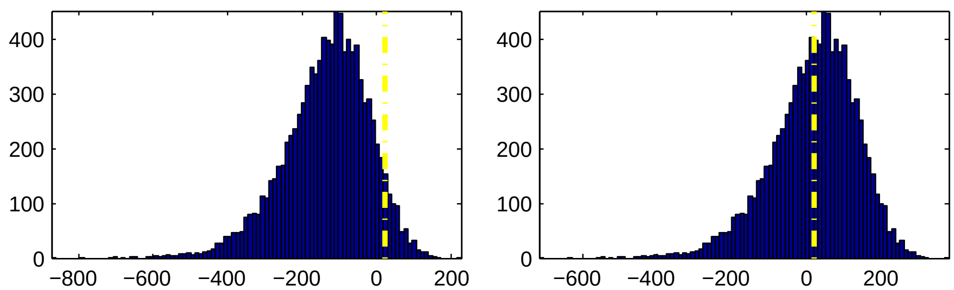


(f) Unbiased $\hat{r}_c$, $r = 56.99$, bias $= -0.01347$

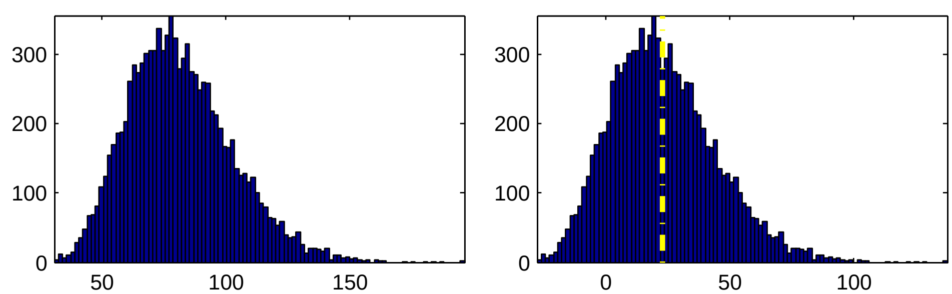
Figure 4.14: Results on calculating the parameter p. First column: biased p and its numerator and denominator. Second column: improved p obtained with unbiased numerator and denominator. Region size: 5×5. Image: sine-wave with $T = 20$. Input noise is white Gaussian zero-mean noise with $\sigma_w = 10$.



(a) Biased $\hat{p}$, $p = 1.011$, bias $= 2.666$ (b) Improved $\hat{p}_c$, $p = 1.011$, bias $= 0.5978$



(c) Biased $\hat{q}$, $q = 22.99$, bias $= 158.1$ (d) Unbiased $\hat{q}_c$, $q = 22.99$, bias $= 1.807$



(e) Biased $\hat{r}$, $r = 22.74$, bias $= -58.96$ (f) Unbiased $\hat{r}_c$, $r = 22.74$, bias $= -0.368$

Figure 4.15: Results on calculating the parameter p. First column: biased p and its numerator and denominator. Second column: improved p obtained with unbiased numerator and denominator. Region size: 5×5. Image: sine-wave with $T = 30$. Input noise is white Gaussian zero-mean noise with $\sigma_w = 10$.

4.5 Adaptation in the PUPa structure

The expressions derived for the PPaU structure in the previous sections will now be derived for the PUPa structure. The adaptive PUPa analysis filter bank is shown in Figure 4.16 and the equivalent simplified structure is shown in Figure 4.17.

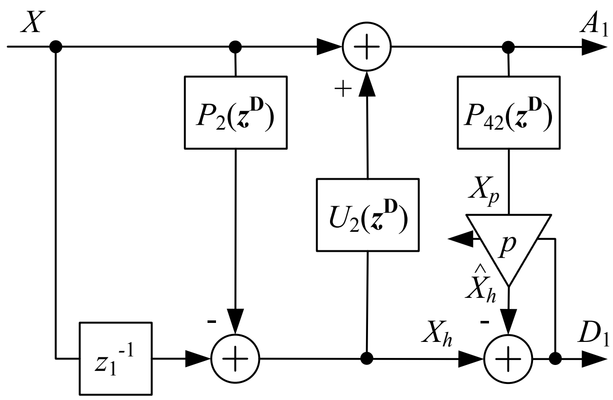


Figure 4.16: The first decomposition level of the adaptive PUPa filter bank where $P_{42} = P_4 - P_2$.

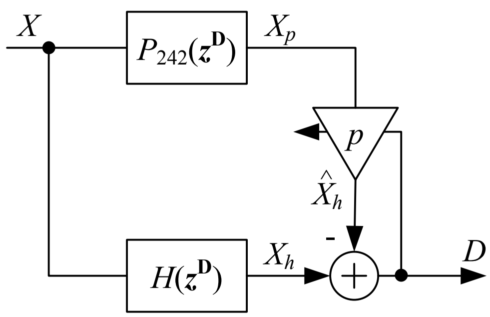


Figure 4.17: Simplified adaptive prediction stage.

The impulse response of the filter P_{242} is

$$p_{242} = \frac{1}{1024}\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 3 & -4 & 2 & -4 & 3 & 0 & 0 \\ 0 & 3 & -8 & -31 & 0 & -31 & -8 & 3 & 0 \\ 1 & -4 & -31 & 8 & 52 & 8 & -31 & -4 & 1 \\ 0 & 2 & 0 & 52 & 32 & 52 & 0 & 2 & 0 \\ 1 & -4 & -31 & 8 & 52 & 8 & -31 & -4 & 1 \\ 0 & 3 & -8 & -31 & 0 & -31 & -8 & 3 & 0 \\ 0 & 0 & 3 & -4 & 2 & -4 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \tag{4.37}$$

The obtained impulse response is significantly wider than the one of the PPaU structure from equation 4.5. The image obtained after applying the fixed prediction branch to the input image is denoted as x_h. It is a result of a convolution of the original image with the high-pass filter H whose impulse response is

$$h = \begin{pmatrix} 0 & -\frac{1}{4} & 0 \\ -\frac{1}{4} & 1 & -\frac{1}{4} \\ 0 & -\frac{1}{4} & 0 \end{pmatrix} \tag{4.38}$$

The variances of the x_{pi} and x_{hi} pixels are

$$Var(x_{pi}) = Var(w_{pi}) = \sigma_w^2 \sum_n p_{242n}^2 = 0.0193\sigma_w^2 \tag{4.39}$$

and

$$Var(x_{hi}) = Var(w_{hi}) = \sigma_w^2 \sum_n h_n^2 = \sigma_w^2 \left(1 + 4\left(-\frac{1}{4}\right)^2\right) = 1.25\sigma_w^2. \tag{4.40}$$

Similarly as in section 4.4.1, the bias of the numerator can be obtained as

$$bias(q) = -\sum_i E\left[w_{hi}w_{pi}\right], \tag{4.41}$$

where

$$\begin{aligned} E\left[w_{hi}w_{pi}\right] &= E\left[-\frac{1}{4}\cdot\frac{52}{1024}(w_{i2}^2 + w_{i3}^2 + w_{i4}^2 + w_{i5}^2)\right] \\ &= -\frac{1}{4}\cdot\frac{13}{256}\cdot 4E[w^2] = -0.0508\sigma_w^2, \end{aligned} \tag{4.42}$$

which is independent on the position i and the values of the original image x_0. Therefore,

$$bias(q) = -N \cdot E\left[w_{hi}w_{pi}\right] = 0.0508N\sigma_w^2. \tag{4.43}$$

Using the fact that $E[w_{pi}] = 0$, the bias of the denominator is obtained similarly as in section 4.4.2:

$$bias(r) = -\sum_i E[w_{pi}^2] = -N \cdot Var(w_{pi}) = -0.0193N\sigma_w^2. \tag{4.44}$$

Hence, the improved estimation of the parameter p can be obtained as

$$\hat{p}_c = \frac{\hat{q}(x_0, w) + bias(q)}{\hat{r}(x_0, w) + bias(r)} = \frac{\left(\sum x_{hi}\cdot x_{pi}\right) + 0.0508N\sigma_w^2}{\left(\sum x_{pi}^2\right) - 0.0193N\sigma_w^2} \tag{4.45}$$

4.6 Adaptation in the Iterated Filter Bank

4.6.1 The Iterated PPaU Structure

The results in the previous sections were presented for a single-level filter bank structure. However, in practice, four to eight-level iterated filter banks are used. The first two levels of the iterated PPaU structure are shown in Figure 4.18.

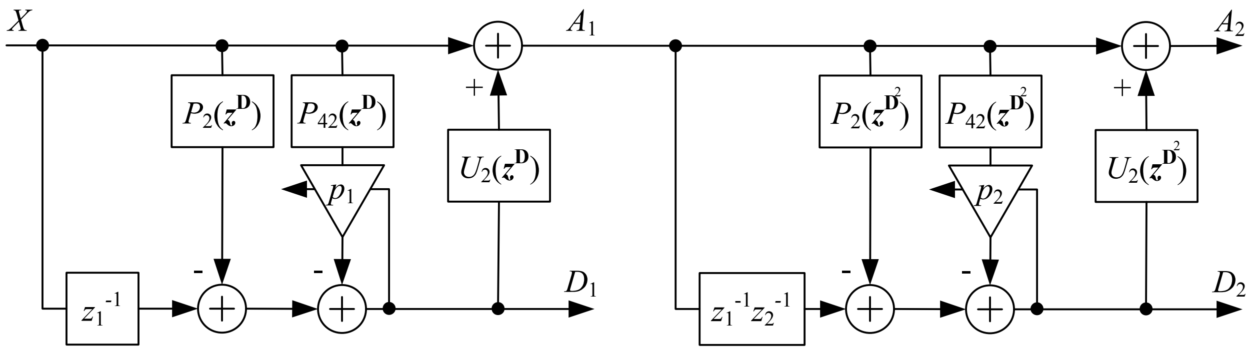


Figure 4.18: The first two decomposition levels of the adaptive PPaU filter bank.

The adaptation of the parameter p_j in the j-th level will be based on the corresponding P_{42j} and H_j filters. Therefore, the p_{42j} will be the impulse response of the j-level filter bank structure up to the input to the multiplier p_j. Also, the h_j will be the impulse response of the j-level filter bank structure up to the summation operator at the output of the p_j multiplier, similar as in Figure 4.3.

It is important to note that the adaptation in lower levels affects the P_{42j} and H_j filters in all the successive levels. Since the adaptation is being performed pixelwise, the impulse responses p_{42ji} and h_{ji} will be different for every pixel position i. Therefore, the bias correction in level j will have different values for every pixel of the image, depending on the adaptation in the previous $j-1$ levels. The estimation of the parameter p at level j and for the position i will be obtained as

$$\hat{p}_{cji} = \frac{\hat{q}_{ji}(x_0, w) + bias(q_{ji})}{\hat{r}_{ji}(x_0, w) + bias(r_{ji})} \tag{4.46}$$

The numerator's bias is obtained as

$$bias(q_{ji}) = -\sum_i E\left[w_{hji} w_{pji}\right] = -N\sigma_w^2 \sum_{m=n} h_{jim} p_{42jin} \tag{4.47}$$

and the denominator's bias is obtained as

$$bias(r_{ji}) = -\sum_i E\left[w_{pji}^2\right] = -N\sigma_w^2 \sum_n p_{42jin}^2 \tag{4.48}$$

where N is the number of pixels included in the region.

4.6.2 The Iterated PUPa Structure

The iterated PUPa filter bank structure is shown in Figure 4.19. The advantage of the PUPa structure is that P_{242j} and H_j filters at level j do not depend on the adaptation in the lower decomposition levels. Therefore, the nonlinearity introduced with the adaptation in level j is not propagated to successive decomposition levels.

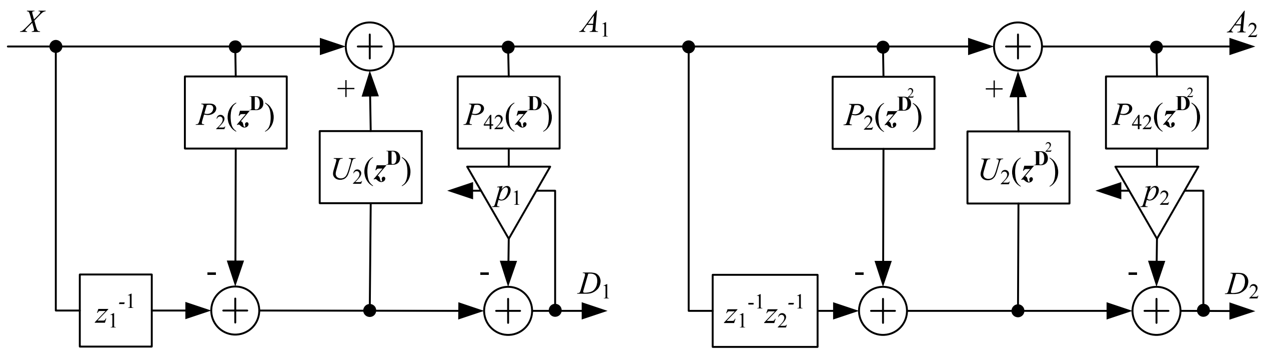


Figure 4.19: The first two decomposition levels of the adaptive PUPa filter bank.

The estimation of the parameter p at level j will be obtained as

$$\hat{p}_{cj} = \frac{\hat{q}_j(x_0, w) + bias(q_j)}{\hat{r}_j(x_0, w) + bias(r_j)} \tag{4.49}$$

The numerator's bias is obtained as

$$bias(q_j) = -\sum_i E\left[w_{hji} w_{pji}\right] = -N\sigma_w^2 \sum_{m=n} h_{jm} p_{242jn} \tag{4.50}$$

and the denominator's bias is obtained as

$$bias(r_j) = -\sum_i E\left[w_{pji}^2\right] = -N\sigma_w^2 \sum_n p_{242jn}^2 \tag{4.51}$$

where N is the number of pixels included in the region.

Chapter 5

Finding the Appropriate Adaptation Region

5.1 Introduction

In the previous chapter we have presented a technique for estimating the parameter p. The parameter p for a single image pixel is being calculated based on a region. As it was shown in Figure 4.11, the standard deviation of $\hat{p}$ decreases as the region size increases. So, in order to obtain more reliable estimates we want to have the parameter p calculated on a region that is as big as possible but still not incorporating parts of the image with different statistical properties.

In order to obtain such regions we will use a statistical method called Intersection of Confidence Intervals (ICI) rule [Goldenshluger 97, Katkovnik 98, Stankovic 98]. The ICI method will allow us to obtain better and more reliable estimates of the parameter p. A short introduction to confidence intervals that are used in this chapter is given is Appendix.

5.2 The Intersection of Confidence Intervals Rule

5.2.1 Overlapping Confidence Intervals

The intersection of confidence intervals rule [Katkovnik 99, Stankovic 04] is based on a number of estimates of some parameter or a signal value. The estimates are based on different window sizes. We define a set of growing window sizes:

$$\mathbb{H} = \{h_j | h_j > h_{j-1}, j = 1, 2, \ldots, J\} \tag{5.1}$$

The window h_{j-1} is be a subset of the next window h_j. The J estimates are calculated for J successive windows. Now we define confidence intervals of the estimate $\hat{x}_j$ as

$$D_j = [\hat{x}_j(t) - \Gamma\sigma_{\hat{x}_j(t)}, \hat{x}_j(t) + \Gamma\sigma_{\hat{x}_j(t)}] \tag{5.2}$$

where Γ is an empirically set constant that defines the width of the confidence interval. In this section D_j will be used for the j-th confidence interval and should not be confused with wavelet detail coefficients.

Let us consider an original two-valued signal corrupted with zero-mean Gaussian noise. A value of a signal or some parameter of the signal is calculated based on a set of growing regions. The wider estimation regions result in a more efficient estimator with smaller variance. On the other hand, if growing windows start covering parts of the signal with different value of the estimate x then the bias will start increasing, as shown in Figure 5.1.

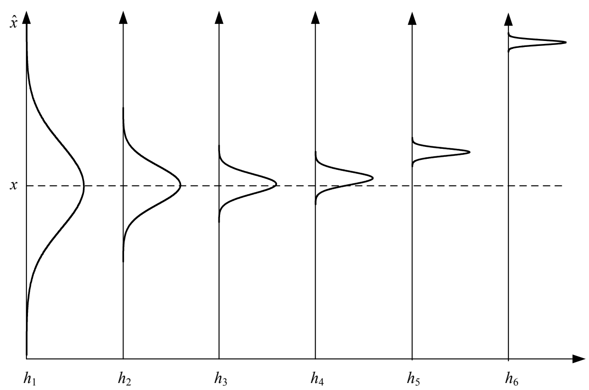


Figure 5.1: Probability distribution functions of $\hat{x}$ for growing window sizes h_i.

5.2.2 The ICI algorithm

The ICI algorithm starts with the smallest window size and calculates the appropriate confidence interval D_1. Then, the next estimate is calculated and the appropriate confidence interval D_2 is obtained. The intersection of confidence intervals for j successive estimates is obtained as

$$I_j = \cap_{i=1}^{j} D_i \tag{5.3}$$

One wants to obtain the estimate that is based on the biggest possible window size but still statistically related to the previous estimates. In terms of confidence intervals this means that the optimal scale that will be denoted as j^+ will be the largest of indices for which $I_j \neq \emptyset$. The ICI algorithm consists of the following steps:

First estimate. Obtain the estimate $\hat{x}$ for the smallest window. Calculate L_1 and U_1 as the lower and the upper bound of the confidence interval.

Successive estimates. Calculate successive estimates based on the growing windows. Track the value of the highest lower confidence interval bound L_{max} and the smallest upper confidence interval bound U_{min}.

Stop enlarging the window. If $L_{max} > U_{min}$, there is no intersection of confidence intervals. The optimal window index j^+ is the one from the previous step.

Select the estimate. The final estimate will be the one obtained with window size h_{j^+}. An example of intersecting confidence intervals is shown in Figure 5.2(b). In the figure, $j^+ = 6$, since for $j = 7$ there is no intersection of confidence intervals.

5.2.3 Motivational Example: Improving the Estimator of the Mean Value

In this section we present a method to improve the estimator of a mean value for purposes of signal denoising. Instead of using a fixed moving average filter, for each signal sample, a moving average filter with appropriate order will be chosen. The ICI rule will be used in order to select the appropriate averaging window.

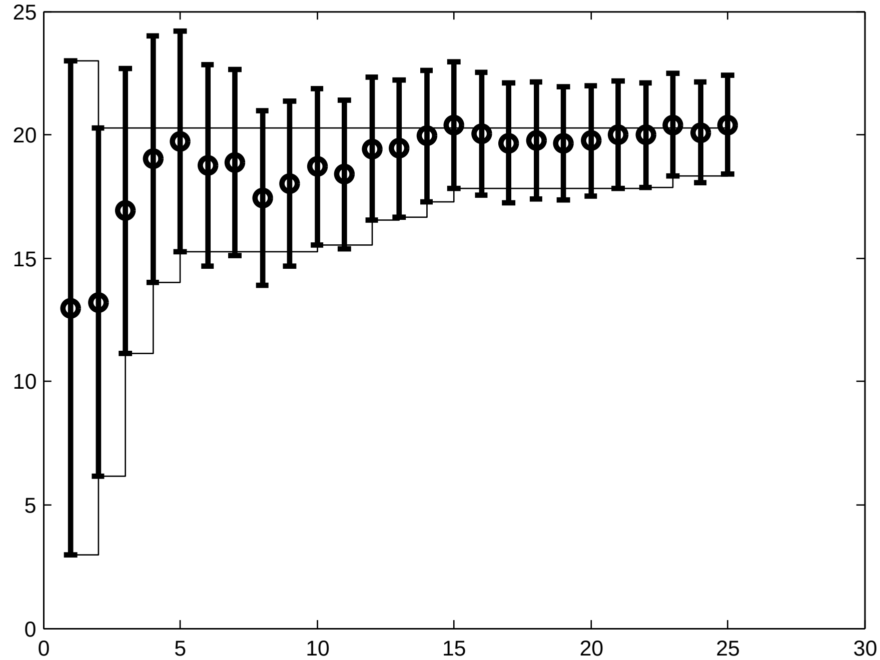
The ICI method will use a number of estimates based on different overlapping supports. The first mean estimate can be obtained with a filter with an impulse response $h[n] = \{\underline{\frac{1}{2}}, \frac{1}{2}\}$, where the underline denotes the sample at $n = 0$. Then, the second estimator could have $h[n] = \{\underline{\frac{1}{3}}, \frac{1}{3}, \frac{1}{3}\}$, the third estimator could have $h[n] = \{\underline{\frac{1}{4}}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\}$ and so on. For each estimate a confidence interval whose width is proportional to the standard deviation of the noise is built. The confidence interval for a mean of n samples is obtained as

$$\bar{x} \pm z\frac{\sigma}{\sqrt{n}} \tag{5.4}$$

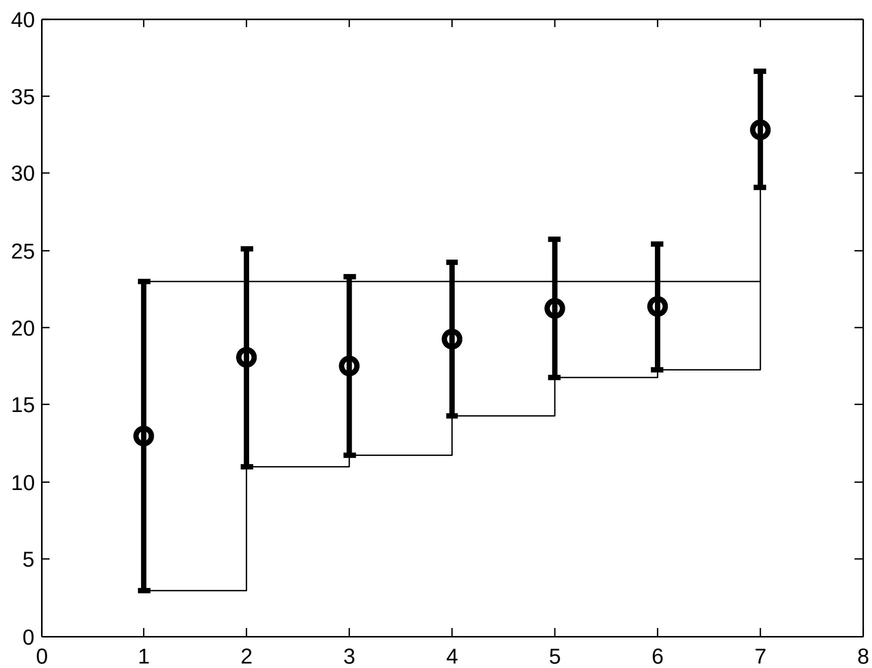
where σ is the standard deviation of the random variable x. For example, the 95% confidence interval (CI) is obtained for $z = 1.96$. For every new estimate the algorithm tracks the intersection of confidence intervals. The support of the moving average filter is growing as long as there is an intersection of a current CI with all the previous CIs (Figure 5.2).

Additionally, we have chosen to grow averaging windows in both directions from a given sample. The ICI rule will give the optimal window sizes for both sides. Then, the two windows will merge in one and the final average value for a given sample will be calculated. An example of ICI based averaging of 1D signal is given in Figure 5.3.

It can be seen that discontinuities in the signal are much better preserved by using ICI based averaging than by using fixed window averaging.

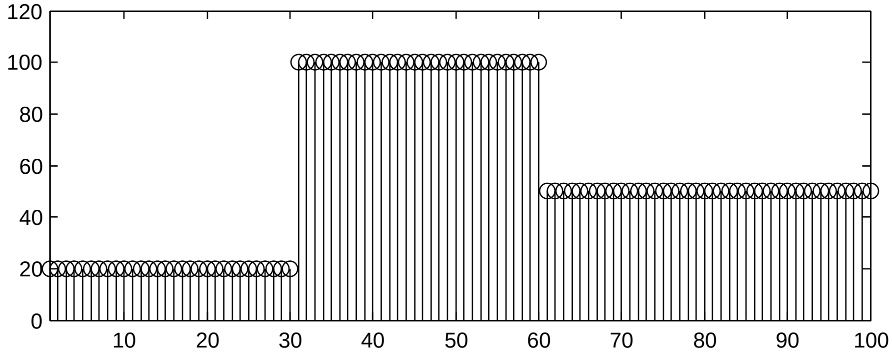


(a) Estimates based on windows growing to the left from the given sample.

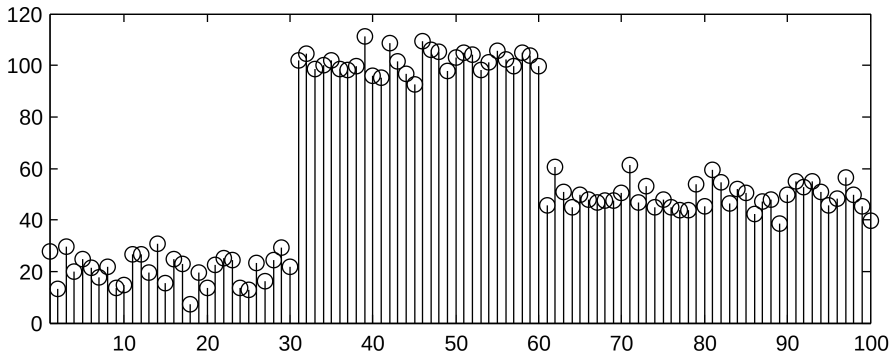


(b) Estimates based on windows growing to the right from the given sample. For the 7th estimate there is no more intersection of CIs. So, the final estimate is the one obtained is step 6.

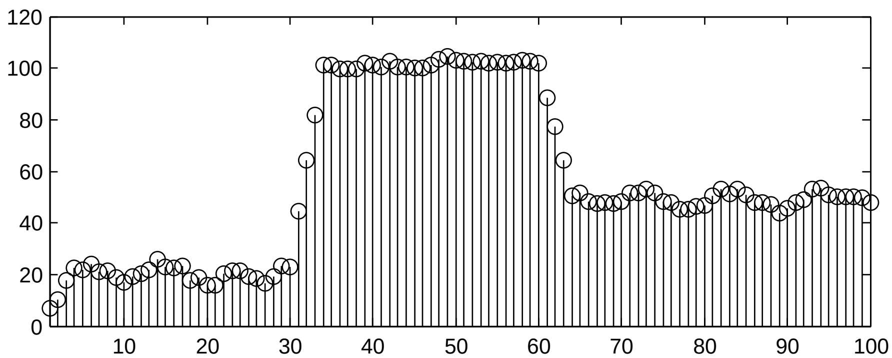
Figure 5.2: Intersections of confidence intervals for estimating $x(25)$ from Figure 5.3(b).



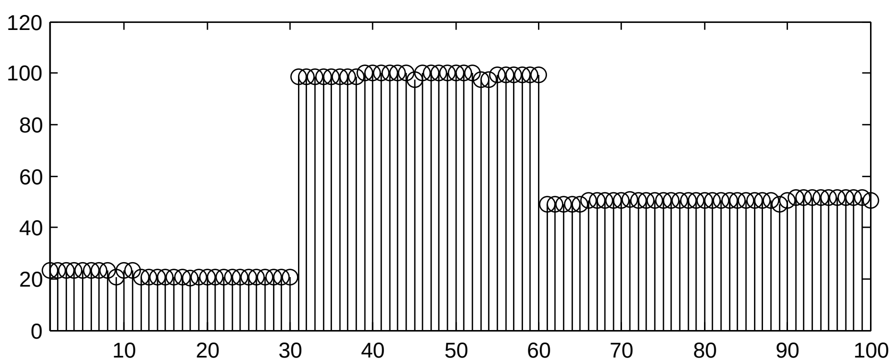
(a) Original signal



(b) Noisy signal: Gaussian noise with $\sigma = 5$ added to the original signal.



(c) Denoising with moving average filter $h = \{\underline{0.25}, 0.25, 0.25, 0.25\}$.



(d) Denoising with the ICI adaptive moving average filter.

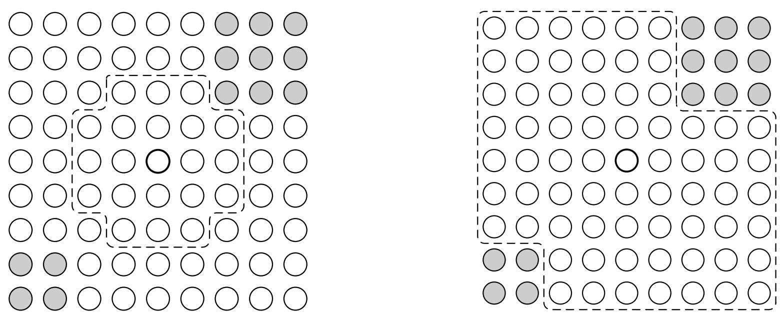
Figure 5.3: Mean estimate improved with the ICI rule.

5.3 ICI-based Estimation of the Parameter p

We will use the ICI algorithm to improve the properties of the estimator of the parameter p. The example from the previous section will be generalized to the two-dimenisonal case in order to obtain two-dimensional regions adapted to local image statistics. For every pixel, in order to obtain a more efficient estimate, the parameter p will be calculated on a different adaptive neighborhood selected using the ICI rule.

5.3.1 Wedge-Shaped Regions

The possible solution for different window sizes are concentric circles that grow from the pixel for which the adaptation is being performed. However, such adaptive regions will not be able to "avoid" parts of the image of different properties as shown in Figure 5.4(a). The ideal region would be the one shown in Figure 5.4(b).

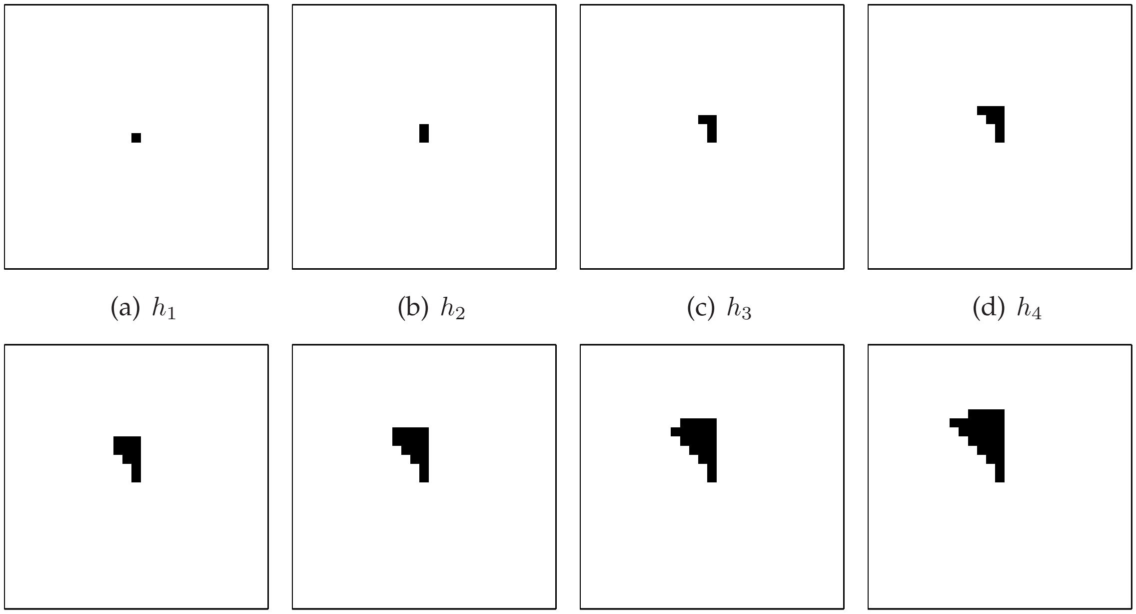


(a) The widest circular region. (b) Ideally estimated adaptive region.

Figure 5.4: Pixels of different values of the ideal parameter p are marked with different colors. Dashed line represents the adaptive region boundary.

In order to obtain regions closer to the one from Figure 5.4(b) the wedge-like regions will be grown from the central pixels into different directions. Later on, when each wedge grows to its maximum allowed size, the overall region will be obtained as a combination of all wedge-like regions. The wedge of angle $2\pi/8$ is used in order to obtain sufficient angular resolution. Therefore, 8 wedges are grown in 8 distinct directions in order to obtain an anisotropic adaptation region. Narrower wedge regions of smaller angles like $2\pi/16$ will not be used since the improvement is not significant.

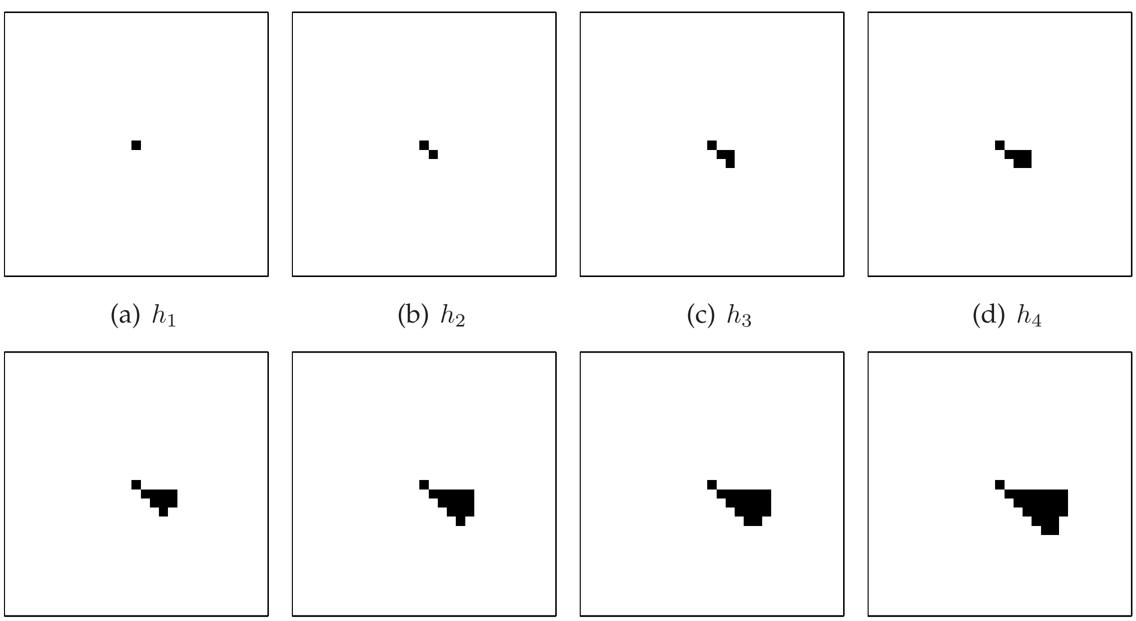
The wedge-shaped windows of growing sizes for a single direction are shown in Figures 5.5 and 5.6. The wedge-shaped windows of the same size for all directions are shown in Figures 5.7 and 5.8. Improved estimates of p are obtained by combining the 8 resulting windows into a single region and calculating $\hat{p}_c$ for the new region.



(a) h_1 (b) h_2 (c) h_3 (d) h_4

(e) h_5 (f) h_6 (g) h_7 (h) h_8

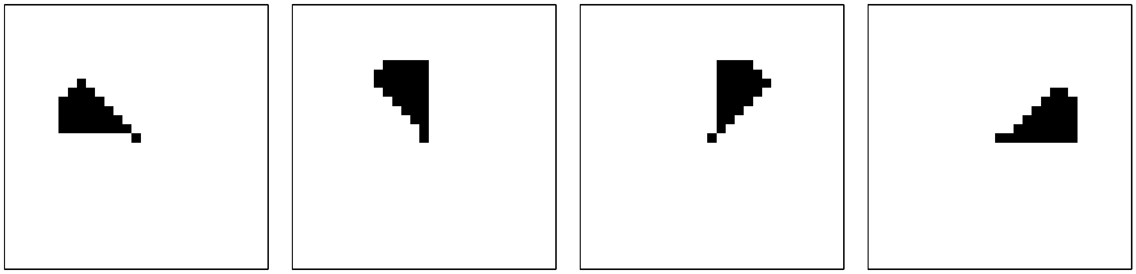
Figure 5.5: Window of different sizes for a single direction.



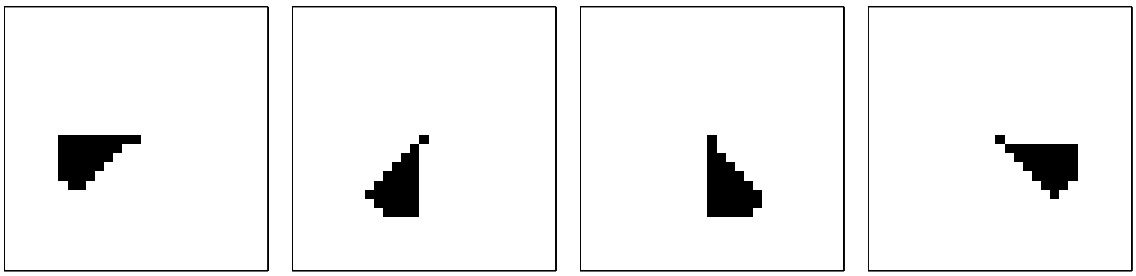
(a) h_1 (b) h_2 (c) h_3 (d) h_4

(e) h_5 (f) h_6 (g) h_7 (h) h_8

Figure 5.6: Window of different sizes for a single direction.

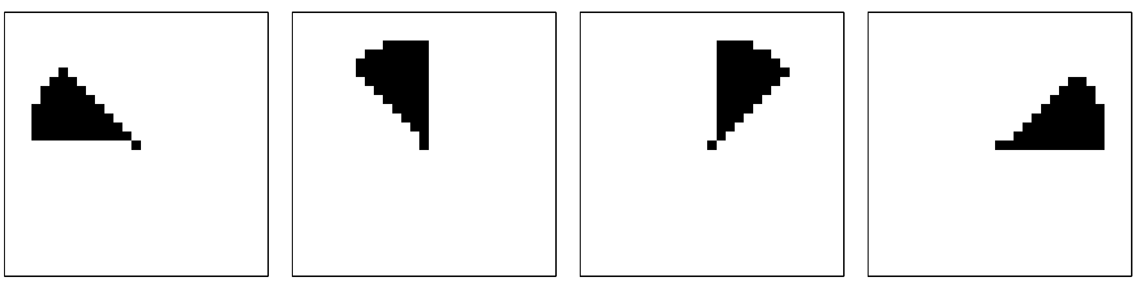


(a) h_9 (b) h_9 (c) h_9 (d) h_9

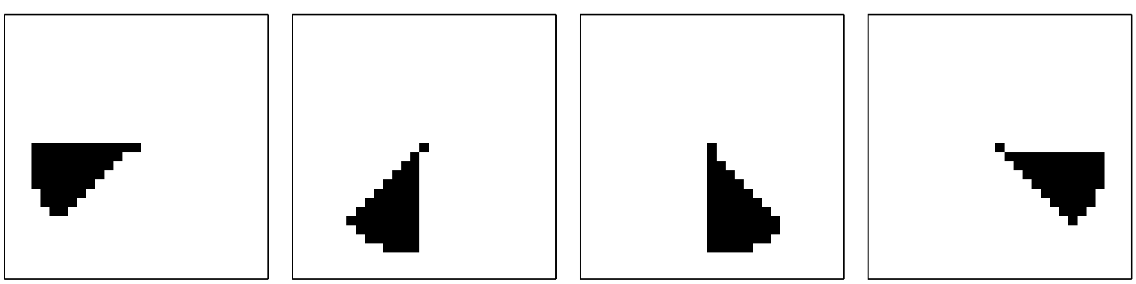


(e) h_9 (f) h_9 (g) h_9 (h) h_9

Figure 5.7: Windows of the same size for different directions.



(a) h_{12} (b) h_{12} (c) h_{12} (d) h_{12}



(e) h_{12} (f) h_{12} (g) h_{12} (h) h_{12}

Figure 5.8: Windows of the same size for different directions.

5.3.2 The Confidence Interval for p

The width of the confidence interval is central to implementing the ICI algorithm. The confidence interval will be proportional to $\sigma_{\hat{p}_c}$ with the constant Γ. In general, higher values of Γ give wider confidence intervals that result in smoother p parameters. Contrary, smaller values of Γ will give sharper edges among regions of various p parameters.

It is important to note that the distribution of $\hat{p}_c$ is slightly asymmetrical (see Figures 5.11, 5.12, and 5.13) so the corresponding confidence intervals are asymmetrical as well (see Appendix).

Figures 5.9 and 5.10 show empirical confidence intervals for estimating the parameter p of the PPaU structure for different periods T of the sine wave present in the original image and for different values of the standard deviation of the input noise σ_w. The sample standard deviation S_p of the parameter $\hat{p}_c$ has been obtained based on 1000 realizations of the noise. The confidence intervals are defined as $\hat{p}_c \pm S_p$.

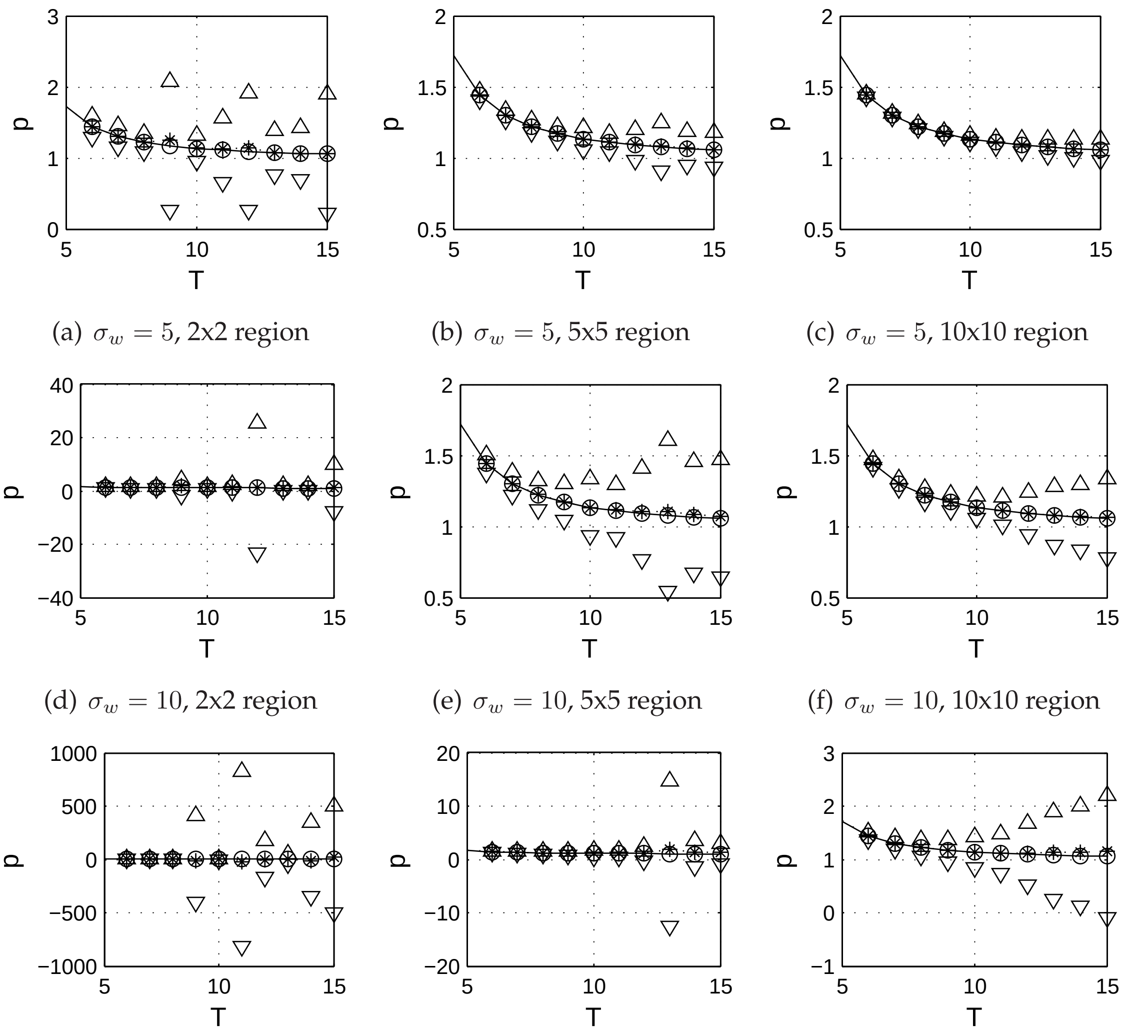
Graphs in the first column of Figure 5.9 are obtained for small regions that contain only 4 pixels. The obtained confidence intervals for such small regions are often very large (see Figure 5.9(g)). In order to obtain a more reliable estimate, wider regions should be used. Figures 5.9(f) and 5.10(b) show that the regions containing 100 pixels are much more acceptable since such regions result in much narrower confidence intervals. It is important to note that the confidence interval width directly depends on the period T of the signal. As the period T increases so does the standard deviation of $\hat{p}_c$.

Figure 5.10(b) shows that the region of size 10×10 allows the adaptation algorithm to differentiate the two sine waves of periods $T_1 = 6$ and $T_2 = 9$ corrupted with zero-mean noise of $\sigma_w = 10$.

Based on the experimental results, and in order to obtain reliable estimates of p, we have chosen that the smallest allowed regions can be either 25, 50, or 100 pixels in size. If the region does not grow over a predefined size, the obtained estimate will not be accepted and the value of p parameter will be set to zero, degrading the structure for that pixel to the fixed filter bank.

In image denoising applications, the standard deviation $\sigma_{\hat{p}_c}$ is unknown since it depends on the unknown original image. Hence, its estimate $\hat{S}_{\hat{p}_c}$ has to be used instead. One possible way to find the estimate of $\sigma_{\hat{p}_c}$ is to use the Taylor series approximation method.

The estimator $\hat{p}_c$ from equation 4.36 can be expressed as a function of n statistically independent random variables of the input noise w_i that are included in the filters'



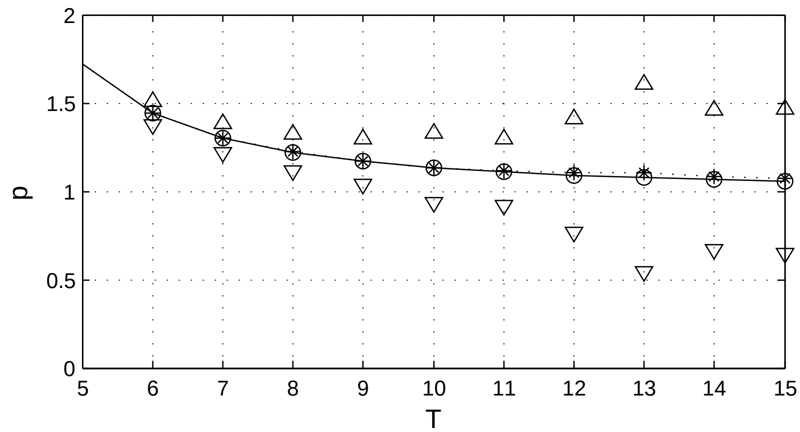
(a) $\sigma_w = 5$, 2x2 region
(b) $\sigma_w = 5$, 5x5 region
(c) $\sigma_w = 5$, 10x10 region
(d) $\sigma_w = 10$, 2x2 region
(e) $\sigma_w = 10$, 5x5 region
(f) $\sigma_w = 10$, 10x10 region
(g) $\sigma_w = 20$, 2x2 region
(h) $\sigma_w = 20$, 5x5 region
(i) $\sigma_w = 20$, 10x10 region

Figure 5.9: Confidence intervals $\hat{p}_c \pm S_p$. Period of the sine wave image is shown on the x-axis. The region size on which the parameter p is being calculated is growing across the rows, while the standard deviation of the input noise σ_w grows with the columns. Circles represent true values of the parameter p and stars represent values of $\hat{p}_c$ averaged over 1000 noise realizations.

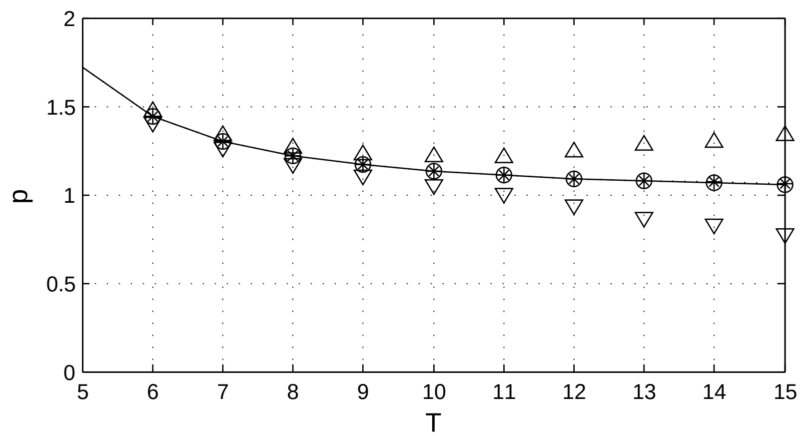
supports for a given region:

$$\hat{p}_c = \frac{\left(\sum_i x_{hi} \cdot x_{pi}\right) + \frac{N}{16}\sigma_w^2}{\left(\sum_i x_{pi}^2\right) - \frac{6N}{256}\sigma_w^2} = g(\mathbf{w}) \tag{5.5}$$

where $\mathbf{w}$ is a vector containing all n random variables w_i included in the calculation of $\hat{p}_c$. Based on the Taylor series expansion, the first-order approximate variance



(a) $\sigma_w = 10$, 5x5 region



(b) $\sigma_w = 10$, 10x10 region

Figure 5.10: Magnified graphs from Figure 5.9 for $\sigma_w = 10$.

[Ayyub 02] can be calculated as

$$\sigma^2_{\hat{p}_c} \approx \sum_{i=1}^{n} \left(\frac{\partial g(\mathbf{w})}{\partial w_i} \bigg|_{E(w_i)=0} \right) \sigma^2_{w_i} \tag{5.6}$$

The results of calculating the approximation of the $\sigma^2_{\hat{p}_c}$ are shown in Table 5.1 with the sample variance $S^2_{\hat{p}_c}$. As presented in the table, the approximation error increases with the region size. Also, the Taylor approximation of $\sigma^2_{\hat{p}_c}$ has been calculated with the values of the original image pixels that are unknown in practice. In practice, these values would be substituted with the values of the preliminary denoised image. Yet, the calculation of partial derivatives needed in equation 5.6 becomes numerically increasingly demanding as the region size increases. Hence, the approach that uses Taylor approximation of the variance will not be used in numerical simulations in the next chapter.

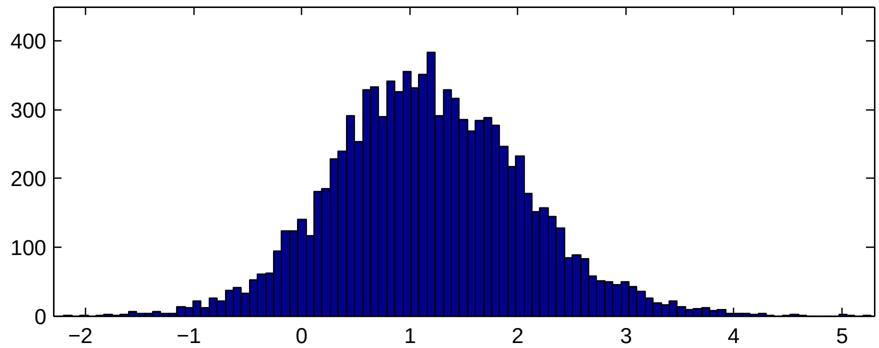


Figure 5.11: Histogram of $\hat{p}_c$ values for a region of size 1×2 with input sine wave image with period $T = 10$ and input noise with standard deviation $\sigma_w = 10$.

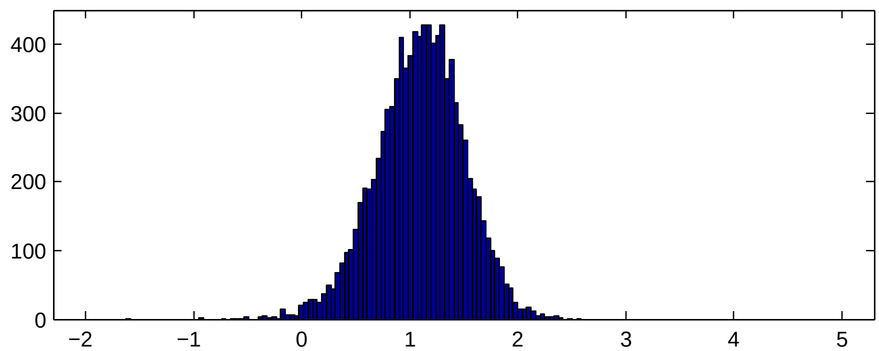


Figure 5.12: Histogram of $\hat{p}_c$ values for a region of size 2×2 with input sine wave image with $T = 10$ and input noise with $\sigma_w = 10$.

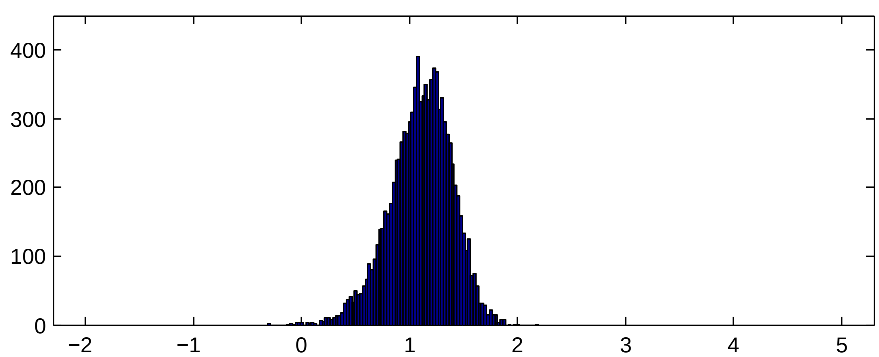


Figure 5.13: Histogram of $\hat{p}_c$ values for a region of size 3×3 with input sine wave image with $T = 10$ and input noise with $\sigma_w = 10$.

Region size	1×2	2×2	3×3
Sample variance	0.7805	0.1648	0.0823
Taylor approximation	0.7099	0.1332	0.0541
Error	-9%	-19%	-34%

Table 5.1: Taylor approximation values for distribution of $\hat{p}_c$ from Figures 5.11, 5.12, and 5.13.

Instead of using Taylor series method to approximate $\sigma_{\hat{p}_c}$, the estimate of the confidence interval width will be obtained empirically:

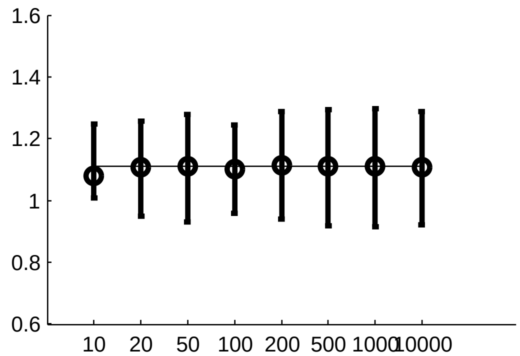
Pilot estimate First, a pilot estimate of the original image will be obtained by denoising the input image by using the wavelet thresholding [Donoho 95a]. For that purpose we will use fixed separable undecimated wavelet decomposition.

Noise realizations Then, a number of realizations of the zero-mean Gaussian noise with standard deviation σ_w will be created and added to the pilot estimate of the original image. This way, a number of noisy images will be obtained with similar statistical properties as the one for which the $\hat{p}_c$ parameter is being calculated.

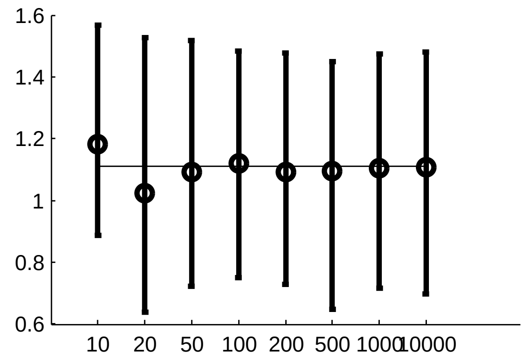
Estimate the CI width For a given region, $\hat{p}_c$ parameter will be calculated for every realization of the noise. The confidence interval for $\sigma_{\hat{p}_c}$ will be estimated based on the histogram of the obtained $\hat{p}_c$ parameters.

For more realizations of the noise, the estimation of $\sigma_{\hat{p}_c}$ will be more accurate. Yet, in order not to make the algorithmic implementation computationally too demanding, the number of noise realization should not be set to high. Figure 5.14 shows empirical confidence interval for $\hat{p}_c$ obtained for various numbers of noise realizations and (values on x-axes) and for regions of different sizes. The most accurate estimate of the confidence interval shown is the one based on 10000 realizations. As the region size increases, the difference between CI estimates decreases. Based on this fact and in order to make the adaptation algorithm feasible in practice, we have chosen to use 20 noise realizations in order to obtain the estimate of the CI width.

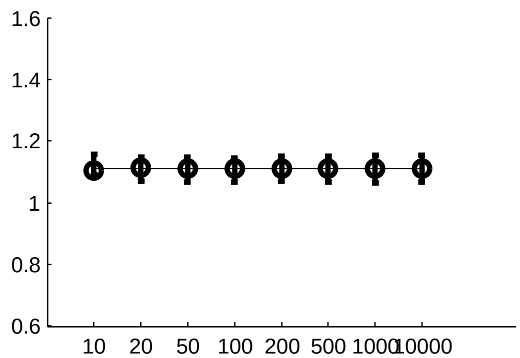
In image denoising applications the standard deviation of the input noise is usually unknown. However, robust algorithm can be used to calculate it, such as median absolute deviation (MAD) of wavelet detail coefficients [Johnstone 97].



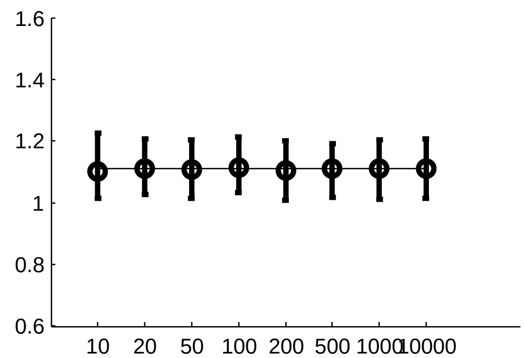
(a) $\sigma_w = 5$. Wedge-shaped window with radius 5 contains 14 pixels.



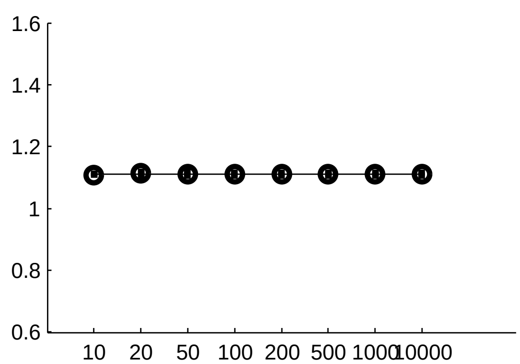
(b) $\sigma_w = 10$. Wedge-shaped window with radius 5 contains 14 pixels.



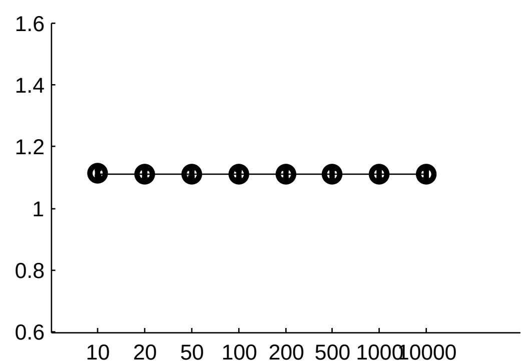
(c) $\sigma_w = 5$. Wedge-shaped window with radius 15 contains 98 pixels.



(d) $\sigma_w = 10$. Wedge-shaped window with radius 15 contains 98 pixels.



(e) $\sigma_w = 5$. Wedge-shaped window with radius 50 contains 1014 pixels.

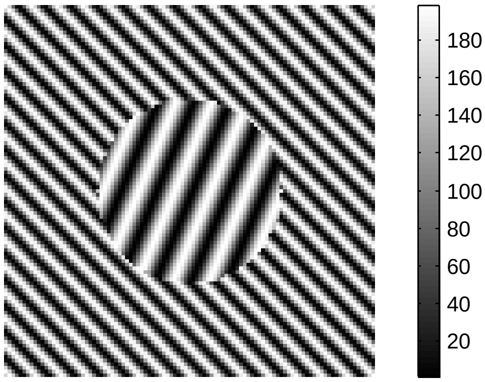


(f) $\sigma_w = 10$. Wedge-shaped window with radius 50 contains 1014 pixels.

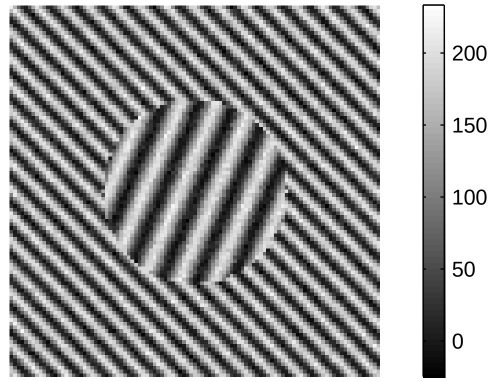
Figure 5.14: Empirical estimation of the 0.9 confidence intervals of $\hat{p}_c$ for sine-wave image with $T = 9$ as a function of number of noise realizations (value on the x-axis) for the PPaU structure. Horizontal line represents the true value of p.

5.3.3 The Resulting Regions

The test image that will be used in examples is an image with two distinct regions containing two different sine waves as shown in Figure 5.15.



(a) The two regions with sines waves with $T_1 = 9$, $\theta_1 = \pi/8$ and $T_2 = 5$, $\theta_1 = -\pi/4$.



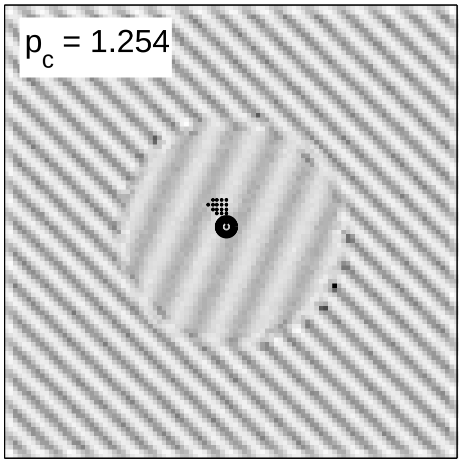
(b) The test image corrupted with additive Gaussian noise with $\sigma_w = 10$.

Figure 5.15: The test image composed of two sine waves.

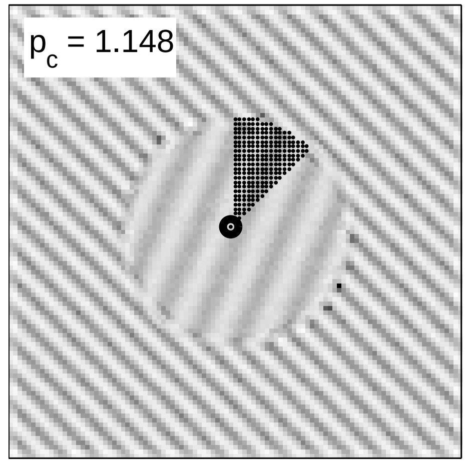
The ICI graphs for $\hat{p}_c$ in the first decomposition level for both PPaU and PUPa structures are shown in figures 5.16 and 5.17. The ICI algorithm resulted with the most appropriate region in Figure 5.16(b). Yet, for a different direction the noise present in the image caused the algorithm to stop growing the window before the boundary of the two sine waves was reached (Figure 5.16(a)). As shown in Figure 5.16(e), the 95% confidence interval for the estimate obtained with the h_5 window did not cover the true value of the parameter p (shown with the horizontal line) and it significantly narrowed the intersections of confidence intervals that followed.

The adaptive regions for the first decomposition level of the PPaU filter bank are shown in Figure 5.18. The similar figures obtained for the PUPa structure are shown in Figure 5.19.

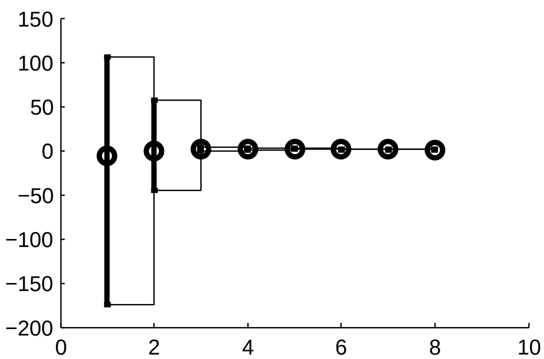
The sensitivity of the ICI algorithm can be increased by using narrower confidence intervals. Figures 5.20 and 5.21 show a region obtained for the same starting pixel but with different CI widths. In general, narrower CIs will cause the region to stop growing earlier, while wider CIs will result in wider regions that possibly spread across the boundaries of the regions with different statistical properties.



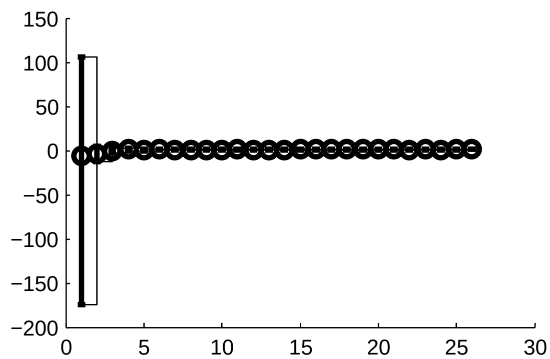
(a) The resulting window h_{j+} for direction 1.



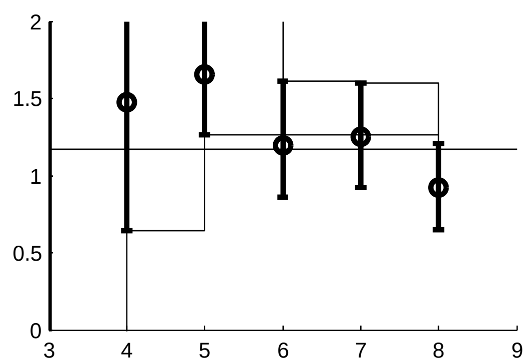
(b) The resulting window h_{j+} for direction 2.



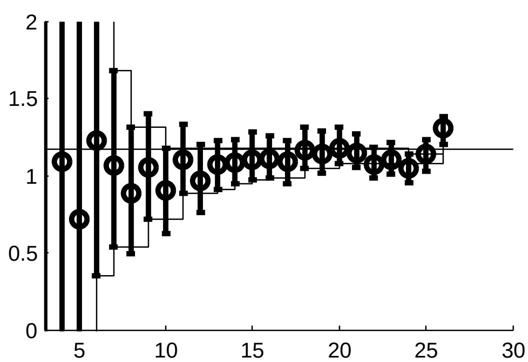
(c) The ICI graph.



(d) The ICI graph.

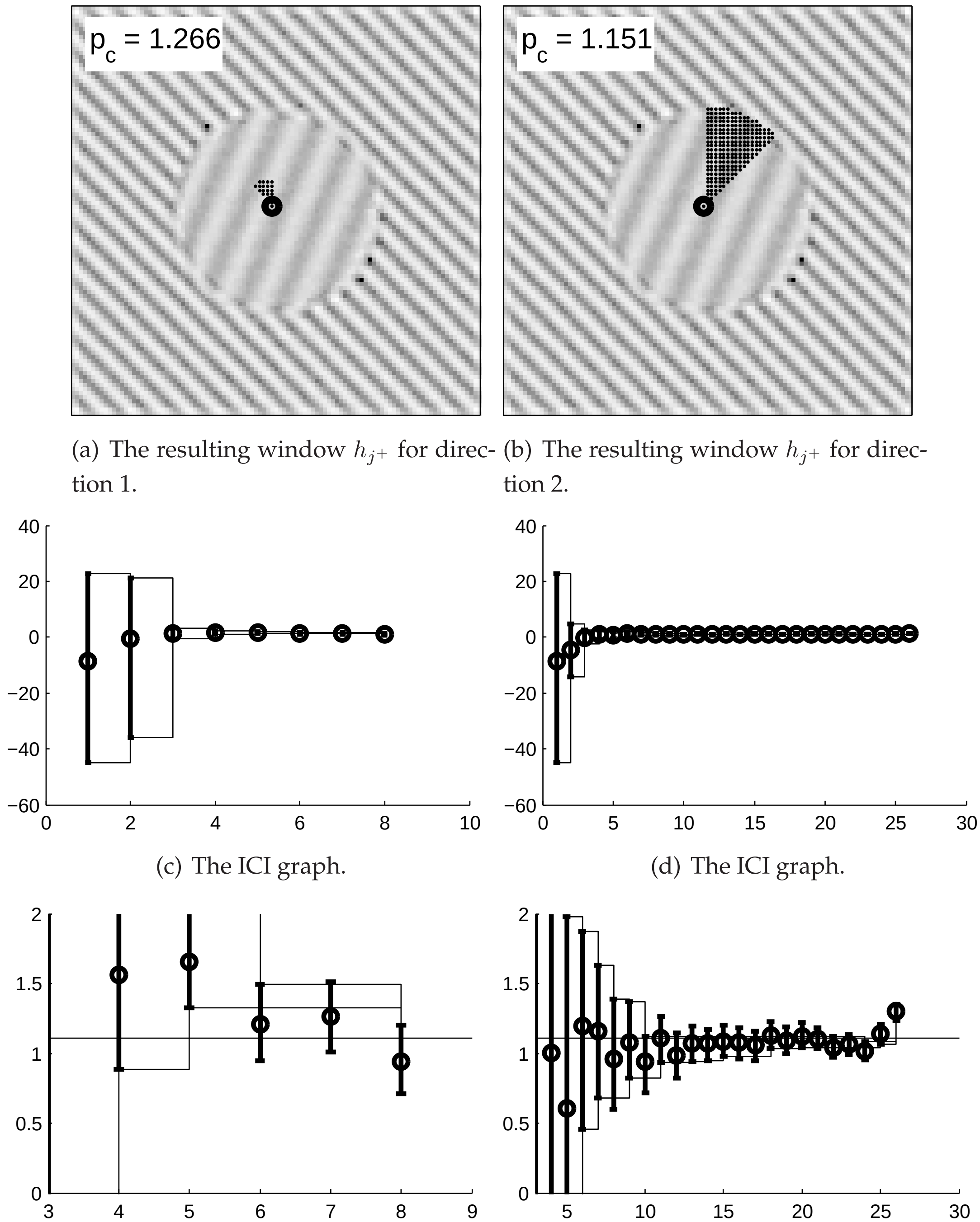


(e) The magnified ICI graph.



(f) The magnified ICI graph.

Figure 5.16: The ICI rule gives windows for calculating $\hat{p}_c$ for the PPaU structure. Input image contains Gaussian white noise with $\sigma_w = 10$. Adaptive regions obtained for $P = 0.95$ are shown with black dots. The origin pixel is shown with the black circle. True values of p are 1.169 and 1.945 for inner and outer sine wave respectively. Under the region image, there are corresponding ICI graphs. The horizonatal line in the ICI graphs represents the true value of p.



(a) The resulting window h_{j+} for direction 1.

(b) The resulting window h_{j+} for direction 2.

(c) The ICI graph.

(d) The ICI graph.

(e) The magnified ICI graph.

(f) The magnified ICI graph.

Figure 5.17: The ICI rule gives windows for calculating $\hat{p}_c$ for the PUPa structure. Input image contains Gaussian white noise with $\sigma_w = 10$. Adaptive regions obtained for $P = 0.95$ are shown with black dots. The origin pixel is shown with the black circle. True values of p are 1.111 and 1.742 for inner and outer sine wave respectively. Under the region image, there are corresponding ICI graphs. The horizonatal line in the ICI graphs represents the true value of p.

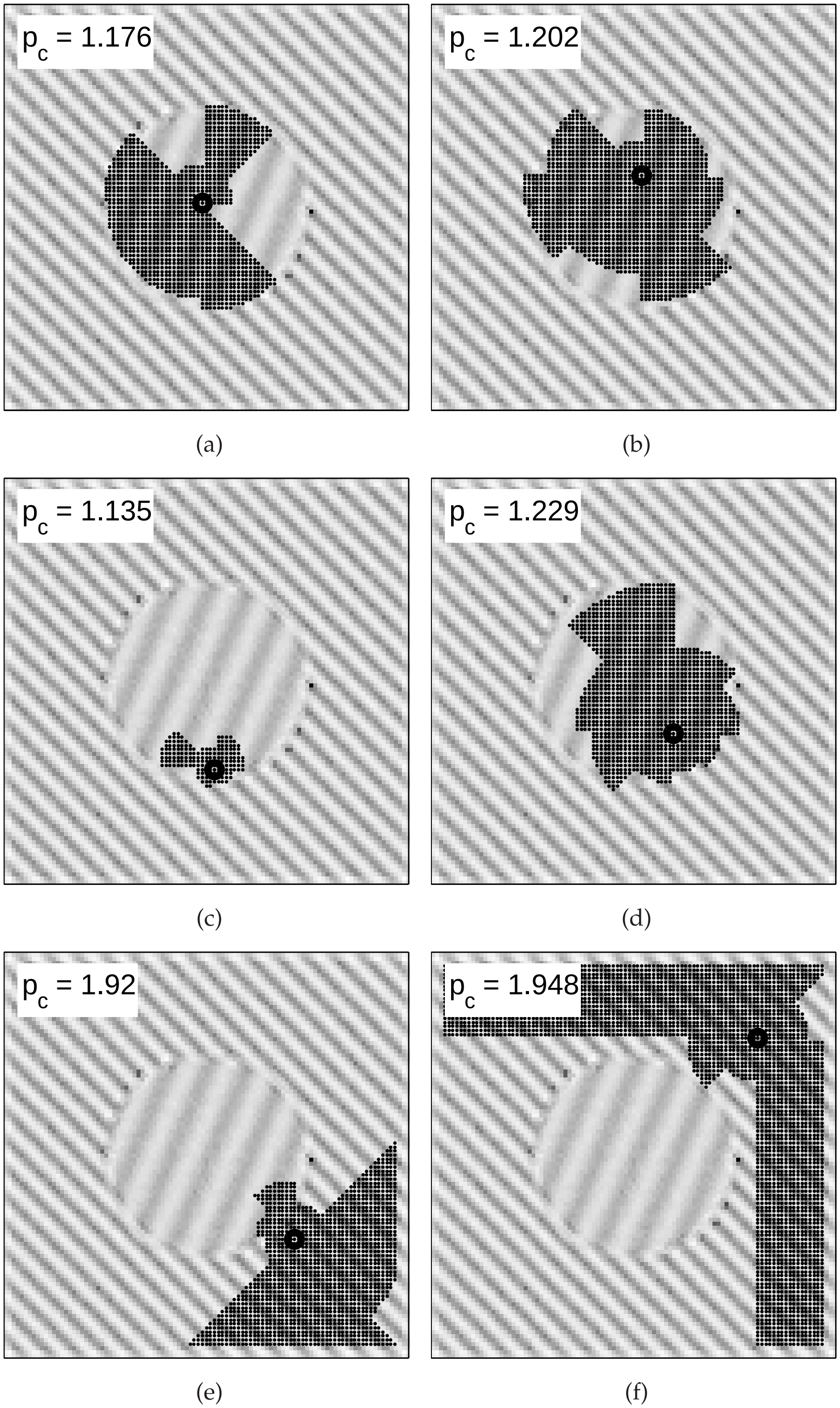


Figure 5.18: Regions obtained for different starting points (marked with bold circle) for calculating $\hat{p}_c$ in the PPaU structure for $\sigma_w = 10$. Adaptive regions obtained for CI with $P = 0.9$ are shown with small black dots. True values of p are 1.169 and 1.945 for inner and outer sine wave respectively.

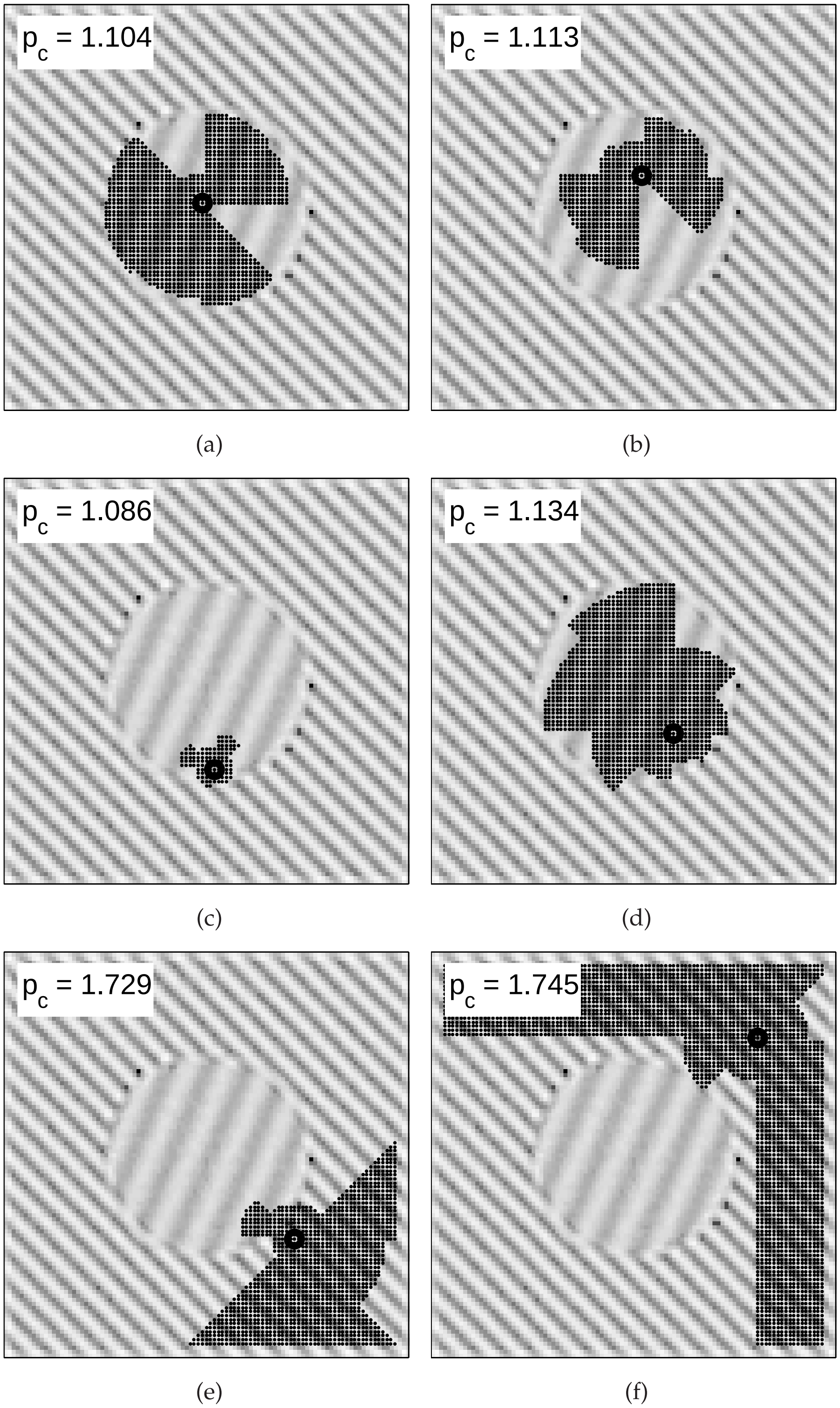
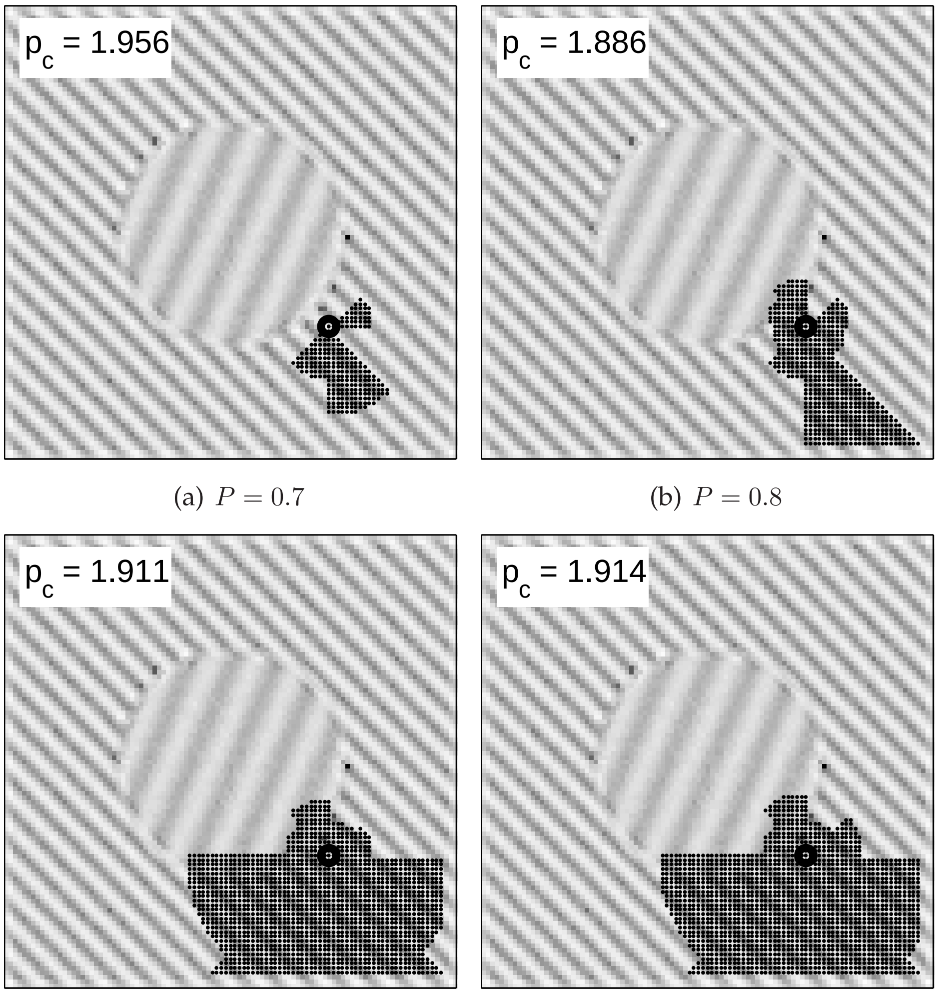
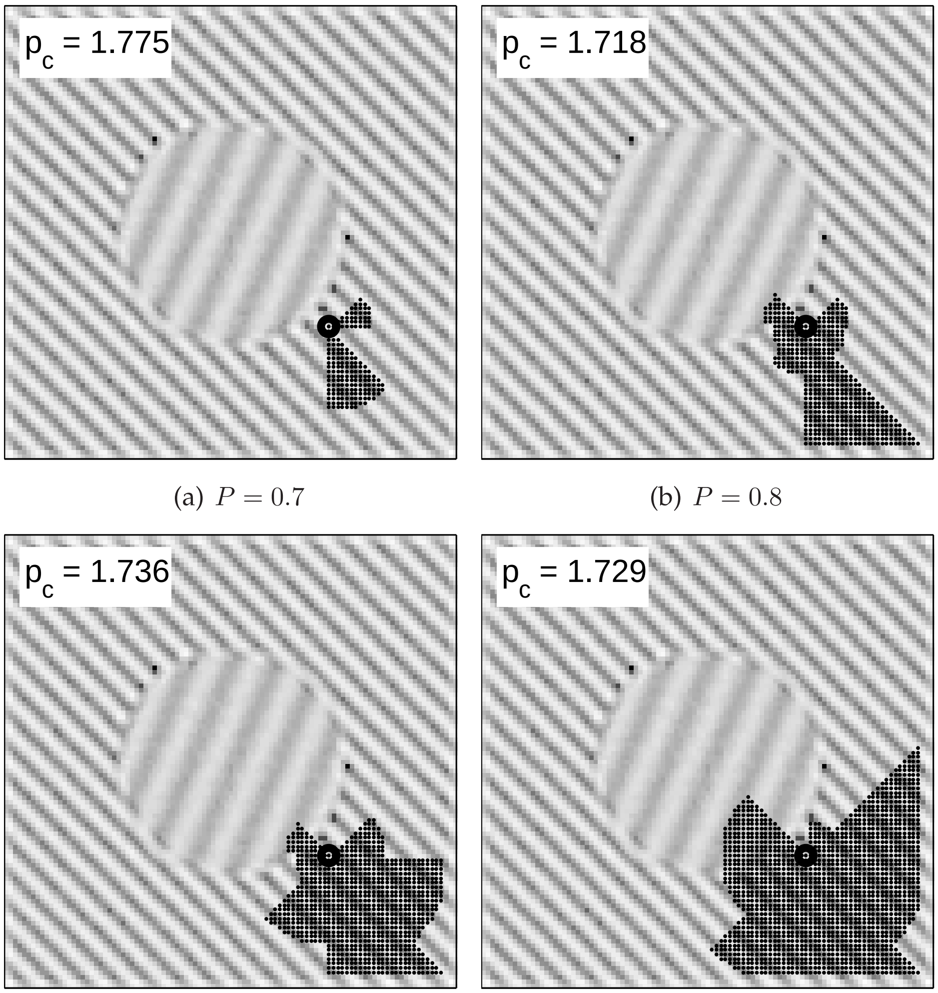


Figure 5.19: The PUPa structure, $\sigma_w = 10$. Adaptive regions obtained for $P = 0.9$ are shown with small black dots. True values of p are 1.111 and 1.742 for inner and outer sine wave respectively.



(a) $P = 0.7$ (b) $P = 0.8$ (c) $P = 0.9$ (d) $P = 0.95$

Figure 5.20: The PPaU structure. Adaptive regions for different CI widths.



(a) $P = 0.7$ (b) $P = 0.8$

(c) $P = 0.9$ (d) $P = 0.95$

Figure 5.21: The PUPa structure. Adaptive regions for different CI widths.

5.3.4 Implementation Issues

Recursive Computation

In this section we present methods to obtain computationally more efficient adaptation algorithms. The concepts are shown for the first decomposition level of the PPaU structure but the generalization to any level and also to the PUPa structure is straightforward.

In order to make the ICI calculation of $\hat{p}_c$ computationally more efficient the $\hat{p}_c$ for window h_j

$$\hat{p}_c^j = \frac{\left(\sum_{i \in h_j} x_{hi} \cdot x_{pi}\right) + \frac{N}{16}\sigma_w^2}{\left(\sum_{i \in h_j} x_{pi}^2\right) - \frac{6N}{256}\sigma_w^2} = \frac{\hat{q}_c^j}{\hat{r}_c^j} \tag{5.7}$$

is being calculated by reusing the results of the previous smaller window h_{j-1} that is included in the current window h_j.

$$\hat{p}_c^j = \frac{\hat{q}_c^{j-1} + \Delta\hat{q} + \frac{n}{16}\sigma_w^2}{\hat{r}_c^{j-1} + \Delta\hat{r} - \frac{6n}{256}\sigma_w^2} \tag{5.8}$$

where n is the number of pixels belonging to a difference of the two windows $h_j \backslash h_{j-1}$ and

$$\Delta\hat{q} = \sum_{i \in h_j \backslash h_{j-1}} x_{hi} \cdot x_{pi}, \tag{5.9}$$

$$\Delta\hat{r} = \sum_{i \in h_j \backslash h_{j-1}} x_{pi}^2. \tag{5.10}$$

Therefore, in the step j, the previously stored values of $\hat{q}_c^{j-1}$ and $\hat{r}_c^{j-1}$ will be reused, so only the calculations for the difference pixels will have to be performed. We have chosen regions to grow for one pixel in radial direction, so the difference pixels are those that belong to the next wider ring of one pixel width.

Acceptable Values of p

As shown in Chapter 3, as the value of the p parameter increases, the limit wavelet and scale functions become irregular. Based on experimental results we define an allowed range of p values to be

$$p_a \in [-5, 5] \tag{5.11}$$

For the values of p parameter inside the interval, the limit wavelet and scale functions are satisfactory regular for image processing applications.

Considering the estimation of the p parameter, the values of the estimate $\hat{p}_c$ can be strongly biased for its denominator from equation 4.36 being close to zero. Therefore, as stated in Section 4.4.3, when the values of denominator are found to be close to zero, the adaptation will not be performed. For such cases the value of p will be set to zero, leading to the basic lifting scheme structure. Usually, the value of denominator will be close to zero for the parts of the input image that contain low-pass frequency content. For those parts the adaptation will not be necessary since the fixed part of the filter bank will already cancel detail coefficients.

Chapter 6

Results

6.1 Introduction

In this chapter we present the results of the adaptation of the parameter p. In the first part of the chapter we analyze the obtained detail coefficients. As expected, the adaptation in the filter bank results in smaller detail coefficients.

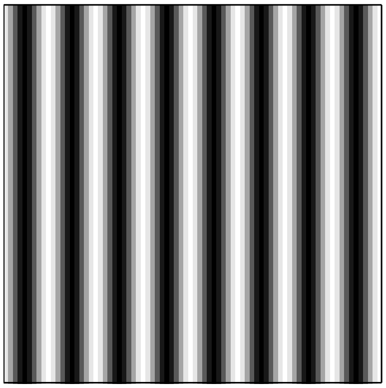
In the second part of the chapter, we present image denoising results. The denoising is based on the undecimated filter bank since it was shown [Donoho 95b] that for denoising purposes it outperforms the decimated filter bank.

6.2 Adaptation of the Parameter p

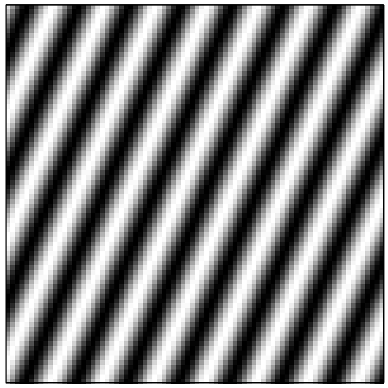
In the previous chapter the method for obtaining an appropriate adaptation region on which to calculate the value of the parameter p for a singe pixel was given. The method was based on the ICI rule. The adaptation region was obtained by merging 8 different wedge-like regions. In this way, the adaptation region can grow asymmetrically and cover a part of the image of a similar statistical property. It this chapter we compare the results based on such shape-adaptive regions with the results obtained with the adaptation based on the window of a fixed size.

6.2.1 Dependence of p on the Signal Frequency

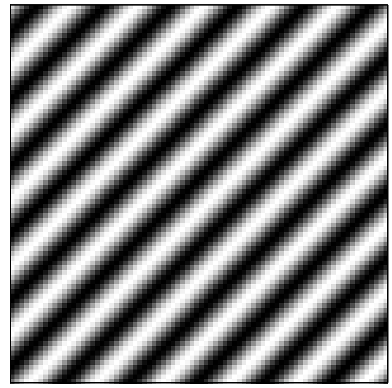
The adaptation of the parameter p on an image with a single harmonic component leads to a single value of the parameter p over the whole image. Spatial sine waves with different periods T and orientation angles θ (see Figure 6.1) result in different values of the parameter p. In the Fourier domain a single value of the parameter p leads to a magnitude frequency response of the corresponding analysis high-pass filter with zeros at different spatial frequencies (see Figure 3.13). Therefore, a single sine wave present in the analyzed image can be perfectly cancelled from wavelet detail coefficients (that are obtained as an output of the high-pass channel) with the correct choice of the parameter p.



(a) $\theta = 0$



(b) $\theta = \pi/8$



(c) $\theta = \pi/4$

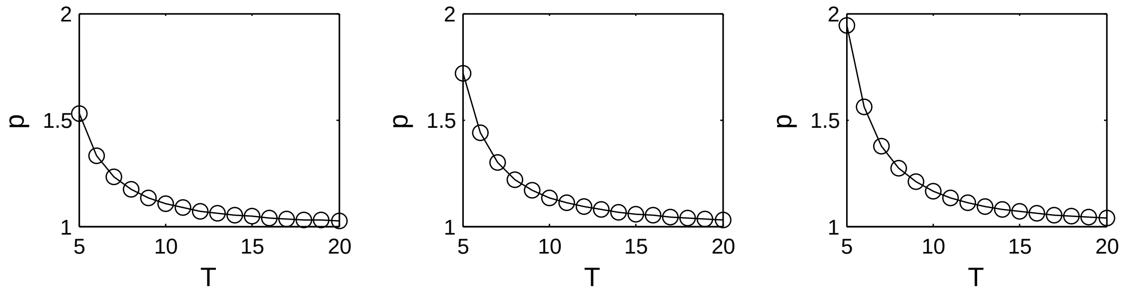
Figure 6.1: Sine waves with period $T = 10$ and different orientations.

Figure 6.2 shows different values of the parameter p corresponding to different periods of the sine wave pattern present in the image. As the signal frequency decreases (T increases) so does the value of the parameter p. Since the value of the parameter p decreases exponentially, the differentiation of smaller frequencies is better. Graphs from Figure 6.2 shows that differently oriented sine waves with the same period T will have different values of the corresponding parameter p. It is the consequence of the the zero ditch in magnitude frequency response of the analysis high-pass filters (see Figure 3.13) which tends to be diamond-shaped.

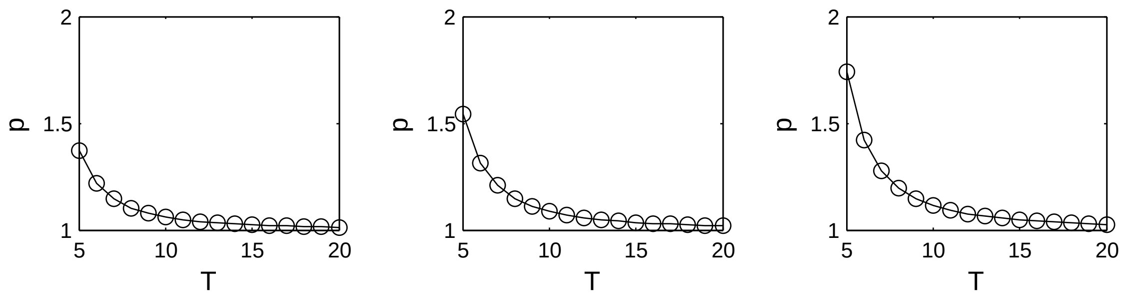
6.2.2 Adaptation for Synthetic Images

In this section we will use a test image called SineCirle composed of two sine waves similar to the one shown in Figure 1.2. The sine wave in the center of the image has period $T = 9$ and orientation $\theta = \pi/8$. For the first two decomposition levels, its corresponding filter bank parameters are $p_1 = 1.1292$ and $p_2 = 1.2261$. The outer sine wave has period $T = 5$ and $\theta = -\pi/4$. Its corresponding filter bank parameters are $p_1 = 1.0257$ and $p_2 = 1.0273$.

The test image has been corrupted with additive zero-mean Gaussian noise with $\sigma_w = 10$. Figures 6.3, 6.4 show the obtained values of the adapted p parameters for the case of the adaptation performed on a fixed size 3×3 and 5×5 windows respectively. As shown in the figures, the larger window suppresses noise better and gives



(a) PPaU structure. $\theta = 0$ (b) PPaU structure. $\theta = \pi/8$ (c) PPaU structure. $\theta = \pi/4$



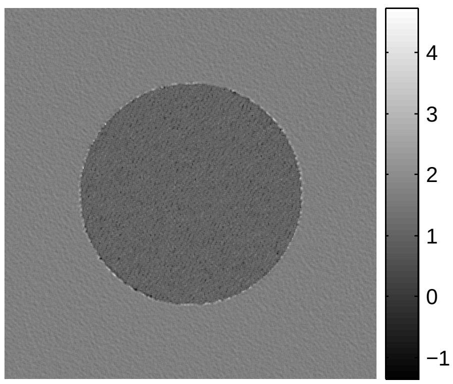
(d) PUPa structure. $\theta = 0$ (e) PUPa structure. $\theta = \pi/8$ (f) PUPa structure. $\theta = \pi/4$

Figure 6.2: p-T graphs. True values of the p parameter for adaptation in the first decomposition level of the PPaU and PUPa structures for sine waves with different periods T (x-axis) and different orientations θ (along columns).

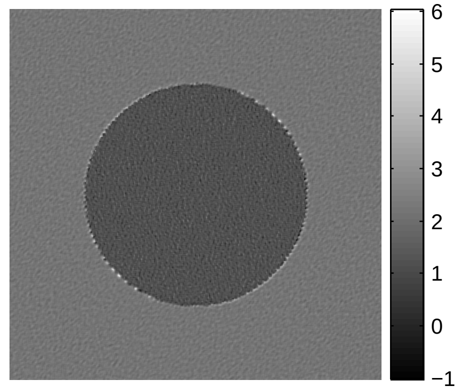
distributions of the $\hat{p}_c$ parameters more concentrated around the true value of the filter bank parameters. However, the larger window across the edges of the two sine-wave regions results in oversmoothing the p values on the boundary.

Figure 6.5 shows the results of the ICI adaptation. For every pixel of the image, the parameter $\hat{p}_c$ was calculated based on a shape-adaptive region. Histograms in Figures 6.5(c) and 6.5(d) show that the p values are much more concentrated around the true p values. It is a very important advantage of the ICI-based adaptation over the one based on the fixed size windows. Additionally, the adaptation algorithm did not allow a region to be smaller than 100 pixels. For such regions smaller than 100 pixels, $\hat{p}_c$ was set to zero, which equals a fixed filter bank with two zero moments. Black dots in Figures 6.5(a) and 6.5(a) represent those zeros.

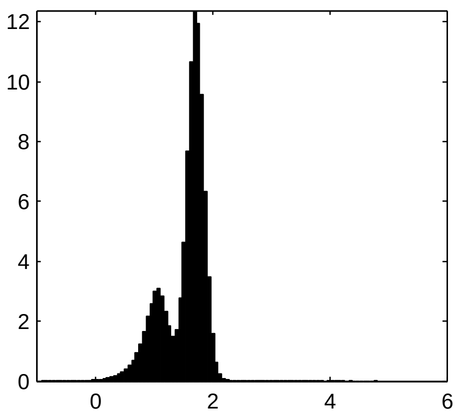
We have introduced a hybrid adaptation algorithm called ICImix that uses ICI adaptation in the first stage and in the second stage it replaces the zero values with ones obtained based on a fixed 9×9 window. Such a method was shown to give improved results in terms of image denoising while still retaining the sharp distributions of the p parameters that are inherent to the original ICI method.



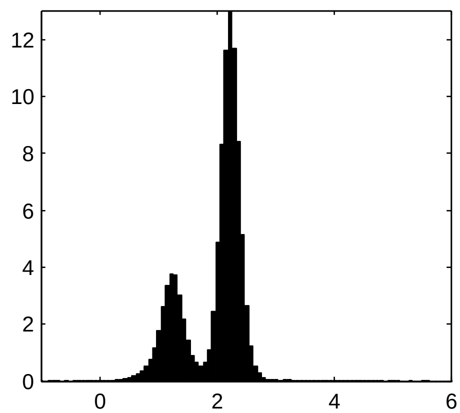
(a) $\hat{p}_c$ for level 1.



(b) $\hat{p}_c$ for level 2.

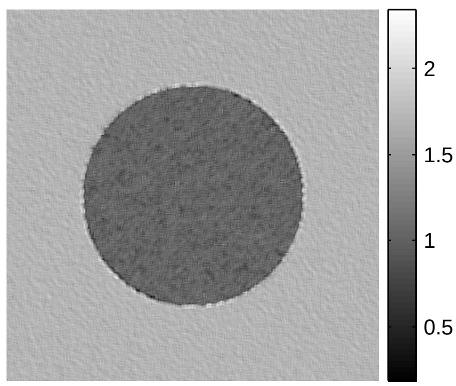


(c) Histogram of $\hat{p}_c$ parameters for level 1.

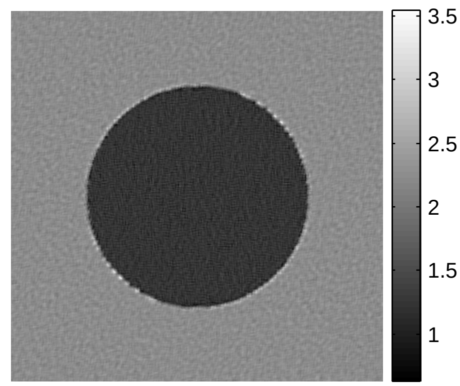


(d) Histogram of $\hat{p}_c$ parameters for level 2.

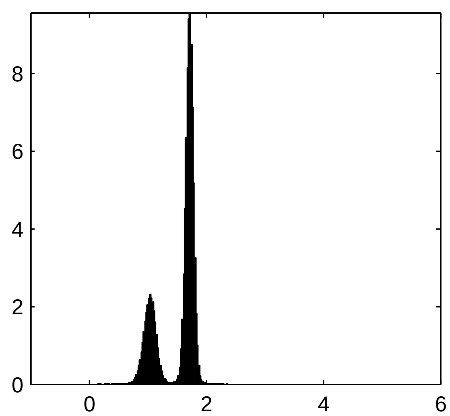
Figure 6.3: The undecimated PUPa structure with adaptation of p on a fixed 3×3 window size. The $\hat{p}_c$ parameters for levels 1 to 2. The image was corrupted with zero-mean Gaussian noise with $\sigma_w = 10$.



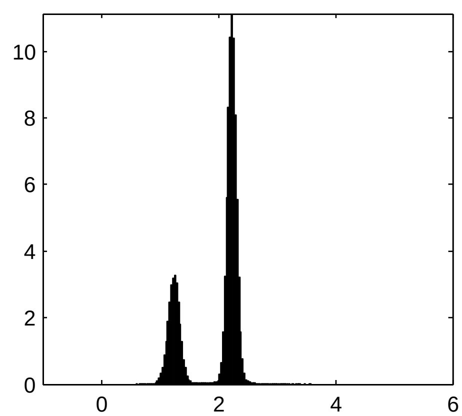
(a) $\hat{p}_c$ parameters for level 1.



(b) $\hat{p}_c$ parameters for level 2.

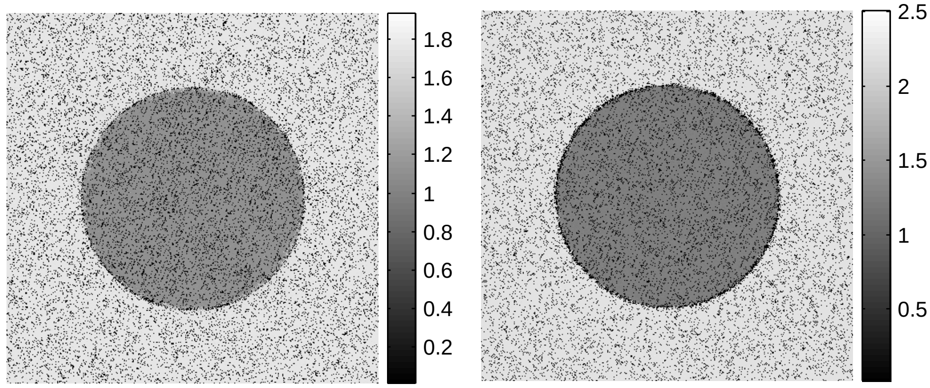


(c) Histogram of $\hat{p}_c$ for level 1.

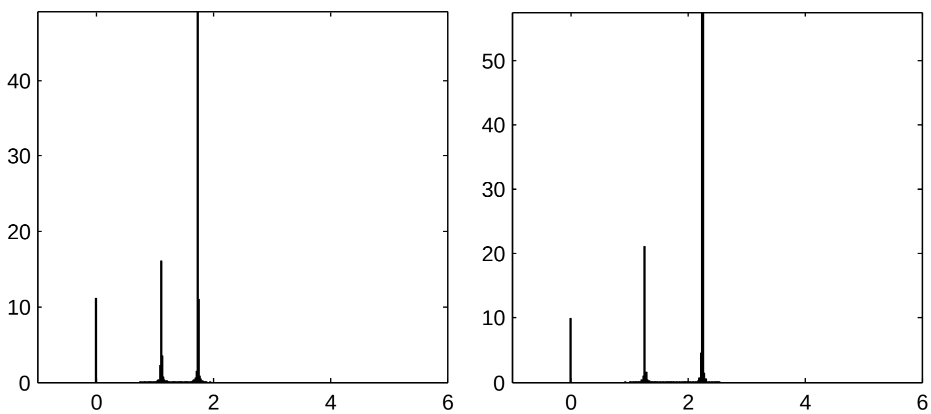


(d) Histogram of $\hat{p}_c$ for level 2.

Figure 6.4: The undecimated PUPa structure with adaptation of p on a fixed 5×5 window size. The $\hat{p}_c$ parameters for levels 1 to 2.

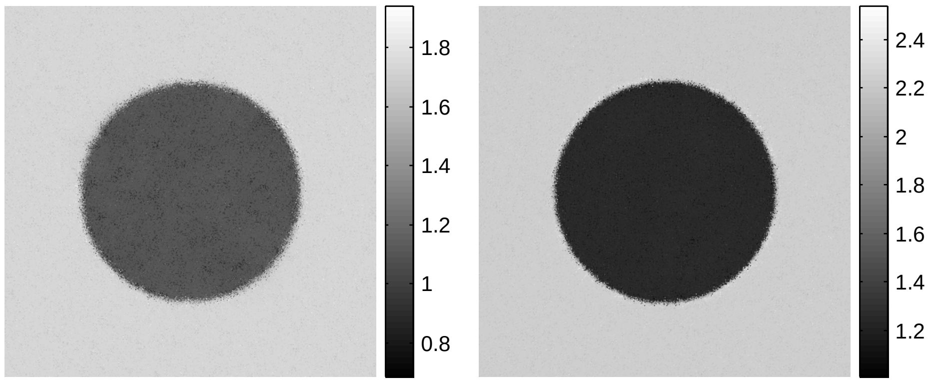


(a) $\hat{p}_c$ parameters for level 1. (b) $\hat{p}_c$ parameters for level 2.

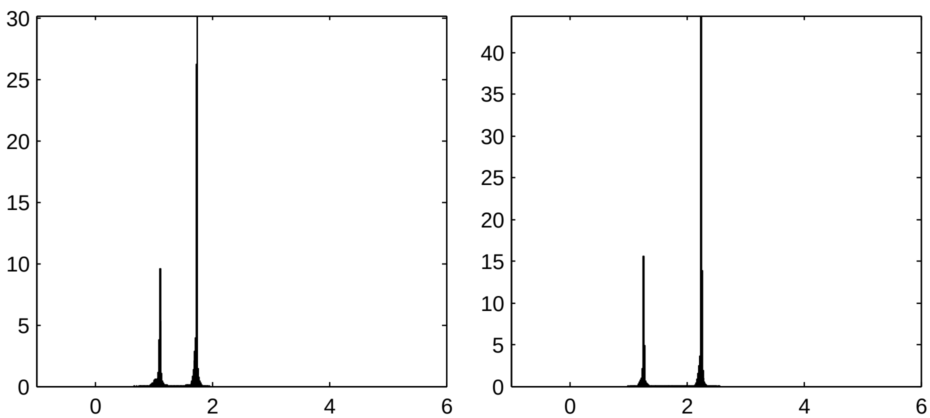


(c) Histogram of $\hat{p}_c$ for level 1. (d) Histogram of $\hat{p}_c$ for level 2.

Figure 6.5: The $\hat{p}_c$ parameters for levels 1 to 2 of the undecimated PUPa structure were obtained with ICI for $P = 0.9$. The image was corrupted with zero-mean Gaussian noise with $\sigma_w = 10$.



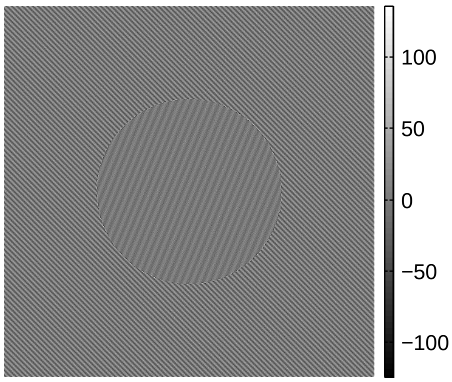
(a) $\hat{p}_c$ parameters for level 1. (b) $\hat{p}_c$ parameters for level 2.



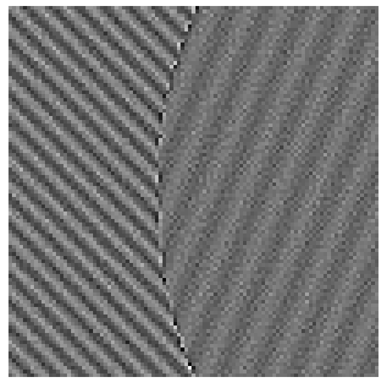
(c) Histogram of $\hat{p}_c$ for level 1. (d) Histogram of $\hat{p}_c$ for level 2.

Figure 6.6: The $\hat{p}_c$ parameters for levels 1 to 2 of the undecimated PUPa structure were obtained with ICI for $P = 0.9$. Additionally, the zero values have been replaced with ones obtained with a fixed 9×9 window. The image was corrupted with zero-mean Gaussian noise with $\sigma_w = 10$.

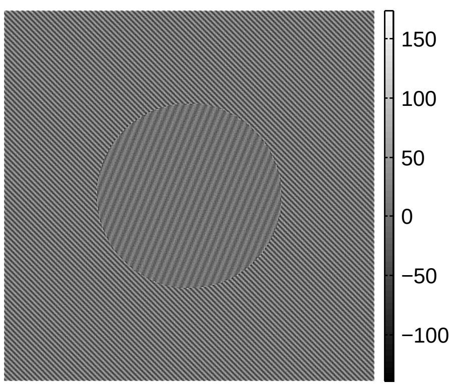
The resulting detail coefficients obtained with the adapted p parameters from the previous section are shown in Figures 6.8 and 6.9. When compared to the detail coefficients obtained with unadapted wavelet decomposition (Figure 6.7), it is obvious that the adaptation introduced significant advantages, i.e. the detail coefficients are set to zero in the whole image except at the circular boundary of the two sine-wave patterns.



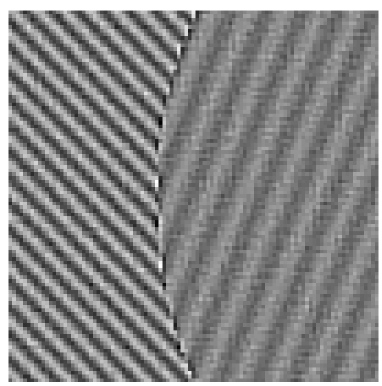
(a) D coefficients for level 1.



(b) Magnified D coefficients for level 1.

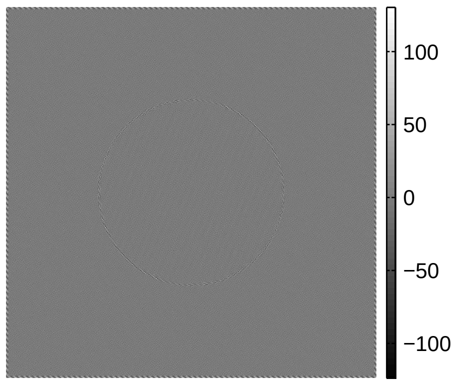


(c) D coefficients for level 2.

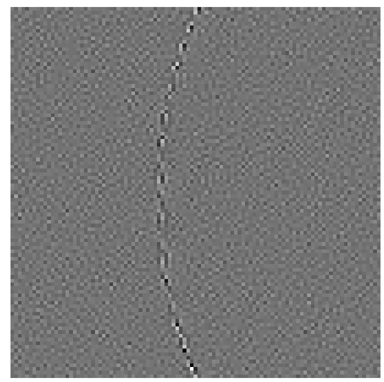


(d) Magnified D coefficients for level 2.

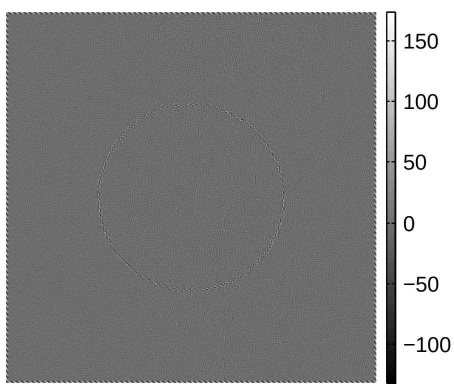
Figure 6.7: The undecimated P_2U_2 structure with no adaptation. The D coefficients for levels 1 to 2.



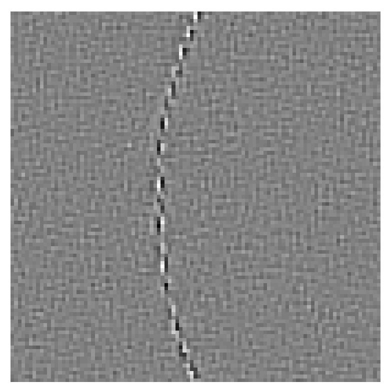
(a) D coefficients for level 1.



(b) Magnified D coefficients for level 1.

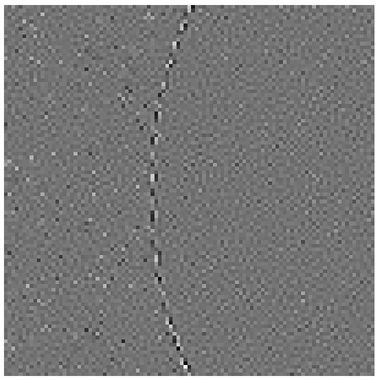


(c) D coefficients for level 2.

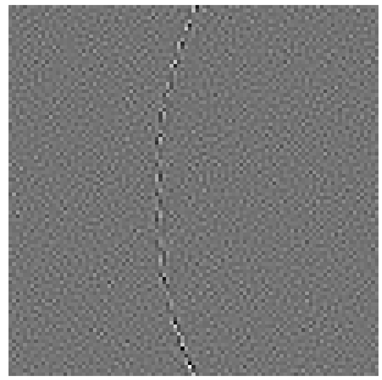


(d) Magnified D coefficients for level 2.

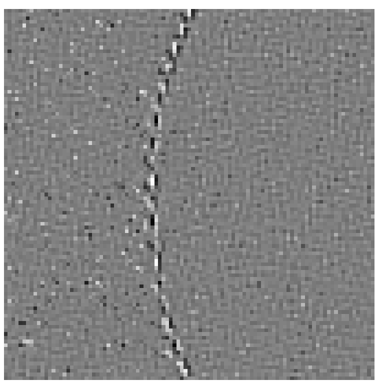
Figure 6.8: D coefficients for levels 1 to 2 of the undecimated PUPa structure obtained with adaptation on the fixed 5×5 window.



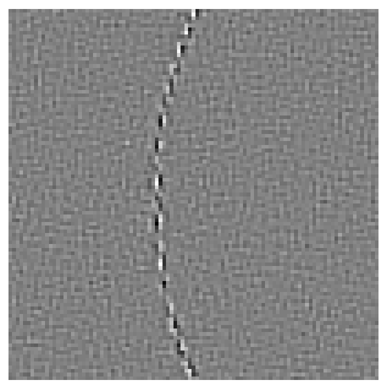
(a) Magnified detail coefficients for level 1 for ICI method.



(b) Magnified detail coefficients for level 1 for ICImix method.



(c) Magnified detail coefficients for level 2 for ICI method.

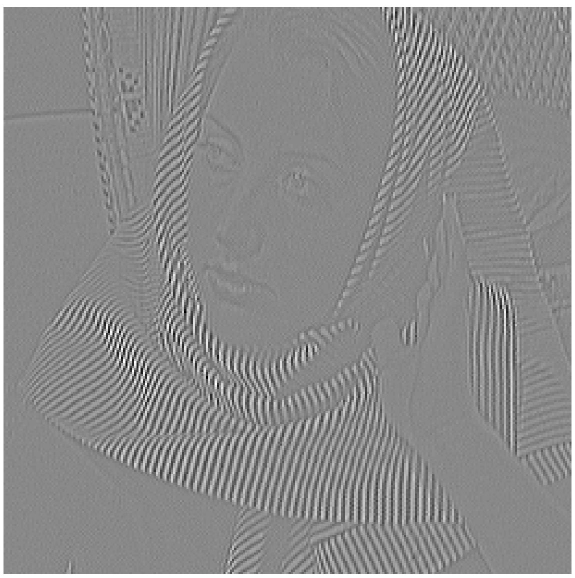


(d) Magnified detail coefficients for level 2 for ICImix method.

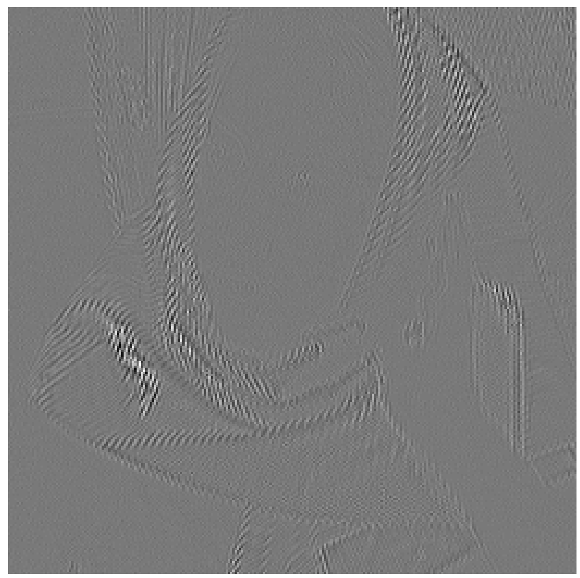
Figure 6.9: D coefficients for levels 1 to 2 of the undecimated PUPa structure obtained with ICI-based adaptation for $P = 0.9$.

6.2.3 Adaptation for Real-World Images

The Barbara image from Figure 6.11 contains lots of periodic components, mostly present on the Barbara's striped robe. As shown in Figures 6.11(b) and 6.11(d), the adaptation on those regions resulted in constant valued p parameters. Therefore, the information of the periodic components locally present in the image is captured in the filter bank parameters while the periodic components are being eliminated from the wavelet detail coefficients. Figure 6.10(a) shows detail coefficients for the part of the Barbara image obtained with fixed wavelet decomposition with two vanishing moments. Opposed to the fixed wavelet decomposition, the adaptation of the filter bank parameters resulted in cancelling most of the periodic components from the detail coefficients, as shown in Figure 6.10(b).



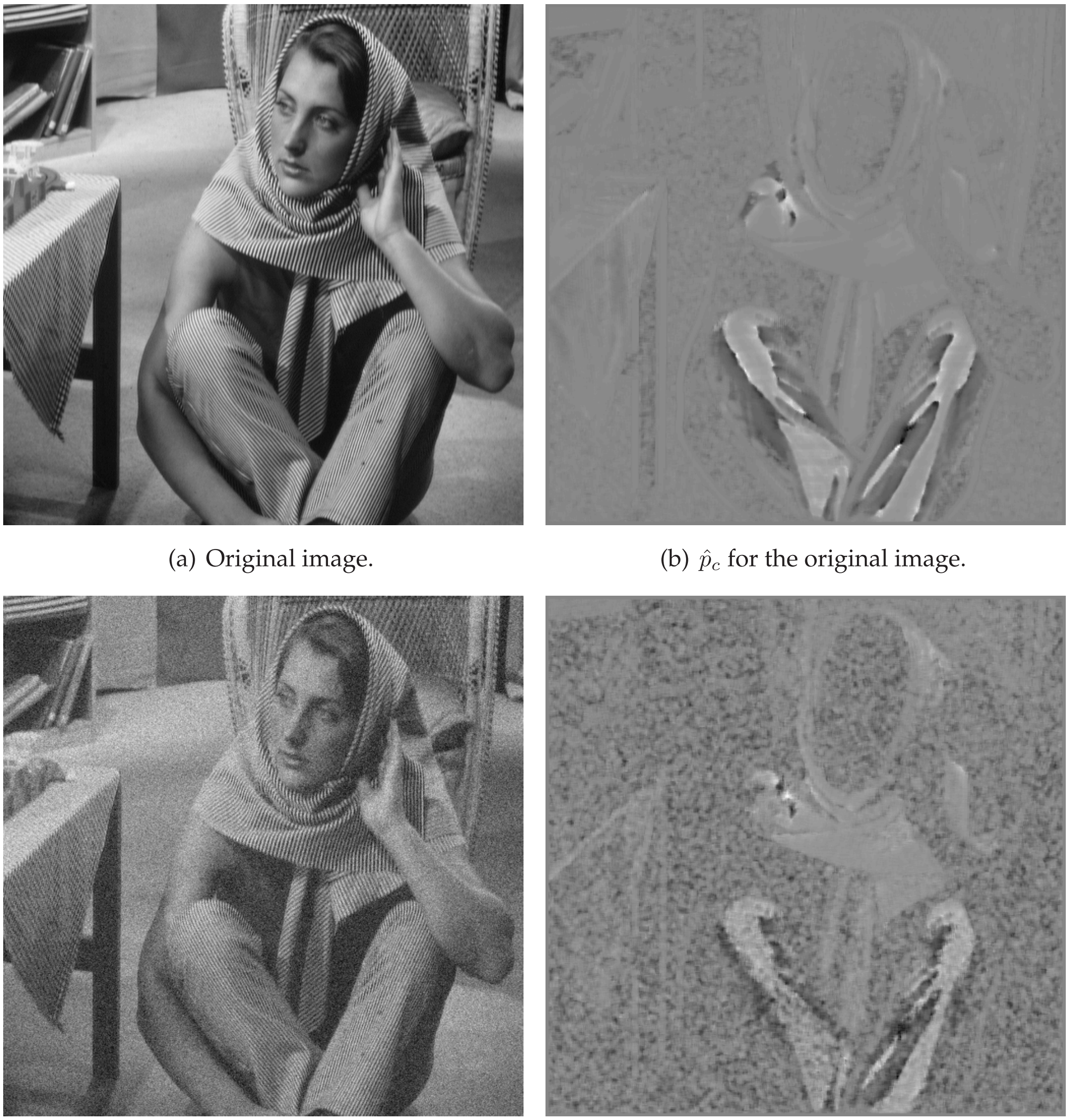
(a) Details coefficients obtained with nonadaptive P_2U_2 structure.



(b) Details coefficients obtained with adaptive PUPa structure.

Figure 6.10: Detail coefficients of the original Barbara image (the part around the head) for the first level of the fixed P_2U_2 structure compared with the ones obtained with the PUPa scheme with fixed-size 5×5 window adaptation.

Figure 6.12(a) shows a part of the Barbara image with a striped robe. The image was corrupted with zero-mean Gaussian noise with $\sigma_w = 10$. As shown in Figure 6.12(b), the adaptation on the window of the fixed 5×5 size was influenced by the presence of the noise and resulted in p parameters with rather spread histogram. Contrary to the fixed-size window, the ICI method gave p values that are more concentrated around a single value. Figure 6.12(c) shows the appropriate histogram of p values which reveals the presence of a single dominant periodic component. This example shows an impor-

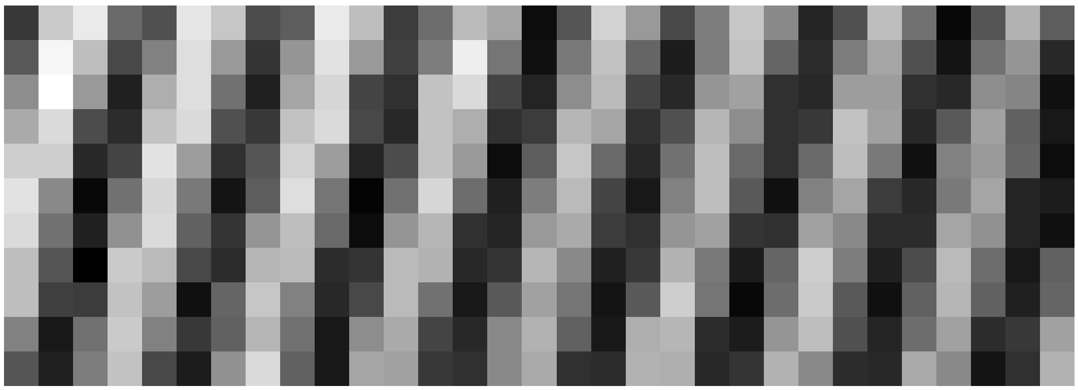


(a) Original image. (b) $\hat{p}_c$ for the original image.

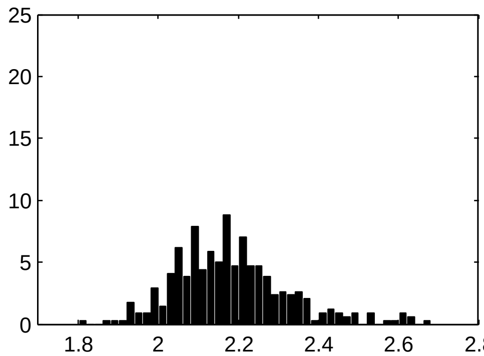
(c) Noisy image with $\sigma_w = 20$. (d) $\hat{p}_c$ for the noisy image.

Figure 6.11: The adapted $\hat{p}_c$ parameters for the first level of the PUPa scheme obtained with the fixed-size 5×5 window.

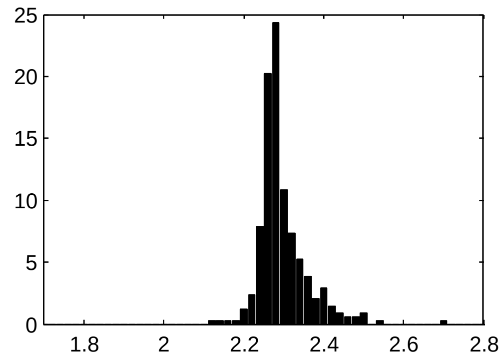
tant advantage of the ICI based adaptation method: the periodic components locally present in the image can be isolated more efficiently when compared to the adaptation on a fixed region.



(a) Part of the noisy Barbara image with $\sigma_w = 10$ for which the p_c histograms are shown.



(b) Histogram for $\hat{p}_c$ obtained with adaptation on the fixed 5×5 window.



(c) Histogram for $\hat{p}_c$ obtained with ICI-based adaptation with $P = 0.95$.

Figure 6.12: The histograms of $\hat{p}_c$ parameters for the part of the Barbara's scarf at level 1 of the undecimated PUPa structure.

6.3 Image Denoising

The wavelet thresholding was shown to be a very efficient method for image denoising [Donoho 95a]. The wavelet thresholding is a nonlinear operation performed on wavelet detail coefficients obtained in all decomposition levels. Wavelet coefficients lower than a given threshold are being set to zero while the coefficients higher than a given threshold either remain unchanged (hard thresholding) or are being reduced for the value of the threshold (soft thresholding). Soft thresholding usually leads to visually more pleasing results.

The wavelet thresholding relies on the fact that an image can be modelled as regions that are locally smooth and edges that are represented with strong variatons of the pixel values. Those locally smooth parts can be approximated well with the local polynomial representation obtaied with wavelet decomposition. Therefore, the detail coefficients for those smooth parts will be mostly turned to zero or very close to zero. On the other hand the edges will be strongly present in the detail coefficients. The noise present in the image representing an uncorrelated component of the analyzed signal will also be present in the detail coefficients. The choice of the threshold is obtained as a compromise for setting the detail coefficients that correspond to the noise to zero but at the same time not degrading the information about edges present in the detail coefficients.

More advanced approaches are used in order to obtain an appropriate value of the threshold or even to change the threshold locally based on the properties of the underlying image [Chang 00, Kivanc 99]. We will use soft and hard thresholding with the same value of the threshold accros the whole image.

In [Coifman 95] it was shown that significant improvement can be obtained by using the undecimated wavelet decomposition comapred to the decimated one. Therefore, we will present the results obtained by using the undecimated decomposition.

The image denosing results for adaptive wavelets will be compared with results obtainded with corresponding nonadaptive wavelets. The adaptation of the p parameters will be performed locally by either using a fixed window size or an adaptive window obtained by using the ICI rule.

6.3.1 Components of the Processed Image

Regarding the adaptation introduced in the filter bank and the image denoising application, we will consider four elements of an input image:

Polynomial components. The polynomial components are cancelled from detail coefficients. For a filter bank with $\tilde{N}$ vanishing moments the polynomials of degree lower than $\tilde{N}$ will be completely cancelled in the detail coefficients. This property is already guaranteed with the fixed part of the filter bank.

Harmonic components. The adaptation of the parameter p leads to cancelling some periodic components that are locally present in the analyzed image from detail coefficients. The information on these periodic components will now be transferred into the p parameters. Therefore, the thresholding of detail coefficients for purposes of image denoising will not degrade those periodic components present in the original image.

Noise. The noise will remain present in detail coefficients in order to be cancelled by means of wavelet thresholding. In the ideal case, although the adaptation is being performed on the noisy image, the noise component should not be present in the calculated p parameters.

Edges. The adaptation algorithm will mostly fail to detect edges. Therefore, the edges will still remain in the detail coefficients and can be degraded by thresholding operations.

6.3.2 Denoising With Adaptive Quincunx Wavelets

In this section, we present denoising results for various test images. The results are presented in terms of the Peak Signal to Noise Ratio (PSNR) values. PSNR in decibels (dB) is computed as

$$PSNR = 20 \log_{10} \frac{255}{RMSE} \tag{6.1}$$

where 255 is the largest 8-bit pixel value, 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

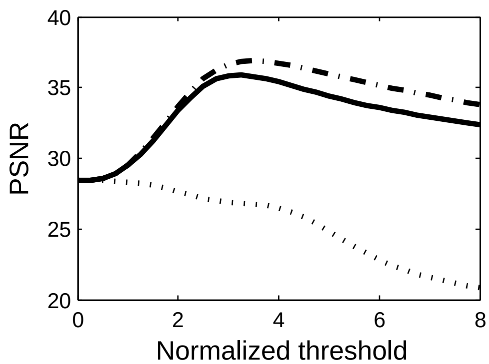
$$RMSE = \sqrt{MSE} = \frac{\sqrt{\sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} (x[n_1, n_2] - \hat{x}[n_1, n_2])^2}}{N} \tag{6.2}$$

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.

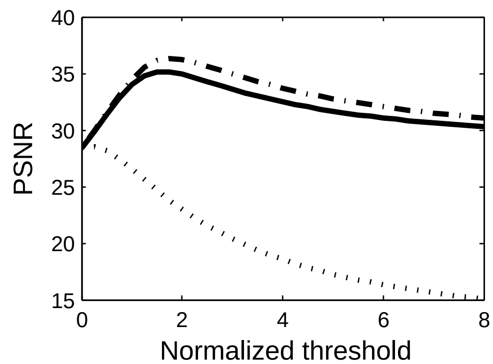
Synthetic Images

Denoising results in terms of PSNR for the SineCircle image are shown in Figure 6.13. Denoising was performed for a range of threshold values which are proportional to the standard deviation of the noise present in the image. The two graphs compare the three wavelet denoising types: denoising with fixed quincunx wavelets (dotted line), denoising with adaptive PUPa structure on a 5×5 region (solid line), and denoising with the ICImix adaptation method for the PUPa structure (dash-dot line). As shown, for $\sigma_w = 10$, the improvement obtained with the fixed-region adaptation over non-adaptive wavelets is 7.5 dB for hard thresholding and 6.5 dB for soft thresholding. The ICImix adaptation introduces additional 1dB improvement over the fixed region adaptation. The part of the noisy SineCircle image and the images obtained with the three denoising methods are shown in Figure 6.14. For each method, a soft threshold value corresponding to its highest PSNR value from Figure 6.13(b) was chosen.

It is important to note that the ICI-based method, although giving p parameters very sharp distributions did not give a significant improvement over the adaptation of a fixed region which in presence of noise gives less accurate values of p parameters. The reason is that the p parameters obtained with the fixed-window method were still precise enough to cancel the sine-wave components from the detail coefficients, as shown in Figure 6.8.



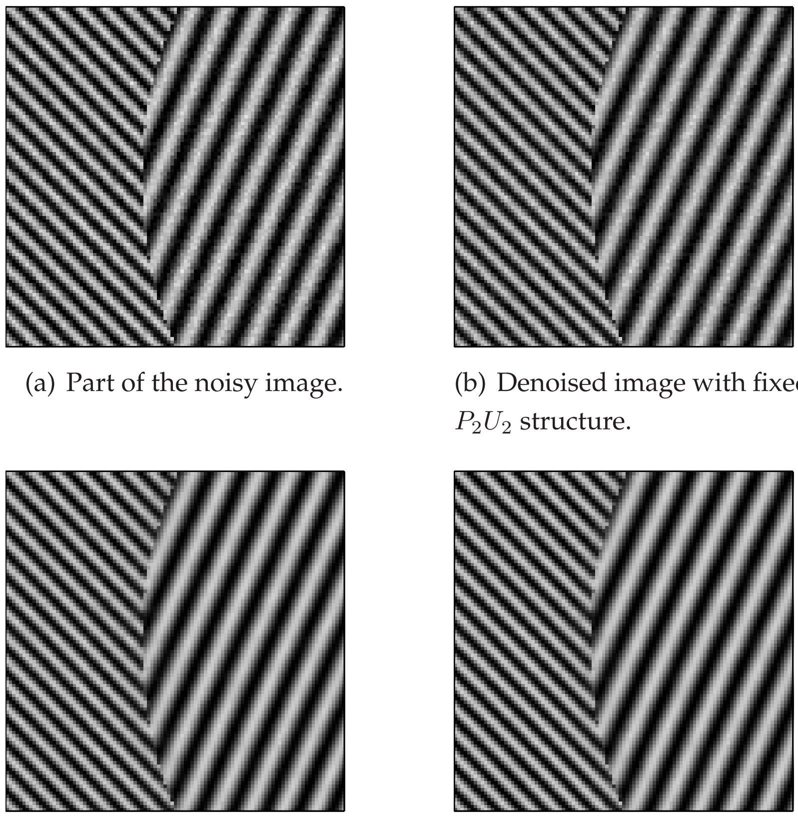
(a) $\sigma_w = 10$, hard threshold.



(b) $\sigma_w = 10$, soft threshold.

Figure 6.13: Denoising results for SineCircle image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure. Dash-dot line represents results obtained with the ICImix method.

Results from Figure 6.15 show that the adaptation introduced in the filter bank significantly improves denoising results. The adaptation transferred the information about the harmonic components in the filter parameters and the noise present in the image was more efficiently suppressed.



(a) Part of the noisy image.

(b) Denoised image with fixed P_2U_2 structure.

(c) Denoised image with adaptation on a 5×5 region.

(d) Denoised image with ICImix adaptation.

Figure 6.14: Denoising results for soft thresholding shown on the part of the SineCircle image.

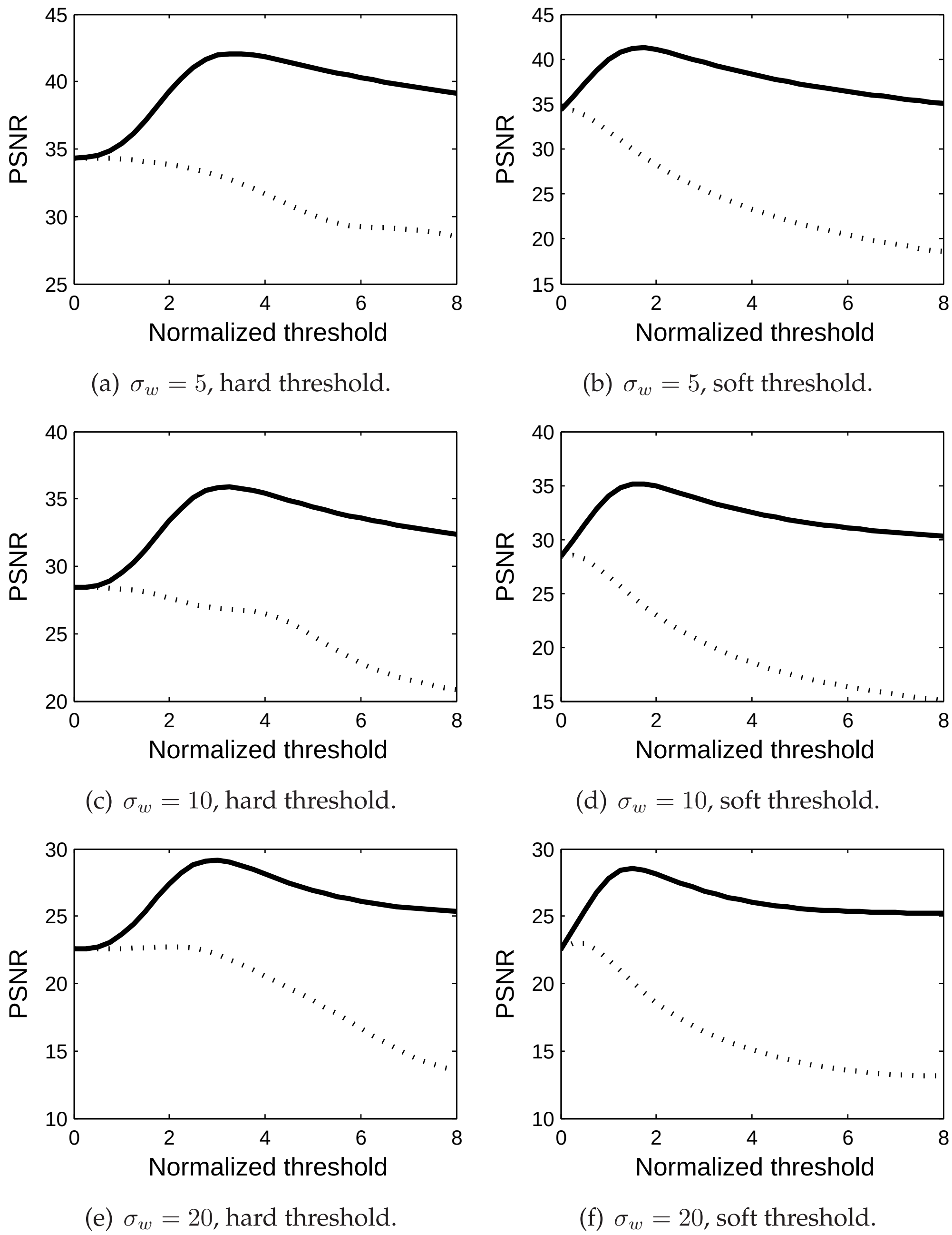
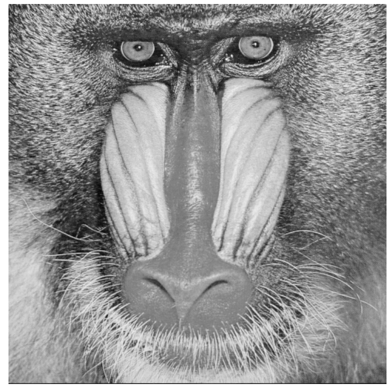


Figure 6.15: Denoising results for SineCircle image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure.

Real-world Images

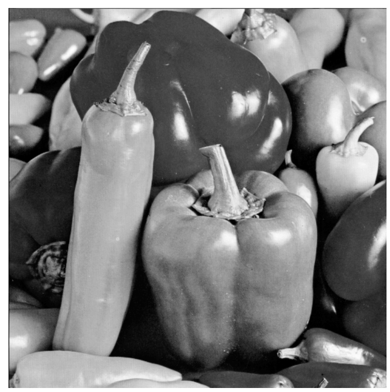
In order to test denoising performance of the adaptive filter bank on real-world images, the four typical test images from Figure 6.16 are used.

(a) Barbara



(b) Mandrill

(c) Goldhill

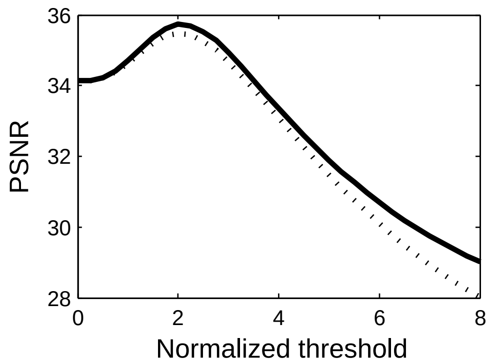


(d) Peppers

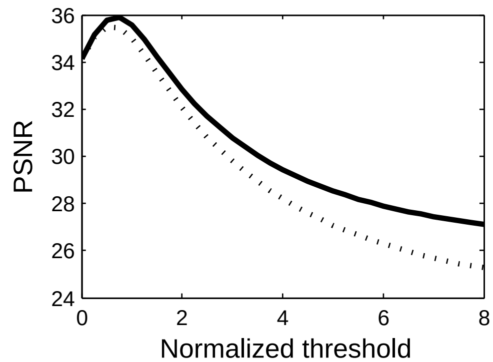
Figure 6.16: The 8-bit real-world test images of size 512×512.

As shown in Figure 6.17, the improvement obtained with the adaptation of the filter bank compared to the denoising with fixed wavelets is rather small and around 1 dB. Visually, there is not a big difference between Figures 6.18(c) (fixed wavelets) and 6.18(d) (adaptive wavelets).

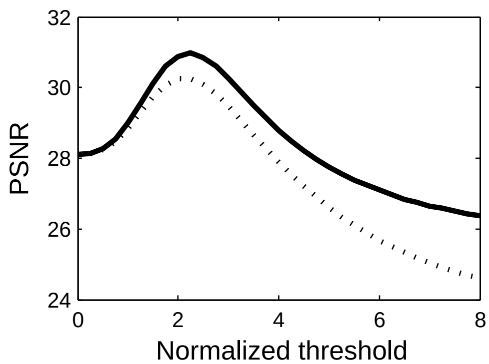
Denoising results are shown in Figures 6.19, 6.20, 6.21, 6.22, 6.23, 6.24 for images Madnrill, Goldhill, and Peppers. The improvement of using adaptive wavelets over nonadaptive wavelets is rather small. It clearly states that the model of locally present harmonic components can only be partially applied to real-world images. While being very suitable for representing images with dominant periodic content, the proposed denoising method is not applicable to real-world images in general.



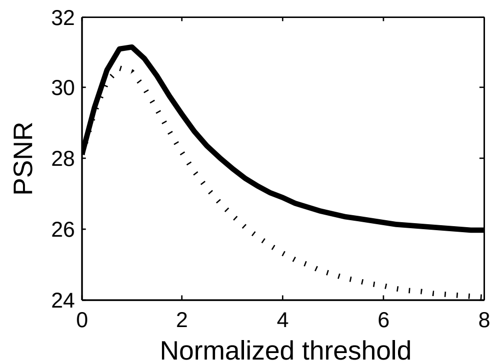
(a) $\sigma_w = 5$, hard threshold.



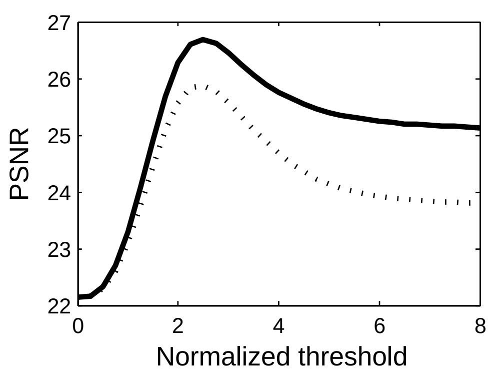
(b) $\sigma_w = 5$, soft threshold.



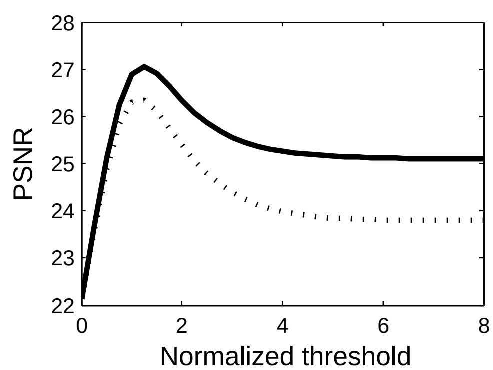
(c) $\sigma_w = 10$, hard threshold.



(d) $\sigma_w = 10$, soft threshold.



(e) $\sigma_w = 20$, hard threshold.



(f) $\sigma_w = 20$, soft threshold.

Figure 6.17: Denoising results for Barbara image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure.

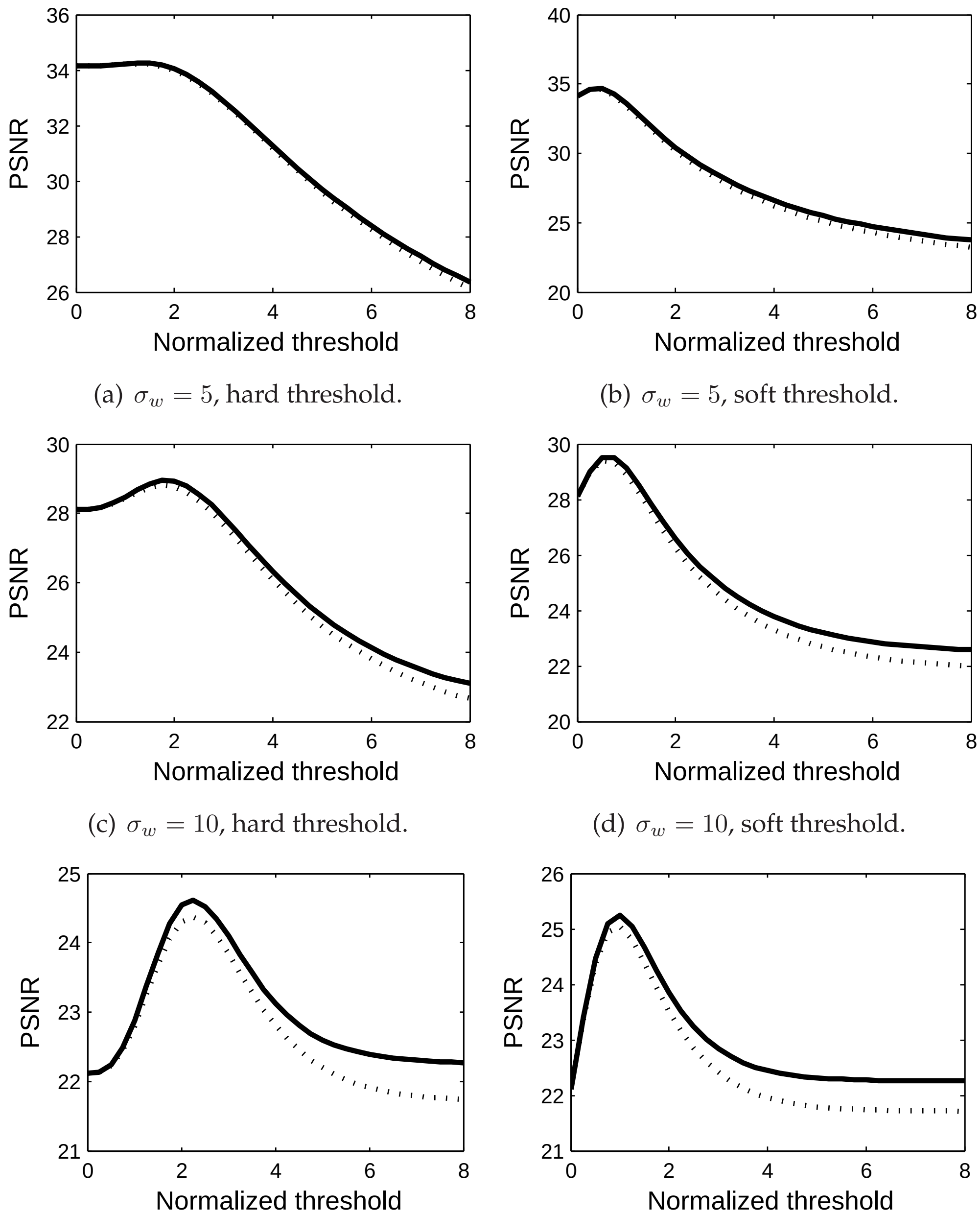
(a) Part of the original image.

(b) Part of the noisy image.

(c) Denoised image with fixed P_2U_2 structure.

(d) Denoised image with adaptation on a 5×5 region.

Figure 6.18: Denoising results for soft thresholding shown on the part of the Barbara image.



(a) $\sigma_w = 5$, hard threshold.

(b) $\sigma_w = 5$, soft threshold.

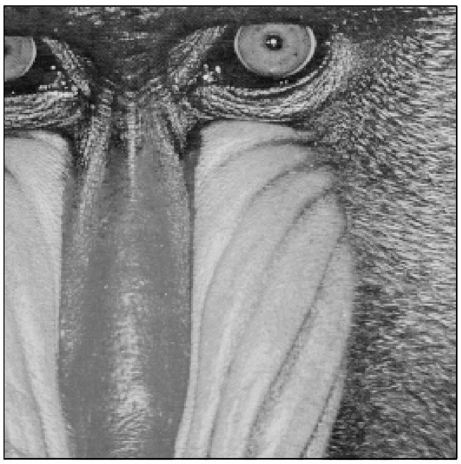
(c) $\sigma_w = 10$, hard threshold.

(d) $\sigma_w = 10$, soft threshold.

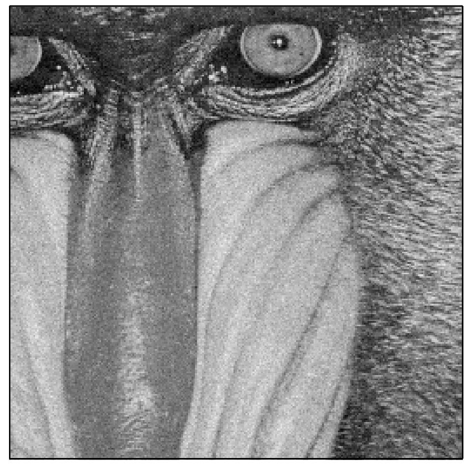
(e) $\sigma_w = 20$, hard threshold.

(f) $\sigma_w = 20$, soft threshold.

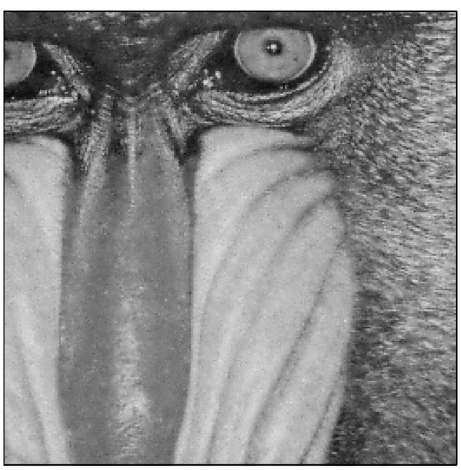
Figure 6.19: Denoising results for Mandrill image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure.



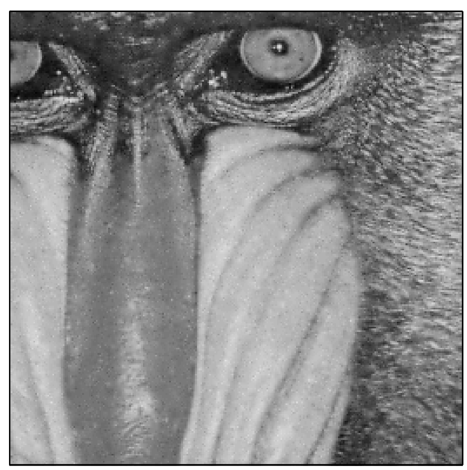
(a) Part of the original image.



(b) Part of the noisy image.

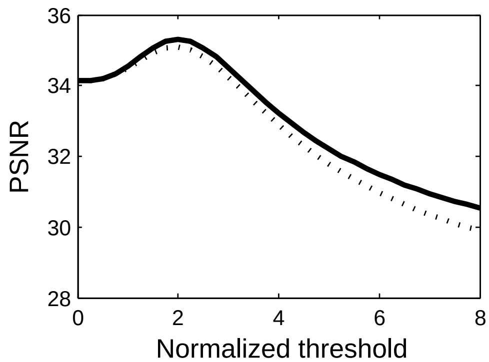


(c) Denoised image with fixed P_2U_2 structure.

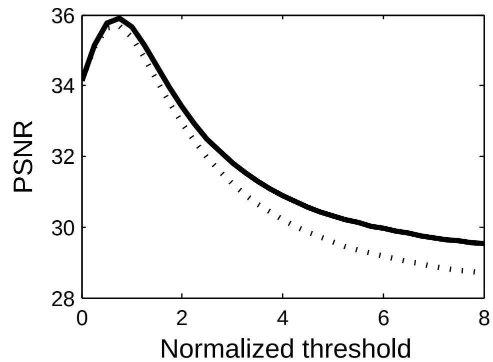


(d) Denoised image with adaptation on a 5×5 region.

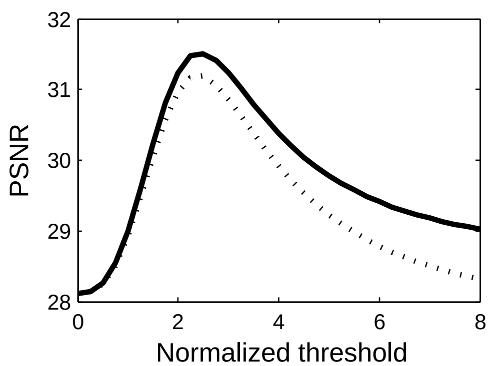
Figure 6.20: Denoising results for soft thresholding shown on the part of the Mandrill image.



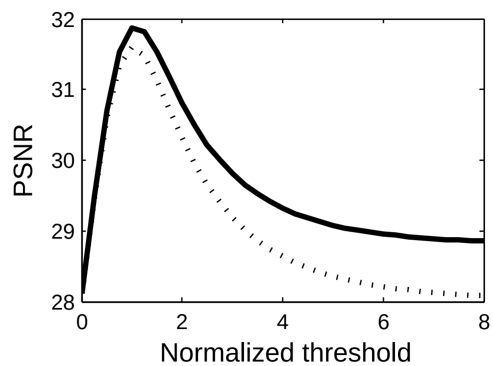
(a) $\sigma_w = 5$, hard threshold.



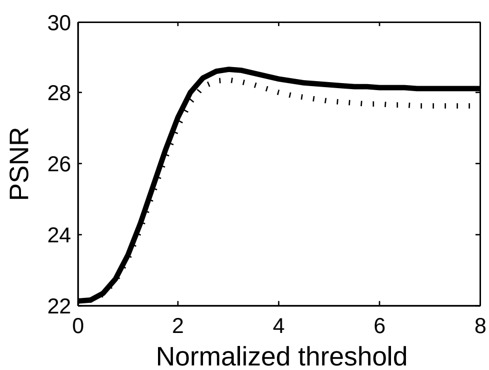
(b) $\sigma_w = 5$, soft threshold.



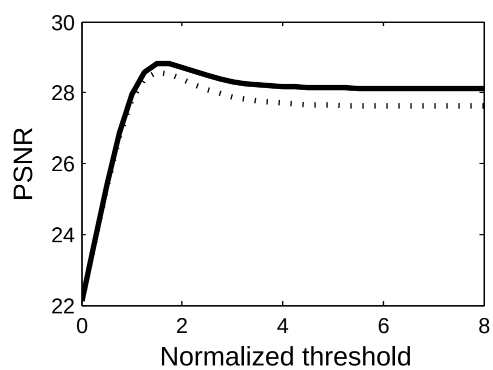
(c) $\sigma_w = 10$, hard threshold.



(d) $\sigma_w = 10$, soft threshold.



(e) $\sigma_w = 20$, hard threshold.



(f) $\sigma_w = 20$, soft threshold.

Figure 6.21: Denoising results for Goldhill image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure.

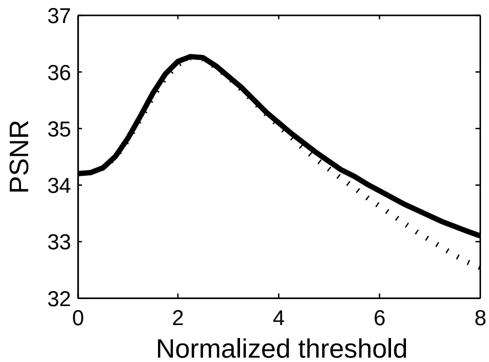
(a) Part of the original image.

(b) Part of the noisy image.

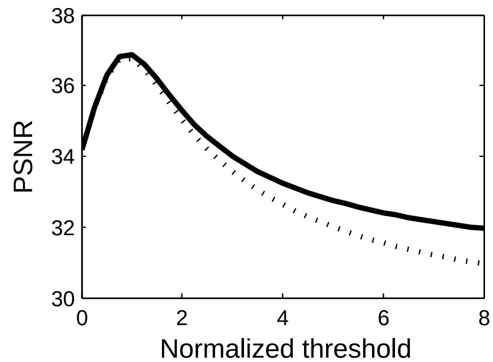
(c) Denoised image with fixed P_2U_2 structure.

(d) Denoised image with adaptation on a 5×5 region.

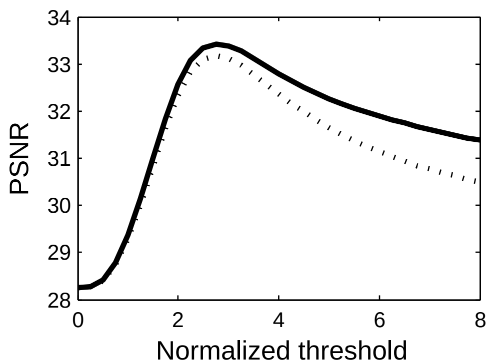
Figure 6.22: Denoising results for soft thresholding shown on the part of the Goldhill image.



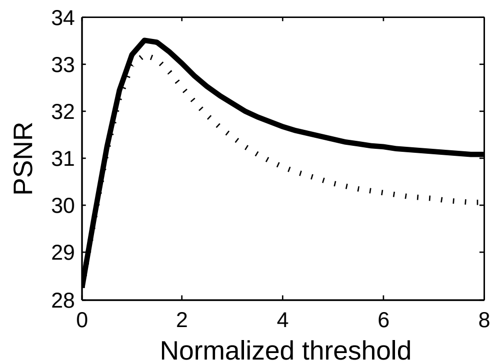
(a) $\sigma_w = 5$, hard threshold.



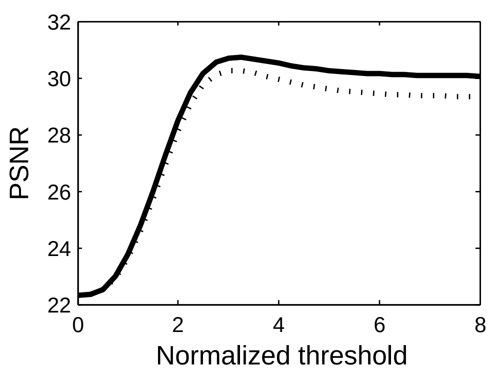
(b) $\sigma_w = 5$, soft threshold.



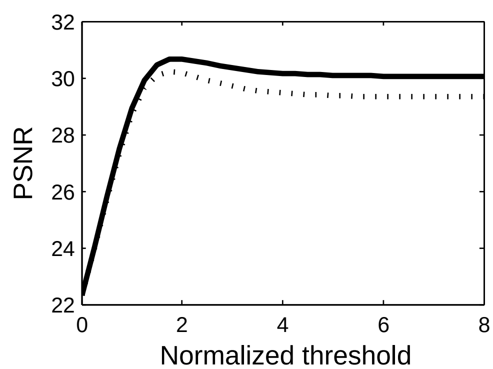
(c) $\sigma_w = 10$, hard threshold.



(d) $\sigma_w = 10$, soft threshold.



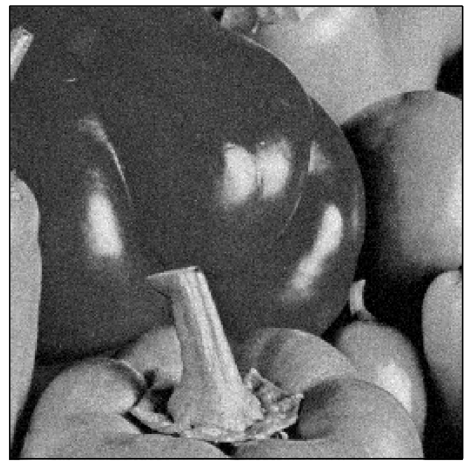
(e) $\sigma_w = 20$, hard threshold.



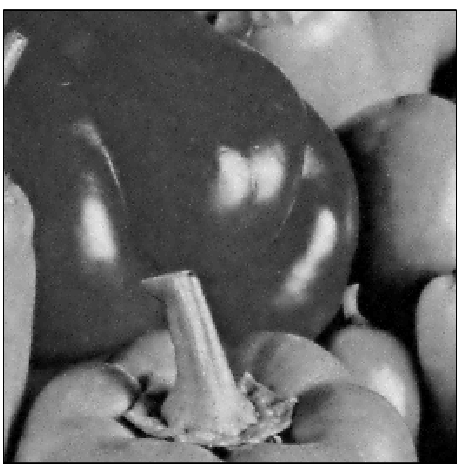
(f) $\sigma_w = 20$, soft threshold.

Figure 6.23: Denoising results for Peppers image. Solid line represents PSNR values for the PUPa structure based on the 5×5 adaptation window. Dashed line represents results obtained with the fixed P_2U_2 structure.

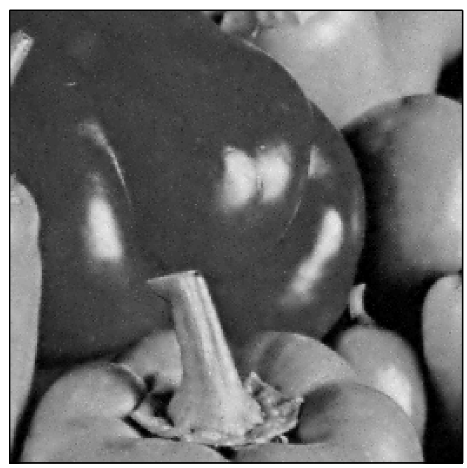
(a) Part of the original image.



(b) Part of the noisy image.



(c) Denoised image with fixed P_2U_2 structure.



(d) Denoised image with adaptation on a 5×5 region.

Figure 6.24: Denoising results for soft thresholding shown on the part of the Peppers image.

Chapter 7

Conclusions and Future Directions

7.1 Main Conclusions

In this thesis we presented a novel construction of adaptive wavelet decompositions based on adaptive wavelet filter banks. The purpose of the adaptation was to obtain a more compact representation of an analyzed image when compared to the nonadaptive wavelet transform.

We used the biorthogonal wavelet filter bank based on the lifting scheme, as proposed by Kovacevic and Sweldens in [Kovacevic 00]. The filter bank is nonseparable, and it is based on quincunx sampling, which gives a more isotropic transform than the one obtained using the separable sampling.

Based on the selected filter bank, we created a wavelet decomposition that adapts its properties according to the local properties of the analyzed image. The main design requirement was to make the wavelet decomposition locally adaptive while at the same time retaining the polynomial reconstruction property of the transform, i.e. keeping the vanishing moments. The adaptation was introduced by modifying the predict stage in the lifting scheme, and the two schemes named PPaU and PUPa were proposed (Chapter 3). Both schemes retain a desired number of dual and primal vanishing moments which are ensured with the fixed part of the filter bank. As shown in Section 4.6, the advantage of the PUPa structure is that the adaptation in one level does not influence following levels. On the contrary, the PPaU structure suffers from the influence of the adaptation in the lower decomposition levels to higher decomposition levels. A drawback of the PUPa structure is much wider supports of the analysis high-pass filters (see Section 4.5) in comparison to the PPaU structure which gives poorer results near the edges present in the analyzed image.

The adaptation of the filter bank parameters was performed in order to minimize the squared prediction error on a neighborhood of a given pixel, which led to the minimization of the energy of the wavelet detail coefficients. The influence of the additive zero-mean Gaussian noise to the estimation of the adaptive filter parameter p was also considered. In Section 4.3.3, it was shown that the obtained estimate of p is biased and that its bias depends on the variance of the input noise, the values of the original image, as well as the size (and shape) of the region on which the estimate is being calculated. Since there is no closed form expression for the expectation of the estimated p, the unbiased estimator is not available.

In order to decrease the influence of the noise, in Section 4.4 we proposed the improved estimator $\hat{p}_c$. Although this estimator is still not perfectly unbiased, the simulation results showed that the bias is much smaller if the probability distribution of the denominator in the expression for $\hat{p}_c$ (equation 4.23) is negligible around zero. It is important to note that the improvement terms introduced in the expression for estimator $\hat{p}_c$ depend only on the number of pixels included in the adaptation region and the variance of the input zero-mean noise (see equations 4.36, 4.45, 4.46, and 4.49).

In order to obtain the appropriate adaptation region for every pixel, we employed a statistical method called intersection of confidence intervals (ICI) rule [Katkovnik 02]. The ICI method gives regions of similar statistical properties for estimating the value of the parameter p (Chapter 5). Although it is computationally demanding, the ICI method significantly improves the estimation of the parameter p.

Finally, in Chapter 6 the image denoising results were presented for both synthetic and real-world images. We have shown that in synthetic images composed of localized periodic components, noise is significantly reduced by using the proposed adaptive wavelet filter banks.

7.2 Future Research

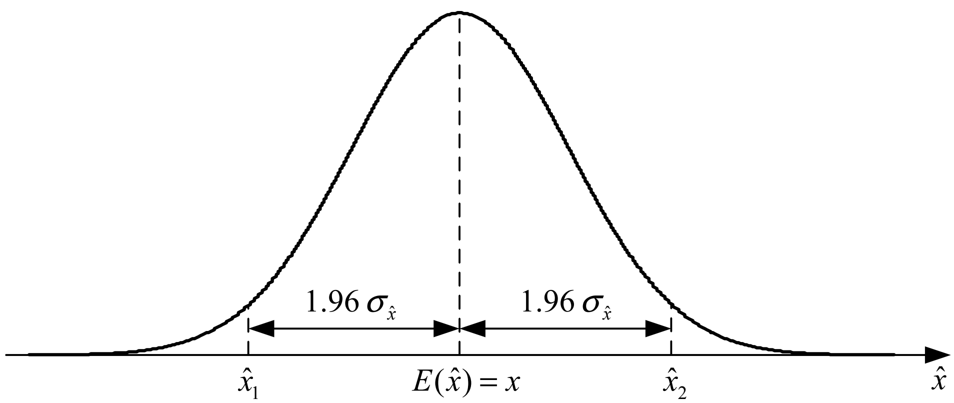
The adaptation of the filter bank results in the p parameters which are different across scales and positions of the analyzed images. In Section 3.4.2, we have shown that if the same value of the parameter p is set in all decomposition levels, then for some range of values of p the limit wavelet and scale functions are regular. However, no explicit expression for the allowed range of p parameters' values was obtained. The impact of the choice of p parameters, which are different across scales and positions, on the regularity of the underlying wavelet functions should be studied in details as an important part of any future research built on the work presented in this thesis.

Our proposed adaptive wavelet filter banks are well suited for representing localized periodical components of the image. However, since the adaptive filter banks are based on the filters of fixed support, the edge localization is not very efficient. It would be interesting to explore how other adaptive structures, that change filters of different supports based on the ICI algorithm, perform. For example, the de Boor-Ron algorithm could be used to build Neville filters of different asymmetrical supports. In this way, the prediction across edges, which happens with the fixed support filters, would be avoided.

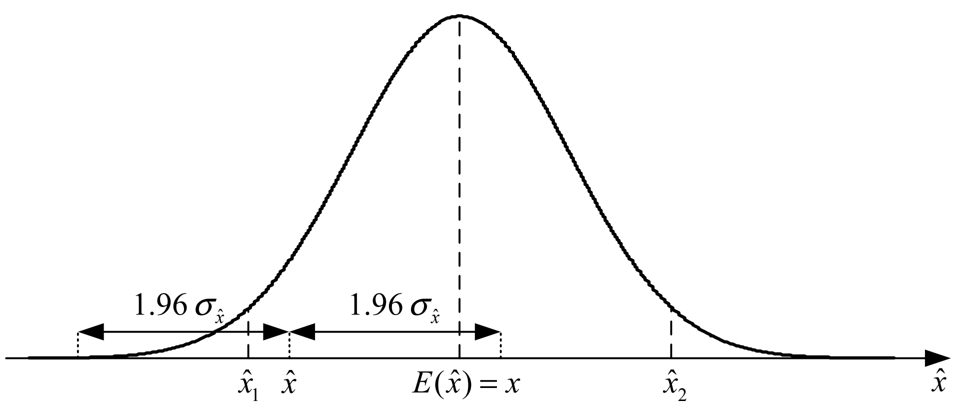
Appendix: Confidence Intervals

In order to give a brief insight into the meaning of the confidence interval [Ross 87] let us consider an estimate $\hat{x}$ of a parameter x. The estimate is unbiased, i.e.

$$E(\hat{x}) = x. \tag{7.1}$$



(a) There is a 95% probability that the estimate $\hat{x}$ is contained within the interval $\langle \hat{x}_1, \hat{x}_2 \rangle = \langle x - 1.96\sigma_{\hat{x}}, x + 1.96\sigma_{\hat{x}} \rangle$...



(b) ...which further means that for a single estimated value $\hat{x}$ there is 95% probability that the true value x is contained within the interval $\langle \hat{x} - 1.96\sigma_{\hat{x}}, \hat{x} + 1.96\sigma_{\hat{x}} \rangle$. Hence, the interval $\langle \hat{x} - 1.96\sigma_{\hat{x}}, \hat{x} + 1.96\sigma_{\hat{x}} \rangle$ is called a 95% confidence interval.

Figure 7.1: Confidence interval for a normal probability distribution of the estimate $\hat{x}$.

For example, if the distribution of $\hat{x}$ is normal as shown in Figure 7.1(a) then it holds that

$$P\left(x - 1.96\sigma_{\hat{x}} < \hat{x} < x + 1.96\sigma_{\hat{x}}\right) = 0.95 \tag{7.2}$$

which means that there is a 95% probability that the estimate $\hat{x}$ obtained in a single trial will be inside the limits posed symmetrically around the true value x. From equation 7.2 it follows that

$$P\left(\hat{x} - 1.96\sigma_{\hat{x}} < x < \hat{x} + 1.96\sigma_{\hat{x}}\right) = 0.95 \tag{7.3}$$

which gives a different perspective to the same estimation problem. It defines a confidence interval of estimating x shown in Figure 7.1(b). The equation states that there is a 95% probability that the parameter x that is being estimated will be inside an interval defined with confidence limits

$$\hat{x} \pm 1.96\sigma_{\hat{x}}. \tag{7.4}$$

Asymmetric distribution

In general, the distribution of an estimate can be asymmetrical as shown in Figure 7.2(a) so the $p\%$ confidence interval shown in Figure 7.2(b) is not symmetrical any more.

From

$$P\left(x - \Delta\hat{x}_1 < \hat{x} < x + \Delta\hat{x}_2\right) = p \tag{7.5}$$

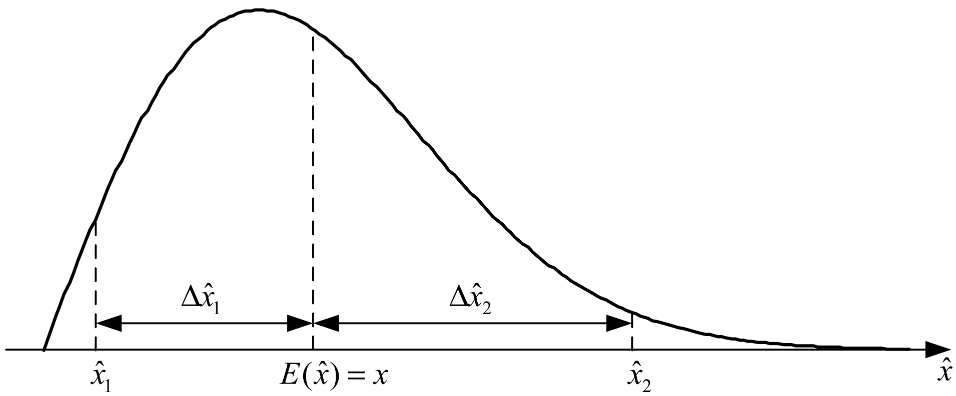
it follows that

$$P\left(\hat{x} - \Delta\hat{x}_2 < x < \hat{x} + \Delta\hat{x}_1\right) = p \tag{7.6}$$

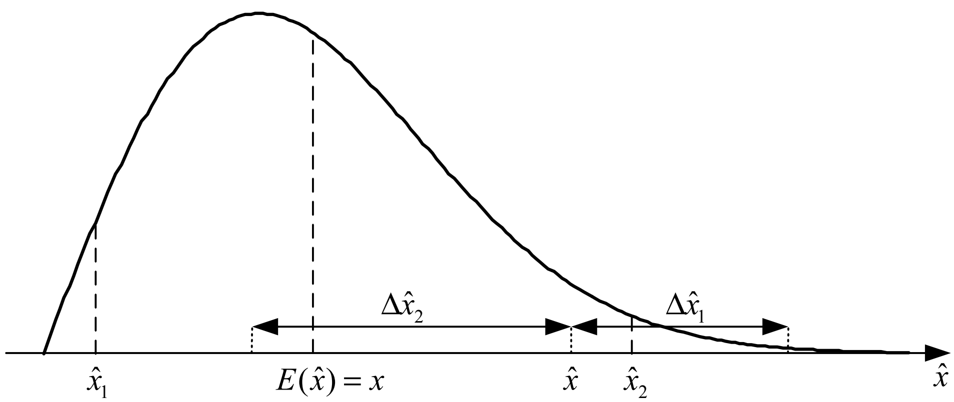
so the confidence interval is obtained as

$$\langle \hat{x} - \Delta\hat{x}_2, \hat{x} + \Delta\hat{x}_1 \rangle \tag{7.7}$$

Note the change of positions of $\Delta\hat{x}_1$ and $\Delta\hat{x}_2$ from equation 7.5 to 7.6.



(a) There is a 95% probability that the estimate $\hat{x}$ is contained within the interval $\langle \hat{x}_1, \hat{x}_2 \rangle = \langle x - \Delta\hat{x}_1, x + \Delta\hat{x}_2 \rangle$...



(b) ...which further means that for a single estimated value $\hat{x}$ there is 95% probability that the true value x is contained within the interval $\langle \hat{x} - \Delta\hat{x}_2, \hat{x} + \Delta\hat{x}_1 \rangle$. Hence, the interval $\langle \hat{x} - \Delta\hat{x}_2, \hat{x} + \Delta\hat{x}_1 \rangle$ is called a 95% confidence interval.

Figure 7.2: Confidence interval for an asymmetrical probability distribution of the estimate $\hat{x}$.

Bibliography

[Ayyub 02] Bilal M. Ayyub. Probability, statistics, and reliability for engineers and scientists, second edition. Chapman & Hall/CRC, June 2002.

[Boulgouris 01] N.V. Boulgouris, D. Tzovaras & M.G. Strintzis. *Lossless image compression based on optimal prediction, adaptivelifting, and conditional arithmetic coding*. IEEE Transactions on Image Processing, vol. 10, no. 1, pages 1–14, 2001.

[Burrus 97] Sidney C. Burrus, Sidney C. Burrus & Ramesh A. Gopinath. Introduction to wavelets and wavelets transforms. Prentice Hall, August 1997.

[Chan 02] Tony F. Chan & H.M. Zhou. *Eno-wavelet transforms for piecewise smooth functions*. SIAM Journal on Numerical Analysis, vol. 40, no. 4, pages 1369–1404, September 2002.

[Chang 00] S.G. Chang, Bin Yu & M. Vetterli. *Spatially adaptive wavelet thresholding with context modeling for image denoising*. IEEE Transactions on Image Processing, vol. 9, no. 9, pages 1522–1531, 2000.

[Chen 93] T. Chen & P.P. Vaidyanathan. *Multidimensional multirate filters and filter banks derived from one-dimensional filters*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 41, no. 5, pages 1749–1765, 1993.

[Claypoole 97] R. Claypoole, G. Davis, W. Sweldens & R. Baraniuk. *Nonlinear wavelet transforms for image coding*. In Signals, Systems & Computers, 1997. Conference Record of the Thirty-First Asilomar Conference on, volume 1, pages 662–667, Pacific Grove, CA, USA, 1997.

[Claypoole 03] R.L. Claypoole, G.M. Davis, W. Sweldens & R.G. Baraniuk. *Nonlinear wavelet transforms for image coding via lifting*. IEEE Transactions on Image Processing, vol. 12, no. 12, pages 1449–1459, 2003.

[Cohen 93] A. Cohen & I. Daubechies. *Non-separable bidimensional wavelet bases*. Revista Matematica Iberoamericana, vol. 9, no. 1, pages 51–137, 1993.

[Cohen 94] Leon Cohen. Time frequency analysis : Theory and applications (prentice-hall signal processing). Prentice Hall PTR, December 1994.

[Coifman 92] R.R. Coifman & M.V. Wickerhauser. *Entropy-based algorithms for best basis selection*. IEEE Transactions on Information Theory, vol. 38, pages 713–718, 1992.

[Coifman 95] R. R. Coifman & D. L. Donoho. *Translation-Invariant De-Noising*. Rapport technique, Stanford University, Department of Statistics, 1995. Wavelets and Statistics, Anestis Antoniadis, ed. Springer-Verlag Lecture Notes.

[Conway 93] J. H. Conway & N. J. A. Sloane. Sphere packings, lattices, and groups. Springer-Verlag, 1993.

[Daubechies 88] I. Daubechies. *Orthonormal Bases of Compactly Supported Wavelets*. Comm. Pure Applied Math., vol. XLI, no. 41, pages 909–996, November 1988.

[Daubechies 92] I. Daubechies. Ten lectures on wavelets. SIAM, Philadelphia, PA, 1992. Notes from the 1990 CBMS-NSF Conference on Wavelets and Applications at Lowell, MA.

[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.

[Donoho 95a] D. L. Donoho & I. M. Johnstone. *Adaptating to Unknown Smoothness via Wavelet Shrinkage*. J. Amer. Stat. Assoc., no. 90, pages 1200–1224, 1995.

[Donoho 95b] D.L. Donoho. *De-noising by soft-thresholding*. IEEE Transactions on Information Theory, vol. 41, no. 3, pages 613–627, 1995.

[Dubois 85] Eric Dubois. *The sampling and reconstruction of time-varying imagery with application in video systems*. Proceedings of the IEEE, vol. 73, no. 4, pages 502–522, Apr 1985.

[Dudgeon 83] Dan E. Dudgeon & Russell M. Mersereau. Multidimensional digital signal processing (prentice-hall signal processing series). Prentice Hall, 1983.

[Gerek 00] O.N. Gerek & A.E. Cetin. *Adaptive polyphase subband decomposition structures for imagecompression*. IEEE Transactions on Image Processing, vol. 9, no. 10, pages 1649–1660, 2000.

[Gerek 06] O.N. Gerek & A.E. Cetin. *A 2-D orientation-adaptive prediction filter in lifting structures for image coding*. IEEE Transactions on Image Processing, vol. 15, no. 1, pages 106–111, 2006.

[Goldenshluger 97] A. Goldenshluger & A. Nemirovski. *Adaptive de-noising of signals satisfying differential inequalities*. IEEE Transactions on Information Theory, vol. 43, no. 3, pages 872–889, 1997.

[Gouze 04] A. Gouze, M. Antonini, M. Barlaud & B. Macq. *Design of signal-adapted multidimensional lifting scheme for lossy coding*. IEEE Transactions on Image Processing, vol. 13, no. 12, pages 1589–1603, 2004.

[Grossmann 84] A. Grossmann & J. Morlet. *Decomposition of Hardy functions into square integrable wavelets of constant shape*. SIAM J. Math. Anal., vol. 15, no. 4, pages 723–736, 1984.

[Harten 97] A. Harten, B. Engquist, S. Osher & S. R. Chakravarthy. *Uniformly High Order Accurate Essentially Non-oscillatory Schemes, III*. Journal of Computational Physics, vol. 131, no. 1, 1997.

[Haykin 86] Simon Haykin. Adaptive filter theory. Prentice-Hall Information and System Science Series. Prentice-Hall, Englewood Cliffs,New Jersey, 1986.

[Johnstone 97] I. M. Johnstone & B. W. Silverman. *Wavelet threshold estimators for data with correlated noise*. Journal of the Royal Statistical Society B, vol. 59, no. 2, pages 319–351, 1997.

[Karlsson 90] G. Karlsson & M. Vetterli. *Theory of two-dimensional multirate filter banks*. Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on, vol. 38, no. 6, pages 925–937, 1990.

[Katkovnik 98] V. Katkovnik & L. Stankovic. *Instantaneous frequency estimation using the Wigner distribution with varying and data-driven window length*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 46, no. 9, pages 2315–2325, 1998.

[Katkovnik 99] V. Katkovnik. *A new method for varying adaptive bandwidth selection*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 47, no. 9, pages 2567–2571, 1999.

[Katkovnik 02] V. Katkovnik, K. Egiazarian & J. Astola. *Adaptive window size image denoising based on intersection of confidence intervals (ICI) rule*. Math. Imaging and Vision, vol. 16, pages 223–235, 2002.

[Katznelson 76] Y. Katznelson. An introduction to harmonic analysis. Dover Publications, 1976.

[Kivanc 99] M. Kivanc, I. Kozintsev & K. Ramchandran. *Spatially adaptive statistical modeling of wavelet imagecoefficients and its application to denoising*. In Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on, volume 6, pages 3253–3256, Phoenix, AZ, USA, 1999.

[Kovacevic 92] J. Kovacevic & M. Vetterli. *Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for R^n*. IEEE Transactions on Information Theory, vol. 38, pages 533–555, 1992.

[Kovacevic 95] J. Kovacevic & M. Vetterli. *Nonseparable two- and three-dimensional wavelets*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 43, no. 5, pages 1269–1273, 1995.

[Kovacevic 00] J. Kovacevic & W. Sweldens. *Wavelet families of increasing order in arbitrary dimensions*. IEEE Transactions on Image Processing, vol. 9, no. 3, pages 480–496, 2000.

[Mallat 89a] S.G. Mallat. *Multifrequency channel decompositions of images and wavelet models*. Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on, vol. 37, no. 12, pages 2091–2110, 1989.

[Mallat 89b] S.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, 1989.

[Mallat 99] Stèphane Mallat. A wavelet tour of signal processing, second edition (wavelet analysis & its applications). Academic Press, September 1999.

[Mertins 99] A. Mertins. Signal analysis: Wavelets, filter banks, time-frequency transforms and applications. John Wiley & Sons, 1999.

[Meyer 95] Yves Meyer. Wavelets and operators: Vol. 37 (cambridge studies in advanced mathematics). Cambridge University Press, January 1995.

[Piella 02a] G. Piella & H.J.A.M. Heijmans. *Adaptive lifting schemes with perfect reconstruction*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 50, no. 7, pages 1620–1630, 2002.

[Piella 02b] G. Piella, B. Pesquet-Popescu & H. Heijmans. *Adaptive update lifting with a decision rule based on derivative filters*. IEEE Signal Processing Letters, vol. 9, no. 10, pages 329–332, 2002.

[Ross 87] Sheldon M. Ross. Introduction to probability and statistics for engineers and scientists. John Wiley & Sons, June 1987.

[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 02] 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.

[Simoncelli 90] E.P. Simoncelli & E.H. Adelson. *Non-separable extensions of quadrature mirror filters to multiple dimensions*. Proceedings of the IEEE, vol. 78, no. 4, pages 652–664, 1990.

[Stankovic 98] L. Stankovic & V. Katkovnik. *Algorithm for the instantaneous frequency estimation using time-frequency distributions with adaptive window width*. IEEE Signal Processing Letters, vol. 5, no. 9, pages 224–227, 1998.

[Stankovic 04] L. Stankovic. *Performance analysis of the adaptive algorithm for bias-to-variance tradeoff*. Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on], vol. 52, no. 5, pages 1228–1234, 2004.

[Strang 96] Gilbert Strang & Truong Nguyen. Wavelets and filter banks. Wellesley College, 1996.

[Sweldens 95] W. Sweldens. *The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions*. In A. F. Laine & M. Unser, editeurs, Wavelet Applications in Signal and Image Processing III, pages 68–79. Proc. SPIE 2569, 1995.

[Sweldens 96a] W. Sweldens. *The lifting scheme: A custom-design construction of biorthogonal wavelets*. Appl. Comput. Harmon. Anal., vol. 3, no. 2, pages 186–200, 1996.

[Sweldens 96b] W. Sweldens. *Wavelets and the lifting scheme: A 5 minute tour*. Z. Angew. Math. Mech., vol. 76 (Suppl. 2), pages 41–44, 1996.

[Sweldens 97] W. Sweldens. *The lifting scheme: A construction of second generation wavelets*. SIAM J. Math. Anal., vol. 29, no. 2, pages 511–546, 1997.

[Theussl 01] T. Theussl, T. Moller & M.E. Groller. *Optimal regular volume sampling*. In Proceedings of the IEEE Visualization Conference, pages 91–98, San Diego, CA, October 2001.

[Vaidyanathan 92] P. P. Vaidyanathan. Multirate systems and filter banks (prentice hall signal processing series). Prentice Hall PTR, September 1992.

[Vetterli 95] Martin Vetterli & Jelena Kovacevic. Wavelets and subband coding. Prentice Hall PTR, April 1995.

[Viscito 91] E. Viscito & J.P. Allebach. *The analysis and design of multidimensional FIR perfect reconstruction filter banks for arbitrary sampling lattices*. IEEE Transactions on Circuits and Systems, vol. 38, no. 1, pages 29–41, 1991.

[Vrankić 03] M. Vrankić. Msc thesis. Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia, July 2003.

[Vrankić 04] M. Vrankić & D. Seršić. *Image Denoising Based on Adaptive Quincunx Wavelets*. In Proc. of the 2004 IEEE 6th Workshop on Multimedia Signal Processing, pages 251–254, Siena, Italy, September 2004.

Abstract

In this thesis, we propose the novel adaptive wavelet filter bank structures that are used to obtain efficient representations of the analyzed images.

We present the lifting scheme structures for building adaptive wavelet decompositions based on the nonseparable quincunx sampling scheme. The resulting wavelet decompositions are adaptive to the local properties of the analyzed image. Despite the introduced adaptation, a desired number of vanishing moments is still retained.

The proposed adaptation is performed in order to minimize the energy of detail coefficients on a neighborhood of each pixel of the analyzed image. The appropriate neighborhood is determined for each pixel separately by using the intersection of confidence intervals (ICI) rule. The application of the ICI rule improves the estimation of the filter bank parameters and makes it more robust to noise.

The image denoising results are presented for both synthetic and real-world images. It is shown that the adaptive wavelet decompositions outperform the existing fixed decompositions in terms of denoising quality of images that contain periodic components, and in general they give more compact image representations.

Keywords: wavelet transforms, adaptive filters, second generation wavelets, adaptive lifting scheme, quincunx sampling, interpolating filters, intersection of confidence intervals, image denoising.

Sažetak

U ovoj disertaciji predlažemo nove adaptivne valićne filtarske strukture koje se koriste za dobivanje efikasnih reprezentacija analiziranih slika.

U radu prikazujemo filtarske slogove temeljene na shemi podizanja za konstrukciju adaptivnih valićnih razlaganja zasnovanih na neseparabilnom quincunx uzorkovanju. Rezultirajuća valićna razlaganja su prilagođena lokalnim svojstvima analizirane slike. Unatoč uvedenoj adaptaciji, zadržan je željeni broj nul-momenata pripadajućih valićnih funkcija.

Adaptacija se vrši sa ciljem minimizacije koeficijenata detalja na okolini svakog slikovnog elementa. Odabir odgovarajuće okoline za svaki slikovni element vrši se korištenjem metode presjecišta invervala pouzdanosti. Primjena dotične metode poboljšava adaptaciju parametara filtarskog sloga i čini je robusnijom na šum.

Rezultati uklanjanja šuma iz slike prikazani su za primjere sintetičkih kao i realnih slika. Pokazano je da dobivena adaptivna valićna razlaganja nadmašuju postojeća fiksna razlaganja s obzirom na kvalitetu uklanjanja šuma iz slika koje sadrže periodičke komponente, te općenito daju kompaktniji zapis slike.

Ključne riječi: valićne transformacije, adaptivni filtri, valići druge generacije, adaptivna shema podizanja, quincunx uzorkovanje, interpolacijski filtri, presjecište intervala pouzdanosti, uklanjanje šuma iz slike.

Curriculum Vitae

Basic Information

Name:	**Miroslav Vrankić**
Position:	Research and teaching assistant at the Faculty of Engineering, University of Rijeka, Croatia.
E-mail:	Miroslav.Vrankic@RITEH.hr
WWW:	http://www.zesoi.fer.hr/∼mvrankic

Education

1999-2003	Master of Sciences Degree (MS) in Electrical Engineering, Faculty Of Electrical Engineering and Computing, University of Zagreb, Croatia.
1994-1999	Diploma in Electrical Engineering (dipl.ing.) Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia.

Knowledge of Languages

Languages:

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

Professional Experience

2004-2006	Research and teaching assistant at the Faculty of Engineering, University of Rijeka, Croatia.
2003-2004	Visiting researcher at the Tampere International Center for Signal Processing, Tampere University of Technology, Finland.
1999-2003	Research and teaching assistant at the Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia.

Teaching Activities

Signals and Systems	2004-2006
Random Processes in Systems	2002
Digital Signal Processing	2001-2006
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. M. Vrankić, K. Egiazarian & A. Gotchev. *Image Denoising by Combined Quincunx and Separable Wavelet-Domain Wiener Filtering*. In Proc. of the 6th Nordic Signal Processing Symposium, pages 113–116, Espoo, Finland, June 2004.
2. M. Vrankić, K. Egiazarian & A. Gotchev. *Phase Shifting in the Quincunx Filter Bank*. In Proc. of the 6th Int. Conf. on Digital Signal Processing and its Applications, pages 5–7, Moscow, Russia, April 2004.

3. M. Vrankić & D. Seršić. *Adaptation in the Quincunx Wavelet Filter Bank with Applications in Image Denoising*. In Proc. of the 2004 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, pages 11–12, Wienna, Austria, September 2004.

4. M. Vrankić & D. Seršić. *Image Denoising Based on Adaptive Quincunx Wavelets*. In Proc. of the 2004 IEEE 6th Workshop on Multimedia Signal Processing, pages 251–254, Siena, Italy, September 2004.

5. D. Seršić, M. Vrankić & I. Štajduhar. *Lossy Image Reconstruction Using Adaptive Wavelet Filter Bank*. Fifth IASTED International Conference on Signal and Image Processing, pages 235–242, Honolulu, Hawaii, USA, August 2003.

6. 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.

7. 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.

8. 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.

9. M. Vrankić. *WAP - Concepts, Technologies and Open Solutions*. Open Systems' Days, Zagreb, Croatia, April 2001.

10. M. Vrankić. *WAP and Open Source Community: Kannel Gateway*. In Proc. of Communication Technologies and Standards in Informatics, Opatija, Croatia, December 2000.

Životopis

Osnovni podaci

Ime:	**Miroslav Vrankić**
Radno mjesto:	Asistent Tehnički fakultet, Sveučilište u Rijeci.
E-mail:	Miroslav.Vrankic@RITEH.hr
WWW:	http://www.zesoi.fer.hr/∼mvrankic

Školovanje

1999-2003	Magistar znanosti u području Tehničkih znanosti, polje Elektrotehnika, smjer Opća elektronika Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu.
1994-1999	Diplomirani inženjer elektrotehnike, smjer Industrijska elektronika, Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu.

Poznavanje stranih jezika

Jezici:

1. Engleski
2. Talijanski

Radno iskustvo

2004-2006	Asistent Tehnički fakultet, Sveučilište u Rijeci.
2003-2004	Gostujući istraživač Tampere International Center for Signal Processing, Tampere University of Technology, Finska.
1999-2003	Znanstveni novak Fakultet elektrotehnike i računarstva, Sveučilište u Zagrebu.

Sudjelovanje u nastavi

Signali i sustavi	2004-2006
Slučajni procesi u sustavima	2002
Digitalna obrada signala	2001-2006
Sustavi za praćenje i vođenje procesa	2000-2001
Programska podrška mjernih i procesnih sustava	1999-2000
Podatkovni višemedijski prijenos i računalne mreže	1999-2000

Objavljeni radovi

1. M. Vrankić, K. Egiazarian & A. Gotchev. *Image Denoising by Combined Quincunx and Separable Wavelet-Domain Wiener Filtering*. In Proc. of the 6th Nordic Signal Processing Symposium, pages 113–116, Espoo, Finland, June 2004.

2. M. Vrankić, K. Egiazarian & A. Gotchev. *Phase Shifting in the Quincunx Filter Bank*. In Proc. of the 6th Int. Conf. on Digital Signal Processing and its Applications, pages 5–7, Moscow, Russia, April 2004.

3. M. Vrankić & D. Seršić. *Adaptation in the Quincunx Wavelet Filter Bank with Applications in Image Denoising*. In Proc. of the 2004 International TICSP Workshop on Spectral Methods and Multirate Signal Processing, pages 11–12, Wienna, Austria, September 2004.

4. M. Vrankić & D. Seršić. *Image Denoising Based on Adaptive Quincunx Wavelets*. In Proc. of the 2004 IEEE 6th Workshop on Multimedia Signal Processing, pages 251–254, Siena, Italy, September 2004.

5. D. Seršić, M. Vrankić & I. Štajduhar. *Lossy Image Reconstruction Using Adaptive Wavelet Filter Bank*. Fifth IASTED International Conference on Signal and Image Processing, pages 235–242, Honolulu, Hawaii, USA, August 2003.

6. 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.

7. 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.

8. 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.

9. M. Vrankić. *WAP - Concepts, Technologies and Open Solutions*. Open Systems' Days, Zagreb, Croatia, April 2001.

10. M. Vrankić. *WAP and Open Source Community: Kannel Gateway*. In Proc. of Communication Technologies and Standards in Informatics, Opatija, Croatia, December 2000.