

# DSP of 1D and 2D signals by DFT and DWT using Matlab

Thomas Strehl, e7701969, TU-Vienna

2014 11 11

## Contents

<b>1 Introduction</b>	<b>3</b>
<b>2 Time and Frequency Domain</b>	<b>4</b>
2.1 Fourier Transform . . . . .	5
2.2 Fourier Series . . . . .	5
2.3 Fourier Transform for Series . . . . .	5
2.4 Discrete Fourier Transform . . . . .	6
<b>3 Discrete Time Signals</b>	<b>6</b>
3.1 Linear Time Invariant Systems . . . . .	6
3.2 Single frequency signal . . . . .	7
3.3 Discrete Fourier Transform . . . . .	8
3.4 Harmonic Analysis . . . . .	9
3.5 Signal and spectra examples . . . . .	9
<b>4 Short Time Fourier Transform</b>	<b>12</b>
4.1 Windowing to limited length signals . . . . .	12
4.2 Window types, leakage and zero padding . . . . .	13
4.3 Gabor transform . . . . .	14
4.4 Example: Wave audio spectrum . . . . .	15
<b>5 Wavelets</b>	<b>16</b>
5.1 Introduction . . . . .	17
5.2 Wavelet construction . . . . .	17
5.3 Multiresolution Analysis . . . . .	18
5.4 Wavelet Properties . . . . .	19
5.5 Example: 1D RS signal . . . . .	20
5.6 Application: Denoising . . . . .	21
<b>6 Transformation of Images</b>	<b>24</b>
6.1 2D Fourier Transform . . . . .	24
6.2 DFT of sample RS image . . . . .	25
6.3 2D Wavelets . . . . .	25
6.4 DWT of sample RS image . . . . .	27
<b>7 Denoising and Local Contrast Enhancement</b>	<b>29</b>
7.1 Denoising and coefficient Enhancement . . . . .	29
7.2 Center-Surround detail enhancement . . . . .	30
7.3 Package dspCers.wavelet . . . . .	32
<b>8 Experiments and Results</b>	<b>32</b>
8.1 wimCoefDenEnh: Denoising . . . . .	32
8.2 wimCoefDenEnh: scale level contrast enhancement . . . . .	33
8.3 wimCoefCenSur: center-surround contrast enhancement . . . . .	34
<b>9 Discussion and Conclusion</b>	<b>35</b>
<b>References</b>	<b>36</b>

## Abstract

This work demonstrates signal processing basics including the discrete Fourier transform, windowing techniques for short time spectral analysis and the discrete wavelet transform. Each topic is first introduced theoretically and then demonstrated using Matlab sample code. The properties of wavelet based signal analysis are explored for several different wavelet families and decomposition depths. Two new thresholding methods for denoising, 'equalEnergy' and 'equalMax', are added to the Matlab 'Wavelet' package supplied 'Soft' and 'Hard' options. Local contrast enhancement of images is demonstrated with two methods. The first is based on denoising and rescaling of the detail coefficients of the wavelet decomposition, the second on the inter-scale level approximation coefficient ratio of an undecimated transform. An unsharp masking inspired amplification of contrast enhancement for the latter method is proposed in this work. The methods are investigated using 1D and 2D signal samples originally acquired for archaeological prospection. The implementation uses Matlab object orientated programming techniques collected in package 'dspCers'.

## 1 Introduction

The Fourier and wavelet transforms are designed to translate time signals into a frequency domain representation. The Fourier transform is based on the observation that any time signal can be decomposed into a superposition of oscillations that can be written as an infinite series of basis frequencies. Due to the infinite nature of these (complex) waves, no information about when a certain frequency occurs can be obtained. The introduction of a windowing function that restricts the Fourier transform to a section of the signal allows for a limited time resolution of the spectral analysis, at the cost of an impacted spectrum. The wavelet transform extends these concepts by probing the signal with basis functions that exhibit limited extension and internal structure. Similar to windowing, localization in the time domain is obtained from translation of the probe along the signal. Dilation of the probe transforms the wave based concept of frequency to the wavelet based notion of scale. This method of signal decomposition directly relates to the pyramids used in image analysis.

The present work will demonstrate how wavelets can be applied for denoising and local contrast enhancement of images using remote sensing data originally acquired for archaeological prospection. Signal processing in archaeological prospection is about classification of groups of pixels into natural vs. remains of human activity, based on the pixels and their surroundings intensity. The structures of interest often range from a very few to several pixels, giving a first hint to relevant properties in the frequency and scale domains. Suppression of non-relevant signal components and enhancement of relevant parts are important tasks of preprocessing for the higher level downstream analysis.

RS signals contain pixel noise originating in several stages of the imaging process. The noise amplitude can be estimated from the signal variance in areas of constant surface reflectivity. Other noise-like signal components emerge from the natural variance of reflectivity of grassland, agricultural areas, woods, snow, rocks or water bodies. Structures of interest consist of either crop marks (vegetation stressed due to subsurface anomalies) or remains of buildings erected in grassland areas.

The 1D sample signal shown in Fig. 1.1 is constructed from several cross-sections through relevant parts of archaeological sites illustrated in Fig. 1.2. These data have been acquired by the Geoye1 platform in the near infrared at a resolution of 0.5 m per pixel, a dynamic range of 11bit and are rescaled to the [0 1] range. The 1D sample is shifted for zero mean.

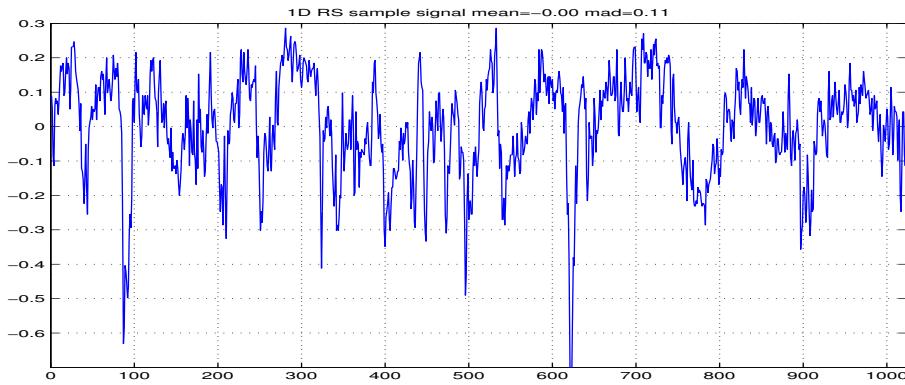


Figure 1.1: 1D sample signal

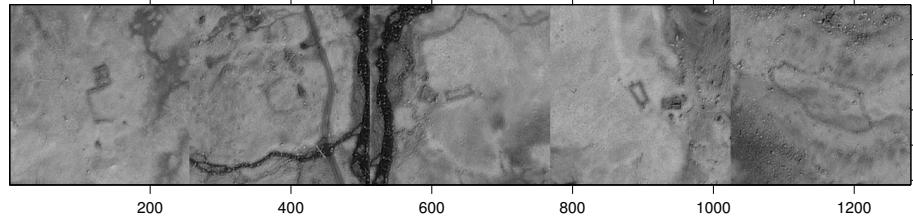


Figure 1.2: 2D sample signals

The document is organized as follows: The Sections 'Time and Frequency Domain', 'Discrete Time Signals', 'Linear Time Invariant Systems' and 'Short Time Fourier Transform' lay out selected fundamentals of digital signal processing and path the way to motivate the subsequent focus on wavelets.

Section 'Wavelets' introduces the concept and basic properties of wavelets, Section 'Transformation of Images' extends the method to 2D signals. Section 'Denoising and Local Contrast Enhancement' gives details on the methods implemented in Matlab package 'dspCers', with results presented in Section 'Experiments and Results'. The document concludes with some comments in Section 'Discussion and Conclusion'. All wavelet related sections use the 1D and 2D RS sample signals (see Figs. 1.1 and 1.2) to demonstrate algorithms and properties.

## 2 Time and Frequency Domain

The mathematical framework describing a signal in its time and frequency domain representation is based on the power series (see Eq. 2.1), and the Laplace transform (see Eq. 2.2), its analog counterpart with  $x = e^{-s}$ .

$$f(x) = \sum a_n x^n \quad (2.1)$$

$$F(s) = \int f(t) e^{-st} dt \quad (2.2)$$

The representations of a signal in the time and frequency domains exhibit characteristic properties, that take on dual values [Werner, 2012]:

- domain: infinite, periodic
- density: analog, discrete

Combinations of the two make for a total of four possible relations (see Sections 2.1 to 2.4). Changing a property in one space flips the other in its dual space. E.g. cutting out a part of an infinite analog signal, that is, making it periodic, turns its frequency representation from an analog to a discrete line spectrum.

time\frequency	$\downarrow$ analog	$\downarrow$ discrete
unlimited $\rightarrow$	$\downarrow$ unlimited $\rightarrow$ analog	$\downarrow$ periodic $\rightarrow$ analog
periodic $\rightarrow$	$\downarrow$ unlimited $\rightarrow$ discrete	$\downarrow$ periodic $\rightarrow$ discrete

Table 2.1: Dual properties in time and frequency domains

In the mathematical formulation relating the time and frequency domain representations of a signal, the aggregation corresponding to density is either a sum or an integral, and the domain sums over an unlimited or limited range.

The formulas in Sections 2.2 to 2.4 are taken from [Werner, 2012].

## 2.1 Fourier Transform

For a signal that is infinite (non-periodic) and continuous in the time domain the frequency spectrum is likewise infinite and continuous.

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(j\omega) e^{j\omega t} d\omega \quad (2.3)$$

$$X(j\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt \quad (2.4)$$

Here the spectrum is formulated in units of the angular frequency  $\omega=2\pi f$ . The integrals represent an inner product of the signal and a set of orthogonal basis functions, such that the  $X(j\omega)$  constitute the coefficients of the projection of the signal onto the basis functions.

## 2.2 Fourier Series

Making a continuous signal periodic transforms the representation in the frequency domain into a discrete spectrum.

$$x(t) = \sum_{k=-\infty}^{+\infty} c_k e^{j2\pi f_0 kt} \quad (2.5)$$

$$c_k = \frac{1}{T_0} \int_{t_0}^{t_0+T} x(t) e^{-j2\pi f_0 kt} dt \quad (2.6)$$

The lines of the discrete spectrum are located at frequencies  $kf_0$  with base frequency  $f_0 = \frac{1}{T_0}$ , where  $T_0$  is the period of the time signal.

## 2.3 Fourier Transform for Series

Switching from continuous to discrete in the time domain (see Discrete Time Signals for details) yields a periodic spectrum for the normalized frequency  $\Omega$  with period  $2\pi$ .

$$x[n] = \frac{1}{2\pi} \int_{2\pi} X(e^{j\Omega}) e^{j\Omega n} d\Omega \quad (2.7)$$

$$X(e^{j\Omega}) = \sum_{n=-\infty}^{+\infty} x[n] e^{-j\Omega n} \quad (2.8)$$

## 2.4 Discrete Fourier Transform

A signal that is periodic and discrete in the time domain exhibits the same behavior in the frequency domain. The line spectrum consists of the normalized frequencies  $\frac{k}{N}$ , starting from  $\frac{1}{N}$ , the base frequency related to the length of the period in the time domain. See Section 3.3 for details.

# 3 Discrete Time Signals

A discrete signal is considered a sample taken from the continuous world by means of a process based on the Dirac distribution  $\delta(x)$  and its sifting property [Sonka et al., 2007], [Werner, 2012]

$$f(\lambda) = \int_{-\infty}^{+\infty} x(t) \delta(t - \lambda) dt, \quad \int_{-\infty}^{+\infty} \delta(t) dt = 1 \quad (3.1)$$

The process can be understood as sampling a signal  $x(t)$  with a pulse train  $p(t)$

$$x_s(t) = x(t)p(t) = x(t) \sum_{n=-\infty}^{+\infty} \delta(t - nT_s) \quad (3.2)$$

After sampling the signal is different from zero only for discrete points in time  $nT_s$ , where  $T_s$  is the sampling interval.

The spectrum of this sampled signal is calculated as the convolution of the spectra of its multiplicative components [Werner, 2012]

$$X_s(j\omega) = \frac{1}{2\pi} X(j\omega) * P(j\omega) \quad (3.3)$$

$$X_s(j\omega) = \frac{1}{2\pi} X(j\omega) * \frac{2\pi}{T_s} \sum_{k=-\infty}^{+\infty} \delta\left(\omega - \frac{2\pi k}{T_s}\right) = \frac{1}{T_s} \sum_{k=-\infty}^{+\infty} X\left(j\left(\omega - \frac{2\pi k}{T_s}\right)\right) \quad (3.4)$$

which is essentially the periodic variant of the spectrum of the analog input signal with a period of  $\frac{2\pi}{T_s}$ , where  $T_s$  is the length of the sampling interval in  $p(t)$ . The spectrum of the original analog signal can be isolated provided the periods do not overlap, that is, the sampling theorem is adhered to. Complying with the sampling theorem translates to  $T_s < \frac{1}{2} \frac{1}{f_0}$ , where  $f_0$  is the highest frequency in the analog signal. Violation of the sampling theorem leads to overlap of the periodic spectra, called Aliasing. As established in Section 2, discretizing in the time domain translates to periodizing in the frequency domain.

## 3.1 Linear Time Invariant Systems

A time discrete system transforms an input  $x[n]$  like

$$y[n] = T\{x[n]\} \quad n \in N_0 \quad (3.5)$$

For a time linear system (LTI), the response to a sum of input signals equals the sum of the responses to the individual signals. Formulating the input signal as

$$x[n] = \sum_{k=-\infty}^{+\infty} x[k] \delta[n-k] \quad (3.6)$$

and using the superposition principle, the output signal can be written as the convolution sum of the input signal with the impulse response. In the time domain this is formulated as [Doblinger, 2001]

$$y[n] = T \left\{ \sum_{k=-\infty}^{+\infty} x[k] \delta[n-k] \right\} = \sum_{k=-\infty}^{+\infty} x[k] h[n-k] = x[n] * h[n] \quad (3.7)$$

where  $h[n]$  is the so called impulse response of the system, giving a full description of the system. Another way to fully describe the behavior of a LTI system is its frequency response  $H(e^{j\Omega})$ . Any frequency  $e^{j\Omega_0}$  present in the input signal is reproduced on the output (aside from a change in amplitude). Thus the output signal can be written as [Doblinger, 2001]

$$y[n] = H(e^{j\Omega_0}) x[n] = H(e^{j\Omega_0}) e^{j\Omega_0 n} \quad (3.8)$$

In the frequency domain the output signal is given by

$$Y(e^{j\Omega}) = H(e^{j\Omega}) X(e^{j\Omega}) \quad (3.9)$$

As can be seen from Eq. (3.7) and Eq. (3.9), the impulse response and the frequency response are connected by the Fourier transform. Convolution in one domain is represented by a product of the dual signal representation in the other domain and vice versa.

## 3.2 Single frequency signal

The amplitude of an analog signal of a single frequency  $f$  is given as Doblinger [2001]

$$x(t) = \sin(2\pi f t) = \sin(\omega t) \quad (3.10)$$

where  $\omega = 2\pi f$  is the angular speed in radians. Sampling this signal at regular intervals  $T_s$  yields

$$x(nT_s) = \sin(2\pi f n T_s) = \sin(\omega n T_s) \quad (3.11)$$

Defining  $x[n] = x(nT_s)$  and the normalized frequency as  $f_0 = fT_s$  (the frequency in fractions of the sampling frequency, usually smaller than 1), the amplitude can be written as

$$x[n] = \sin 2\pi f_0 n = \sin(\omega_0 n) \quad (3.12)$$

Letting the sampling frequency  $F_s = \frac{1}{T_s}$  be a multiple  $N = \frac{1}{f_0} = \frac{1}{T_s f} = \frac{F_s}{f}$  of the frequency  $f$ , the signal can be written as

$$x[n] = \sin\left(\frac{2\pi}{N}n\right) \quad (3.13)$$

Fig. 3.1 shows an analog signal and two discrete versions sampled at different rates. Units on the x-axis are wall time in Fig. 3.1a) and the sample number in Fig. 3.1b).

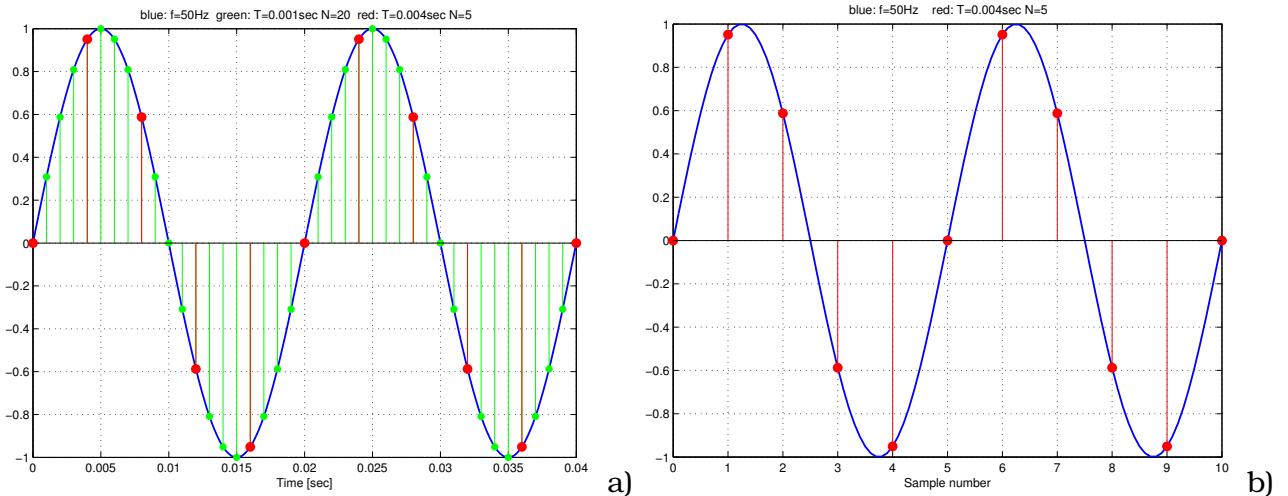


Figure 3.1: 50Hz signal sampled at different rates. a) 0.001sec=1kHz b) 0.004sec=250Hz

### 3.3 Discrete Fourier Transform

The discrete Fourier relates a periodic, discrete time signal with it's dual representation in the frequency domain. Because the signal is periodic in the time domain it's spectrum is discrete, and because the signal is discrete in the time domain it's spectrum is periodic. The DFT and it's inverse, the IDFT, are defined as [Werner, 2012], [Doblinger, 2001]

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi \frac{kn}{N}} \quad (3.14)$$

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi \frac{kn}{N}} \quad (3.15)$$

where  $N$  is the length of the period of the signal, and  $X[k]$  is the intensity of the spectrum line  $k$  of normalized frequency  $\frac{k}{N}$ , with  $\frac{1}{N}$  the normalized base frequency. The spectrum thus consists of as many spectral lines as there are amplitude measurements in the time domain. Thus, any discrete, periodic series, in principle, can be regarded as either a spectral or a time signal. Substituting

$$w_N = e^{-j2\pi \frac{1}{N}} \quad (3.16)$$

the DFT can be written as

$$X[k] = \sum_{n=0}^{N-1} x[n] w_N^{kn} \quad k = 0, 1, \dots, N-1 \quad (3.17)$$

with  $n$  indexing the discrete time signal and  $k$  the spectral lines. Eq. (3.15) can be understood as an inner product, where the signal in the time domain is projected onto each component of the frequency domain. In particular,

$$\frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi}{N} kn} = \frac{1}{N} \sum_{n=0}^{N-1} w_N^{kn} = \begin{cases} 1 & 0 \\ 0 & \text{otherwise} \end{cases} \quad \text{orthonormal} \quad (3.18)$$

forms an orthonormal basis with Eq. (3.18) summing up to one for  $k = mN$  and to zero otherwise.

### 3.4 Harmonic Analysis

A rectangular signal is approximated by the Fourier series [Werner, 2012]

$$x(t) = \frac{1}{2} + \frac{2}{\pi} \sum_{n=0}^{\infty} \frac{1}{2n+1} \sin((2n+1)2\pi t) \quad (3.19)$$

The first one and a half periods of the rectangular signal are shown in Fig. 3.2, together with its synthesis from a Fourier series of 1, 2, 3, 10 and 50 components.

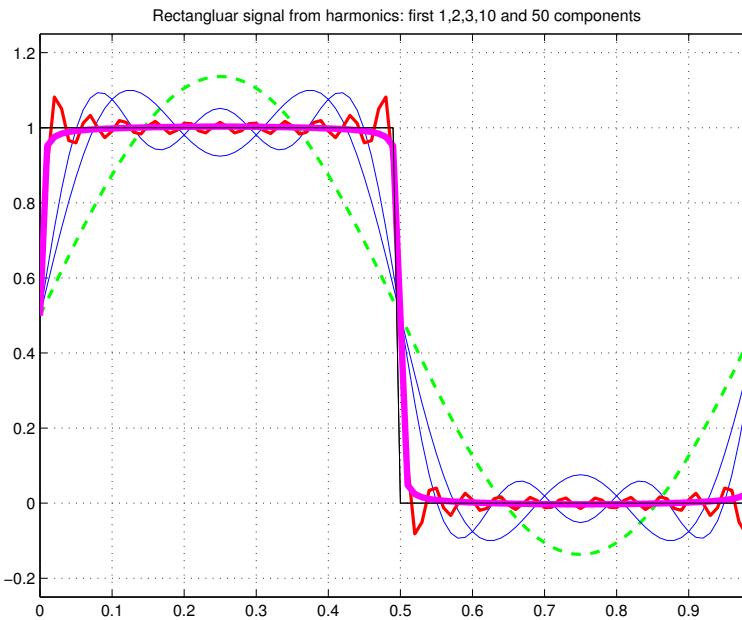


Figure 3.2: Rectangular signal from harmonics (Fourier series)

### 3.5 Signal and spectra examples

Any function  $f(t)$  can be decomposed into an even ( $f(t) = f(-t)$ ) and odd ( $f(t) = -f(-t)$ ) part [Sonka et al., 2007]

$$f_e(t) = \frac{f(t) + f(-t)}{2}, \quad f_o(t) = \frac{f(t) - f(-t)}{2} \quad (3.20)$$

From the definition of

$$e^{i\phi} = \cos(\phi) + i * \sin(\phi) \quad (3.21)$$

it can be seen that  $\cos(\phi)$  and  $\sin(\phi)$  are the even and odd parts of  $e^{i\phi}$ , respectively.

The following sections give examples of the DFT spectrum as calculated by Eq. (3.15) where  $x[n]$  is  $\cos()$ ,  $\sin()$ , and  $\exp()$ . In each case the sample size  $N=32$ . The spectrum consists of one or more lines that are either multiples of the sampling frequency (Sections 3.5.1 and 3.5.2) or not (Section 3.5.3). The plots shown include the real and imaginary parts of the signal and it's spectrum.

#### 3.5.1 Single frequency signal

Fig. 3.5a) shows two spectral lines for  $\cos()$ , at positions  $\phi$  and  $2\pi - \phi$ , reflecting the fact that the sampled value is the same for  $\phi$  or  $2\pi - \phi$  (and any  $2\pi$  multiple thereof). In Fig. 3.5b) the y-axis is scaled by  $10^{-14}$ , showing the odd part of  $e^{i\phi}$  is essentially zero. Fig. 3.6 demonstrates the single

spectral line that emerges from sampling a rotation in the complex plane. The lines amplitude on the y-axis is twice the value of each of the two lines in the  $\cos()$  spectrum, as expected from Eq. (3.20) and (3.21).

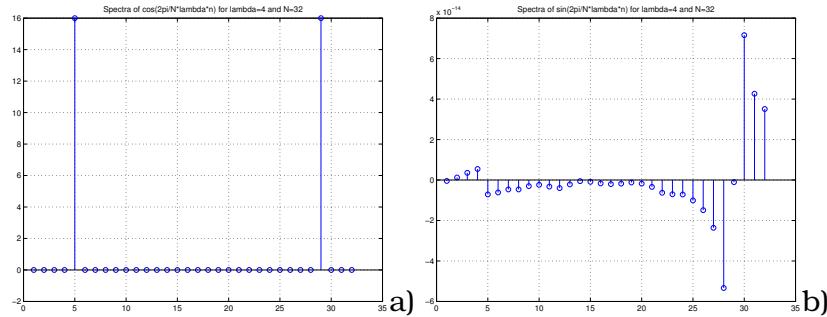


Figure 3.3: DFT of periodic signals a)  $\cos$  and b)  $\sin$

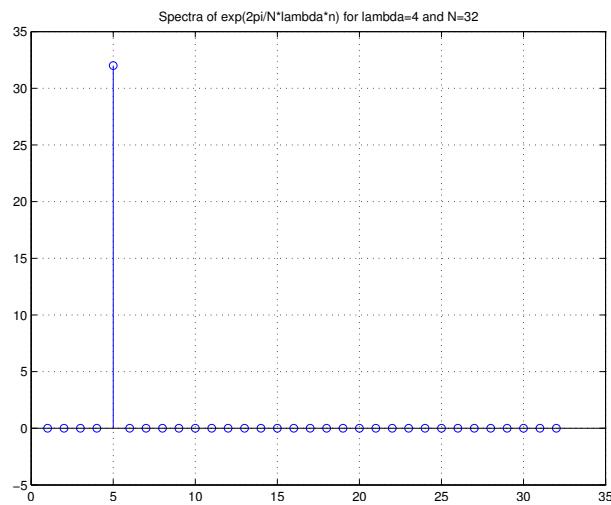


Figure 3.4: DFT of periodic signals a)  $\cos$  and b)  $\sin$

### 3.5.2 Multi-frequency signal

This signal is constructed as a superposition of four frequencies (one, two three and four times the base frequency). The plots demonstrate the real and imaginary parts of the signal and it's DFT spectrum for sin, cos and exp wave forms. For sin and cos wave, the imaginary part of the signal is zero, while for the exp wave form its real and imaginary components are, by definition, just the real parts of the cos and sin wave signal, respectively. Similar to the single frequency signal, the four spectral lines are found at  $\phi_i$  and  $2\pi - \phi_i$  for the real part of the spectrum of cos, and the imaginary part of the sin spectrum (see 3.5). As in the single spectral line case there are four lines in the real part of DFT of the exp wave form (see 3.6).

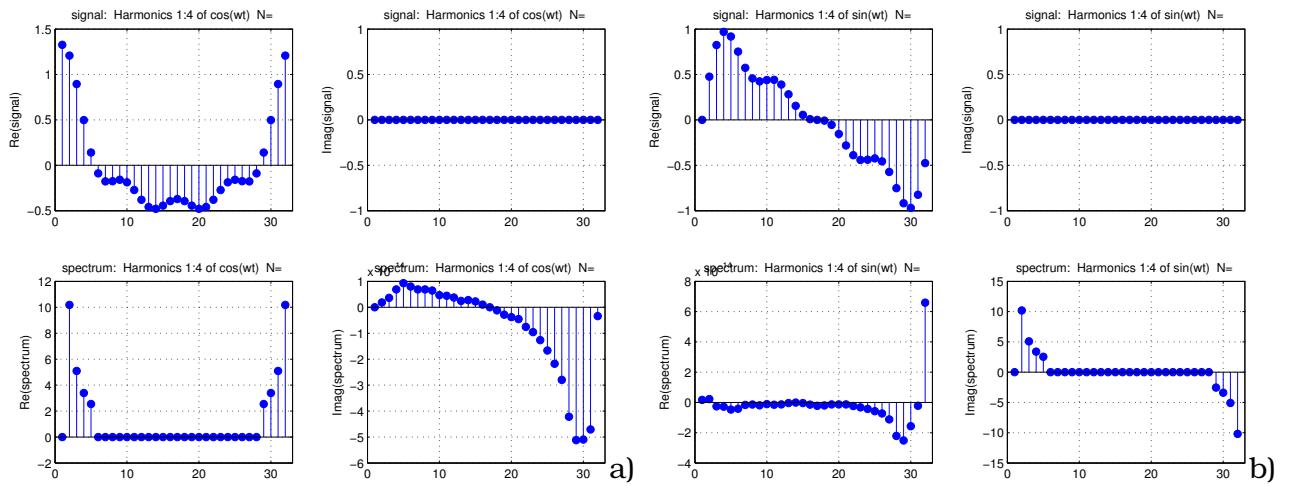


Figure 3.5: Harmonics: Real and imaginary parts of a) cos and b) sin

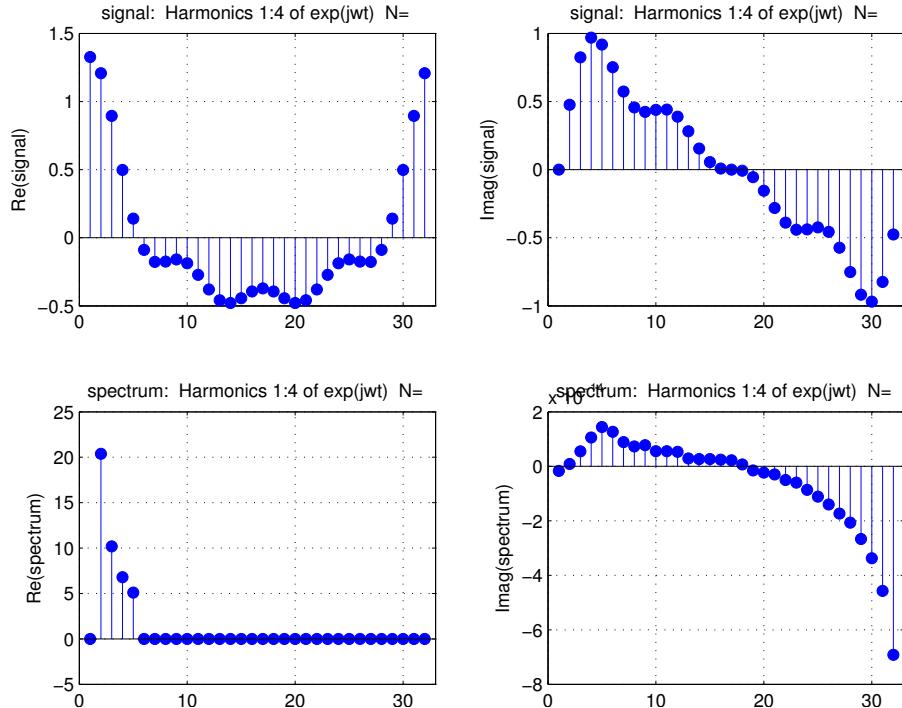


Figure 3.6: Harmonics: Real and imaginary parts of exp(jwt)

### 3.5.3 Leakage of non-periodic components

If the original signal contains frequencies that are not among the spectral lines of the DFT used, then those frequencies will 'leak' into the actual spectrum [Werner, 2012], [Doblinger, 2001]. The signal used to demonstrate this leakage of non DFT spectral frequencies into the DFT consists of seven frequencies from the base frequency to the first octave in steps of 1/6 base frequency. As before, the sin, cos (see Fig. 3.7) and exp (see Fig. 3.8) waves are considered. As above, the unit on the x-axis is the sample number. Thus one corresponds to the DC component, one to the base frequency, and two to the first octave.

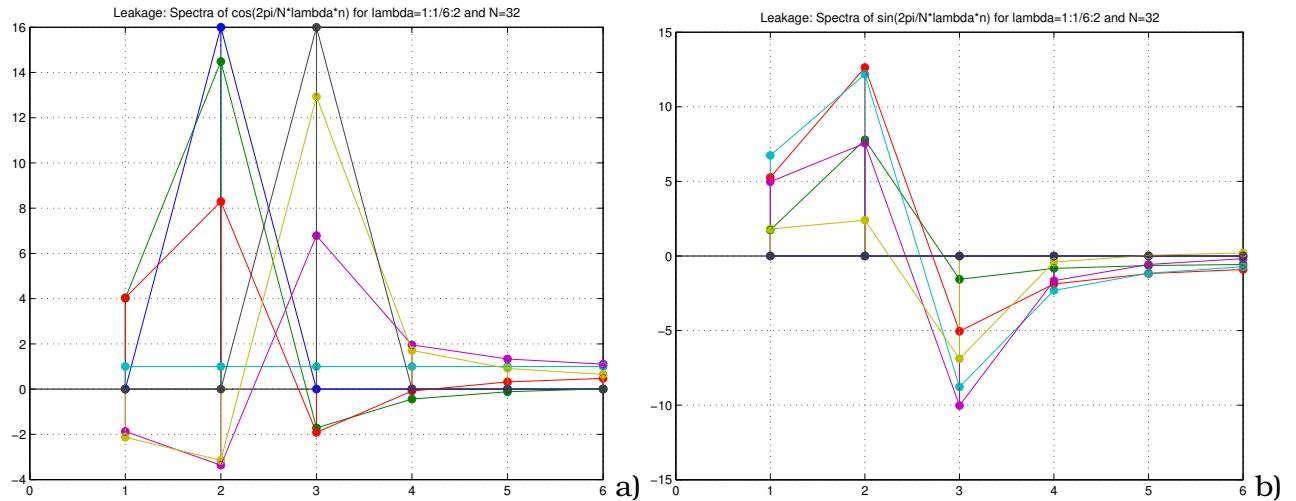


Figure 3.7: Leakage: DFT of non-periodic sin and cos signals

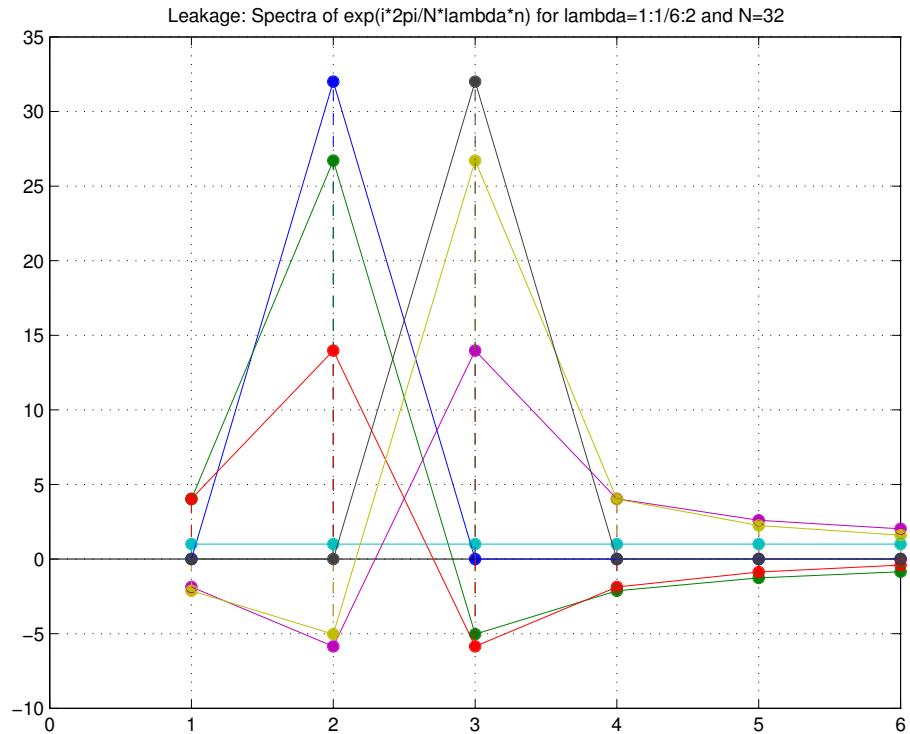


Figure 3.8: Leakage: DFT of non-periodic exp signals

## 4 Short Time Fourier Transform

Localization of spectral components in the time domain is achieved by restricting frequency analysis to sections defined by a window sliding across the signal (STFT). A rectangular window is a possible choice, though in many applications the window is constructed such that the signal is effectively damped on the its edges, allowing for the windowed signal to be extended to infinity.

### 4.1 Windowing to limited length signals

Given a window function  $\omega[n]$ , the windowed signal can be written as [Werner, 2012]

$$x_w[n] = x[n]\omega[n] \quad (4.1)$$

As in Eq. (3.2) and (3.3), the spectrum of the windowed signal is obtained from the convolution of the spectra of its multiplicative time domain components. The spectrum of the windowed signal becomes the periodic convolution

$$X_w(e^{j\Omega}) = \frac{1}{2\pi} X(e^{j\Omega}) * W(e^{j\Omega}) \quad (4.2)$$

The spectrum of a rectangular window of size  $N$  is given as [Werner, 2012]

$$W(e^{j\Omega}) = \sum_{n=0}^{N-1} \omega[n] e^{-jn\Omega} = e^{-j\Omega\frac{N}{2}} \frac{\sin(N\frac{\Omega}{2})}{\sin(\frac{\Omega}{2})} \quad (4.3)$$

and consists of the so called main and the (weaker) side lobes (see Fig. 4.1). The width of the main lobe decreases with increasing window size  $N$ . The main lobe smears out the spectral lines of the windowed signal, its width a measure for the spectral resolution at a given window size. The effect is related to limiting the signal in the time domain (that is, making it periodic), which translates into a discrete spectrum at frequencies given by

$$\Omega_k = k \frac{2\pi}{N} = k \Delta\Omega_{DFT} \quad k = 0 : N - 1 \quad (4.4)$$

Thus the spectral resolution of a STFT with a rectangular window is just

$$\Delta\Omega_{rectangle} = \frac{2\pi}{N} \quad (4.5)$$

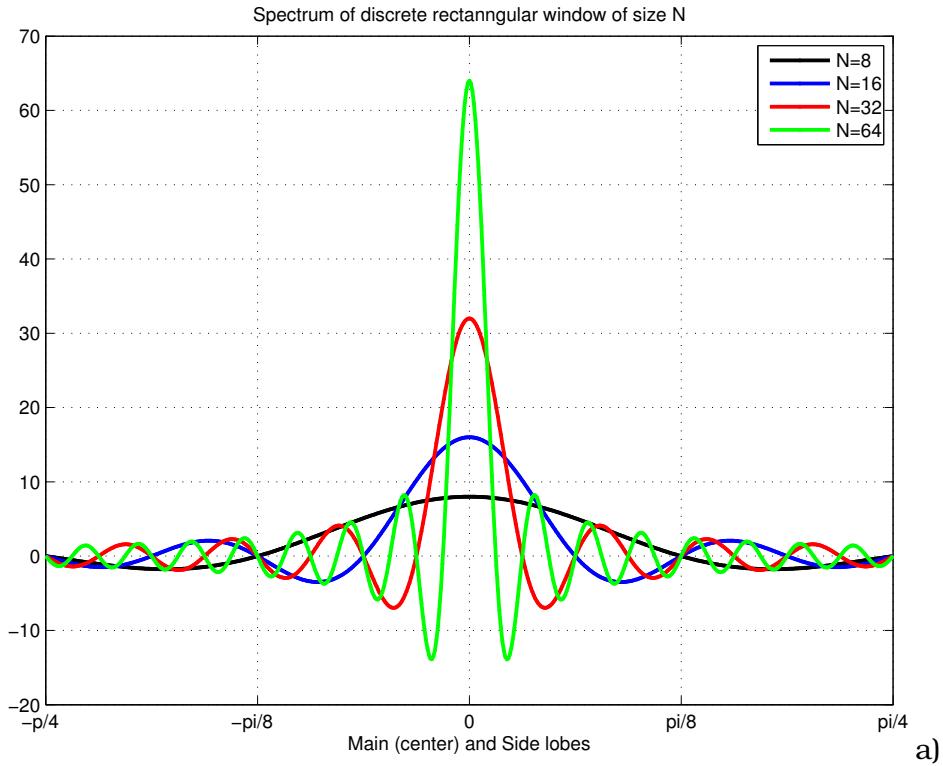


Figure 4.1: Spectrum of rectangular sampling window

## 4.2 Window types, leakage and zero padding

As outlined in Section 2.4, the number of lines in the frequency domain is identical to the number of discrete values in the time domain. Padding the windowed signal with zeros does not

enhance true spectral resolution though, but allows to plot the spectrum on a finer grid [Werner, 2012]. The amplitudes at the non-padded signal spectral lines remain the same. Zero padding can be used to identify frequency components that are not located on the spectrum grid. Such components 'leak' into the spectrum on spectral lines induced by the spectrum of the window function, with a total amplitude imposed by the conservation of energy (see Eq. (4.6), Parceval's law).

The behavior of leakage can be controlled to an extent by the choice of the window form. The effects of the window function are often characterized by the width of the main lobe, and the attenuation of the first side-lobe [Werner, 2012]. The width of the main lobe determines the spectral resolution, the attenuation of the first side lob is a measure for the amount of damping of nearby frequencies. The 'leakage' is defined as the proportion of energy contained in all side lobes. Spectral resolution, side lobe attenuation and leakage cannot be optimized for at the same time, with the choice of window type and parameters depending on desired properties.

Fig. 4.2 shows correlations of pairs of these properties for the rectangular (red), Bartlett (blue, triangular shape), Blackman (black) and Hamming (green, the latter both bell shaped) window functions for several values of  $N$ , the sample size. The properties shown are the sample width, the main lobe width, the side lobe attenuation and the total leakage. As outlined above, the width of the main lobe decreases with increasing sample width (top left). At a given sample size, those window functions that feature a small main lobe width exhibit a decreasing side lobe attenuation and increased total leakage (top right and bottom row of Fig. 4.2).

Figs. 4.3a) and 4.3b) compare properties of the Gauss and Chebychev window functions for a range their parameters (sigma and total leakage, respectively). The bottom right plots (side lobe attenuation vs. main lobe width) e.g. show that, if a main lobe width above 0.05 can be tolerated, the Chebychev window function has advantages in both side lobe attenuation and leakage.

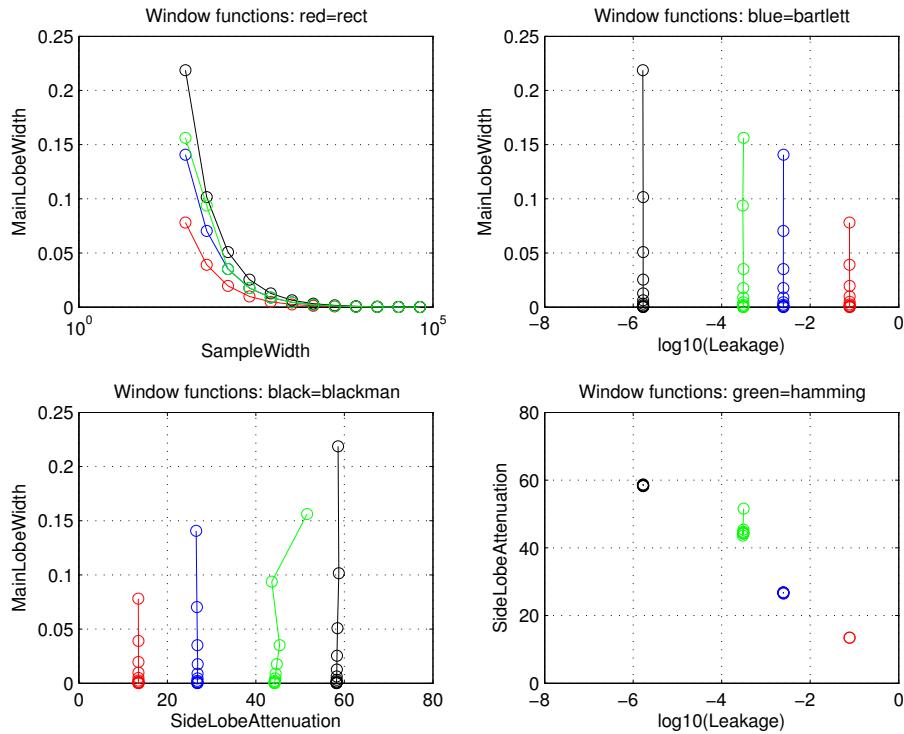


Figure 4.2: Sampling window functions: properties by window type and size

### 4.3 Gabor transform

A window narrow in the time domain transforms to a window wide in the frequency domain and the other way around. This is a consequence of Parceval's theorem states that the energy of a

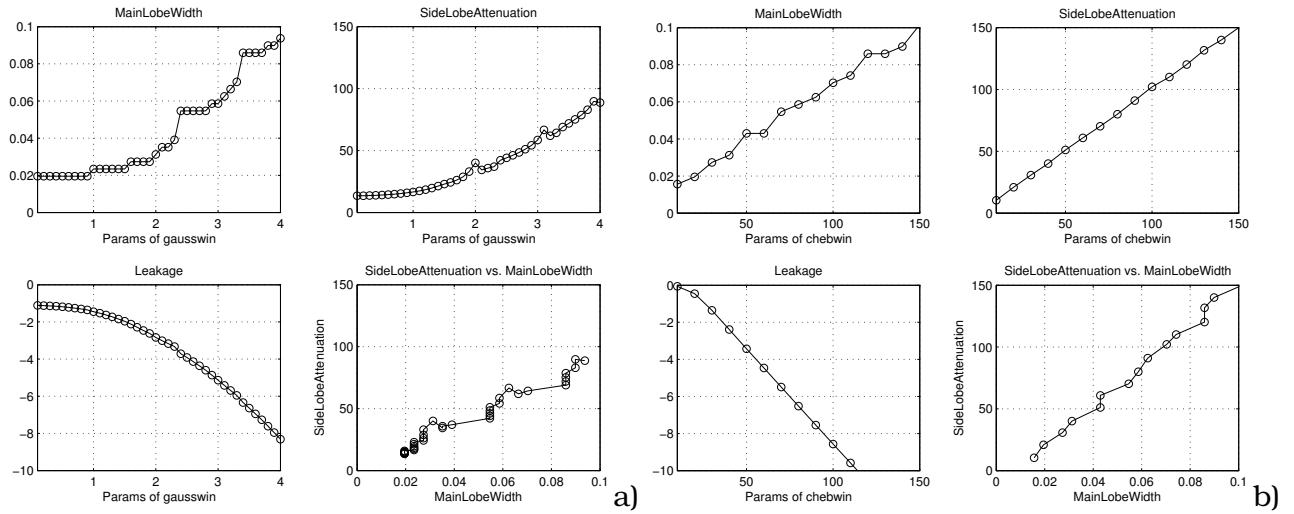


Figure 4.3: Sampling window functions: properties of a) Gauss and b) Chebychev by parameters

signal in the time and frequency domain are the same [Sonka et al., 2007]

$$\int |f(t)^2| = \int |F(s)^2| \quad (4.6)$$

The Gauss window function (the only function that retains its shape under the FT)

$$f(t) = \exp(-t^2) \quad (4.7)$$

minimizes the product of the window standard deviations in the time and frequency domain. This behavior is related to the Heisenberg uncertainty principle that was originally formulated as  $\sigma_f^2 \sigma_f^2 \geq \frac{1}{4}$  for dual entities like momentum and location of moving particles. The STFT with a Gaussian window is also referred to as Gabor transform (see Fig. 4.4 [Misiti et al., 2007]).

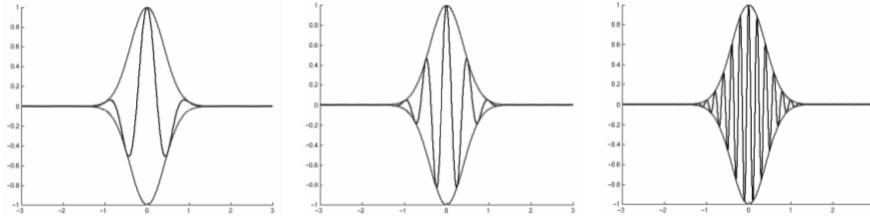


Figure 4.4: Gabor Transform

#### 4.4 Example: Wave audio spectrum

The real time spectrum of audio signals is analyzed by calculating the DFT in a window that slides across the signal stream. The interval between two DFT calculations determines the time resolution of the quasi-continuous spectrum. The window size depends on the low end, the sampling frequency on the high end of the desired frequency range. The example in Figs. 4.5 and 4.6 uses a mono signal sampled at 8kHz, length 4096 samples (~0.5 sec), the hamming window of size 128, no padding, and an interval between two DFT's of 64 samples. The dynamic spectrum is visualized in a waterfall diagram, and a spectrogram.

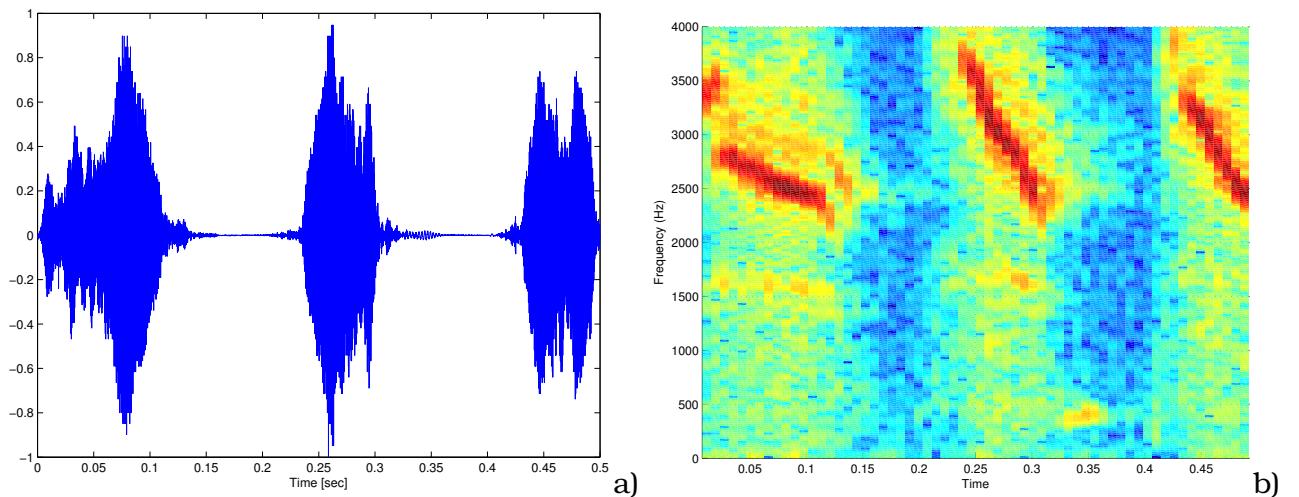


Figure 4.5: Wave audio: Signal a) and spectrogram b)

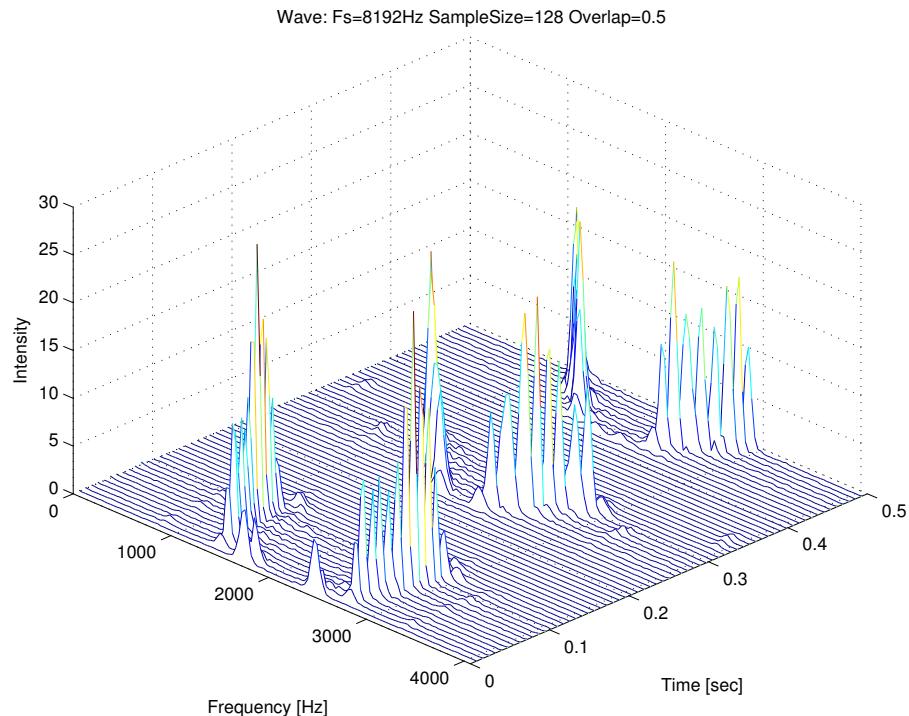


Figure 4.6: Wave audio: Waterfall diagram

## 5 Wavelets

The following sections demonstrate core functionality and important properties of wavelets, and show how they can be applied for smoothing/denoising and structure/contrast enhancement of one and two dimensional signals.

The sample 1D signal used is constructed from a cross-section through a RS image that contains several structures relevant to archaeological prospection. The 2D sample images show remnants of stone walls on grassland, imaged in the near infrared by the GeoEye1 satellite platform (co-owned by Google and used for the Google maps project).

## 5.1 Introduction

The necessity to resort from the infinite FT to the window based STFT to achieve time localization of the transform domain is a direct consequence of the infinite nature of the sin and cos basis functions used in the FT. The idea behind the wavelet transform is to construct basis functions such that these are localized (have 'compact support') by design. Localization in the time domain is achieved via translations and in the frequency domain via dilation of a so called 'mother wavelet'  $\psi$ . The effect of this shifting and scaling process is to produce wavelet coefficients which represent the correlation between the wavelet and a section of the signal at a given scale ('zone of influence'). Like this, any signal can be represented as an infinite series of wavelet basis functions. Wavelets often are constructed with an irregular shape which (as opposed to the smooth sin/cos components of the FT) opens up the possibility to encode the essence of complex signals into a few coefficients. Wavelets functions are different from zero only in a limited domain, oscillate (integrate to zero) and have finite energy (are square integrateable).

The dilation function of the discrete wavelet transform can be represented as a tree of low and high pass filters (see Fig. 5.1a) [Toufik and Mokhtar, 2012]), with each iteration transforming the low pass sub-band. The original signal is successively decomposed into components of lower resolution, while the high frequency components are not analyzed any further. In that sense, one wavelet spectral line is created for each dilation applied (see Fig. 5.1b) [Toufik and Mokhtar, 2012] and c) [Misiti et al., 2007]).

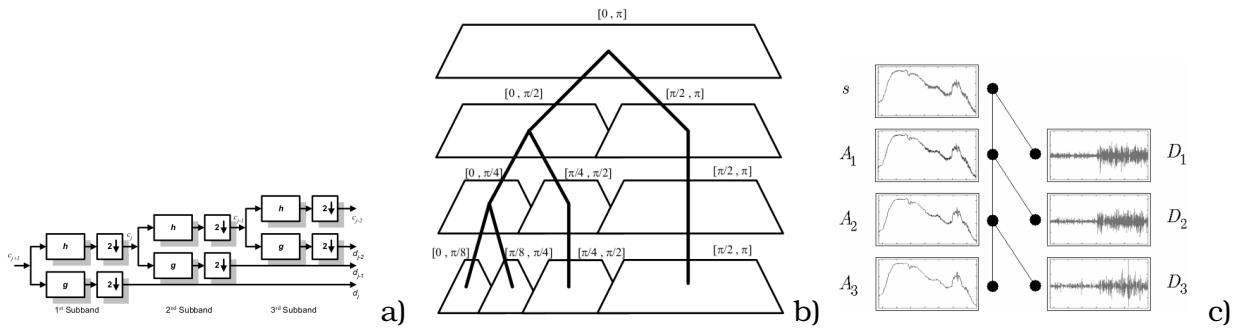


Figure 5.1: Wavelet decomposition tree by high and low pass filters

## 5.2 Wavelet construction

Wavelets are families of functions constructed by translation and dilation [Sifuzzaman et al., 2009].

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad (5.1)$$

The parameter  $a$  is the scaling parameter or scale, and it measures the degree of compression. The parameter  $b$  is the translation parameter which determines the time location of the wavelet. If  $|a| < 1$ , then the wavelet is a compressed version (smaller support in time-domain) of the mother wavelet and corresponds to higher frequencies. On the other hand, when  $|a| > 1$ , then  $\psi_{a,b}(t)$  has a larger time-width than  $\psi(t)$  and corresponds to lower frequencies. Thus, wavelets have time-widths adapted to their frequencies [Sifuzzaman et al., 2009].

During a certain time a wavelet oscillates like a wave and is then localized due to a damping. The oscillation of a wavelet is measured by the number of vanishing moments and its localization is evaluated by the interval where it takes values significantly different from zero.

Candidate wavelet function are at least required to satisfy to the admissibility condition [Misiti et al., 2007]

$$\psi \in L^1 \cap L^2, \quad \int_R \psi(t) dt = 0 \quad (5.2)$$

and are, among other properties, characterized by their number of vanishing moments

$$\int_R t^k \psi(t) dt = 0 \quad k = 0, \dots, m \quad (5.3)$$

The continuous wavelet transform of a function  $f$  then is defined as [Misiti et al., 2007]

$$C_f(a, b) = \int_R f(t) \overline{\psi_{a,b}(t)} dt \quad (5.4)$$

that is,  $f$  is described by its wavelet coeffs. With the support of  $\psi$  compact, the  $C_f$  describe  $f$  in a range around  $b$ , scaled by  $a$  in terms of  $\psi$ . Usually  $a$  and  $b$  are being limited to  $a = 2^j$  and  $b = k2^j = ka$  for  $j, k \in \mathbb{Z}$ .

Thus  $\psi$  takes the following form

$$\psi_{j,k}(x) = \frac{1}{\sqrt{2^j}} \psi \left( \frac{1}{\sqrt{2^j}} x - k \right) \quad (5.5)$$

and the wavelet transform coeffs are given as

$$\alpha_{j,k} = \int_R s(t) \psi_{j,k}(t) dt \quad (5.6)$$

Under suitable conditions on  $\psi$ , these coeffs are sufficient to perfectly reconstruct the original signal [Misiti et al., 2007]

$$s(t) = \sum_j \sum_k \alpha_{j,k} \psi_{j,k}(t) \quad (5.7)$$

Several important wavelets functions  $\psi$  have been constructed such that the family  $\psi_{j,k}$  forms on orthonormal basis of  $L^2$ . As the  $\psi_{j,k}$  are generated by dilation, these are closely related to the concept of multi-resolution analysis (MRA).

### 5.3 Multiresolution Analysis

MRA is designed to give good time resolution at high frequencies and good frequency resolution at low frequencies. As in the STFT, translation is used for localization in the time domain, but spectral components are calculated from varying width windows.

In the following some properties of a MRA are given. The indices used follow wavelet analysis conventions, that is,  $V_0$  denotes the space of the original signal, and  $V_j$ ,  $j > 0$ , the scaled subspaces.

A MRA of  $L^2$  consists of a sequence of nested subspaces  $V_j$ ,  $j > 0$ , that constitute scaled approximations to their predecessors. Associated with the reference subspace  $V_0$  is a scaling function  $\varphi$  such that the set  $\varphi_k(x) = \{\varphi(x - k)\}$  forms an orthonormal basis of  $V_0$ . The subspaces are similar in the sense that if  $f(x) \in V_j$ , then  $f(2x) \in V_{j+1}$  and vice versa. For any subspace  $V_j$  the difference to level  $V_{j-1}$  is its orthogonal complement  $W_j$  such that  $V_{j-1}$  is decomposed into an approximation level  $V_j$  and a detail level  $W_j$  [Lee, 2008].

$$V_{j-1} = V_j \oplus W_j \quad (5.8)$$

Consequently, any subspace  $V_j$  can be constructed from its detail levels and the approximation corresponding to the coarsest detail level used [Misiti et al., 2007]

$$V_k = V_j \oplus W_j \oplus \dots \oplus W_{k+1} \quad \forall k < j \quad (5.9)$$

The integer translates of  $\varphi$ ,  $\varphi_{j,k}$ , generate  $V_j$  ('low pass', approximation) while the translates of  $\psi$ ,  $\psi_{j,k}$ , generate  $W_j$  ('high pass', details) [Misiti et al., 2007]. Since  $L^2$  in this setup is spanned by  $W_j$ , any signal can be written as the sum of its projections  $\alpha_{n,k}$  onto the details spaces spanned by  $\psi_{j,k}$ . Like this, a signal is analyzed in terms of hierarchical approximations by a scale based set of orthogonal functions. For an arbitrary level  $j$ , a signal  $s(t)$  can be written as [Misiti et al., 2007]

$$s(t) = \sum_k \beta_{j,k} \varphi_{j,k}(t) + \sum_{n=j}^{+\infty} \sum_k \alpha_{n,k} \psi_{n,k}(t) \quad \forall j \quad (5.10)$$

with  $\alpha_{n,k}$ ,  $\beta_{n,k}$  the detail and approx wavelet expansion series, respectively, and approximation and detail signals at level  $j$

$$A_j(t) = \sum_k \beta_{j,k} \varphi_{j,k}(t) \quad D_j(t) = \sum_k \alpha_{j,k} \psi_{j,k}(t) \quad (5.11)$$

The multiscale approximation and detail wavelet analysis is implemented as low and high pass decomposition and reconstruction filters  $h_k$  and  $g_k$

$$\varphi_{j+1,0} = \sum_k h_k \varphi_{j,k} \quad \psi_{j+1,0} = \sum_k g_k \varphi_{j,k} \quad (5.12)$$

Thus the entities involved are the original signal  $s$ , the approximation  $\beta_{n,k}$  and detail coeffs  $\alpha_{n,k}$  and , and the approximation and detail signals A and D. The former are in units of decimated/dyadic time, while the latter are in original signal time. Transition from one level to the next is accomplished with the low and high pass filters h and g for the wavelet defined by  $\varphi$  and  $\psi$ .

## 5.4 Wavelet Properties

Several families of wavelets with different characteristics have been constructed and the best selection depends on the requirements of the application. Several properties are to be considered, including the number of vanishing moments and the size of the support.

### 5.4.1 Vanishing moments

"The number of vanishing moments determines what and what not a wavelet see's. Usually a wavelet  $\psi$  has  $p$  vanishing moments if [Misiti et al., 2007]

$$\int_{-\infty}^{+\infty} t^k \psi(t) dt = 0 \quad \text{for } 0 \leq k < p \quad (5.13)$$

This means that it is orthogonal to any polynomial of degree  $p - 1$ . Therefore, a wavelet with two vanishing moments cannot see linear functions; a wavelet with three vanishing moments will be blind to both linear and quadratic functions; and so on. If the signal  $f$  is regular and  $C^k$  (which means  $f$  is  $p$  times continuously differentiable function, when  $k < p$ ), then the wavelet can generate small coefficients at fine scales  $2^j$ . Too high vanishing moment may miss the useful information in signal, and leave more useless information such as noise. The proper number of vanishing moments is thus important in optimal wavelet selection." [Luo and Zhang, 2012]

### 5.4.2 Size of support

"The size of support is the length of interval in which the wavelet values are non-zero. A signal anomaly will be visible in the coefficients of all wavelets that cover the event. If an orthogonal wavelet  $\psi$  has  $p$  vanishing moments, its support size must be at least  $2p - 1$ . The Daubechies wavelets are optimal to have minimum size of support for a given number of vanishing moments. When choosing a wavelet, we have to face a trade-off between number of vanishing moments and size of support. This is dependent on the regularity of signal  $f$ . For instance, the irregular wavelet db2 induces irregular errors in wavelet thresholding of regular signal processing. But it achieves better estimation when applied to estimate transient signal in power system which are often composed by pulses and heavy noises." [Luo and Zhang, 2012]

### 5.4.3 Orthonormal basis

This family of wavelets features orthogonality between the scaling and the wavelet function. The same is true for the wavelet and the scaling coefficients. The corresponding filters are called Quadrature Mirror Filters (QMF) [Toufik and Mokhtar, 2012].

$$h(n) = (-1)^n g(1 - n) \quad (5.14)$$

### 5.4.4 Biorthogonal basis

Here the analysis filter is orthogonal to its synthesis dual filter, hence the name biorthogonal. At the expense of the direct analysis/synthesis orthogonality (and thus violation of the energy partitioning property, Perseval's equality) the bi-orthogonal basis features the linear phase property [Toufik and Mokhtar, 2012].

$$\tilde{h}(n) = (-1)^n g(1 - n) \quad (5.15)$$

## 5.5 Example: 1D RS signal

The following examples uses the 1D sample RS signal to investigate wavelet properties that are relevant to the subsequent Sections on wavelet based signal and image enhancement.

Figs. 5.2 show the haar (db1) wavelet level 5 decomposition detail and approximation coefficients at decimated (but stretched) scale.

Fig. 5.3a) shows the detail coefs of the level 5 decomposition, Fig 5.3b) the approximation coefs at decomposition levels 1-5, in original scale. Both figures suggest that most signal characteristics are found in detail levels two to four, with some contributions from level one and five.

Fig. 5.4a) demonstrates how an increasing number of vanishing moments (wavelets db1 to db5) influences the reconstruction from all detail levels and the remaining difference to the original signal (that is, the reconstruction from the approximation coefs only). As expected, the remaining low pass component becomes smoother with an increasing number of vanishing moments.

Fig. 5.4b) compares five different wavelets with three vanishing moments, using the same reconstruction as in 5.4a). The db3 and sym3 wavelets look almost identical (both support  $2^*N-1$ ), similar for coif3 (2\*3 vanishing moments,  $6^*N-1$  support) and bior3.3 (both close to db4), and rbio3.3 an exotic outlier.

Fig. 5.5 shows the signal reconstructed from an increasing number of detail coefs, starting from the finest and progressively including all levels up to decomposition depth. The figure confirms that reconstruction from detail levels two to five results in a good approximation of the original.

Fig. 5.6 summarizes the results of reconstruction from a range of detail levels and compares to the original signal using db1 in a) and db4 in b). Most of the characteristics of the 1D RS

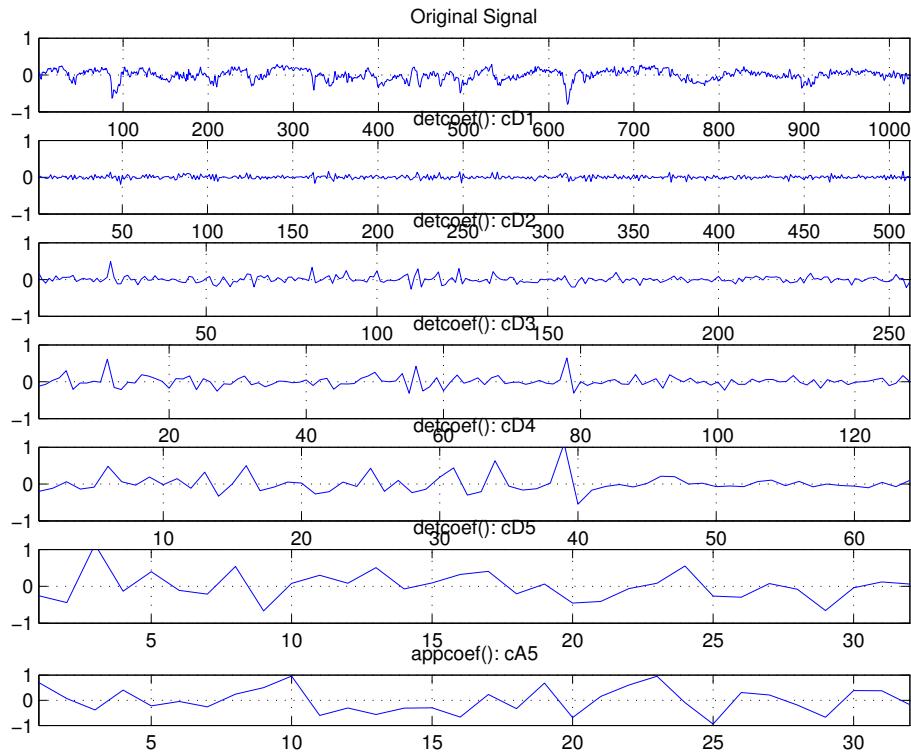


Figure 5.2: wDecRec(): Haar, N=5: Decomposition detail and approximation coefs in scaled coords

signal sample can be reconstructed from detail levels two to four or five, with no obvious signs of usable information in the remaining coef spaces. With db4, the non-local signal variance of the original is clearly visible in the remaining approximation level.

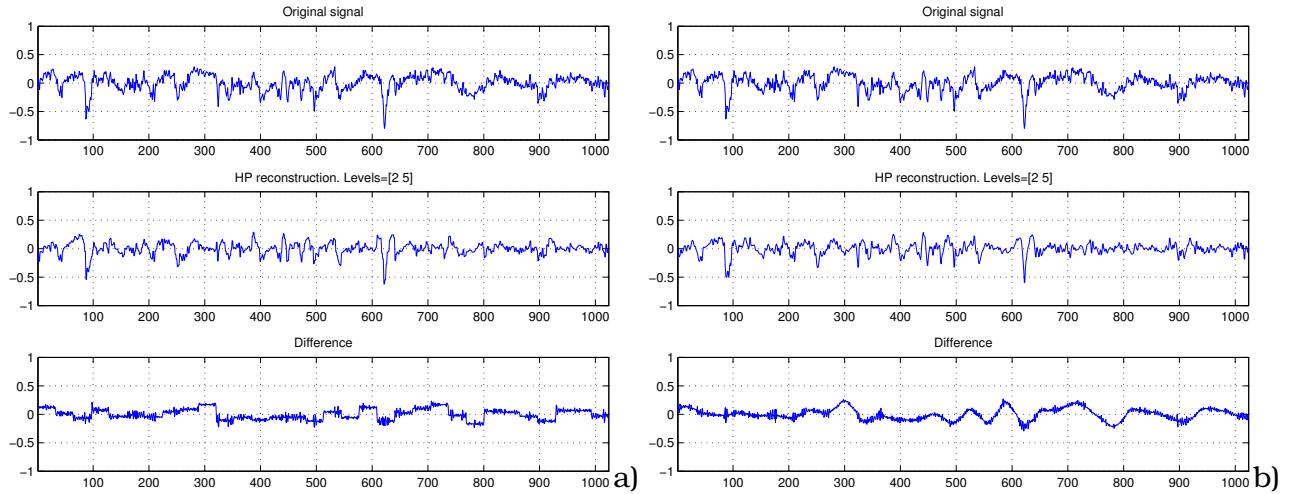


Figure 5.6: wDecRec(): HP reconstruction levels=2-5: a) Haar b) db4

## 5.6 Application: Denoising

The wavelet based denoising procedure is based on the model that an original signal has been disturbed with noise of specific characteristics. Identifying noise amounts to estimating a threshold that classifies detail coefficients into noise and signal components.

Denoising proceeds in three steps [Luo and Zhang, 2012]:

1. Decomposition: The signal is decomposed into N levels by a suitable wavelet.

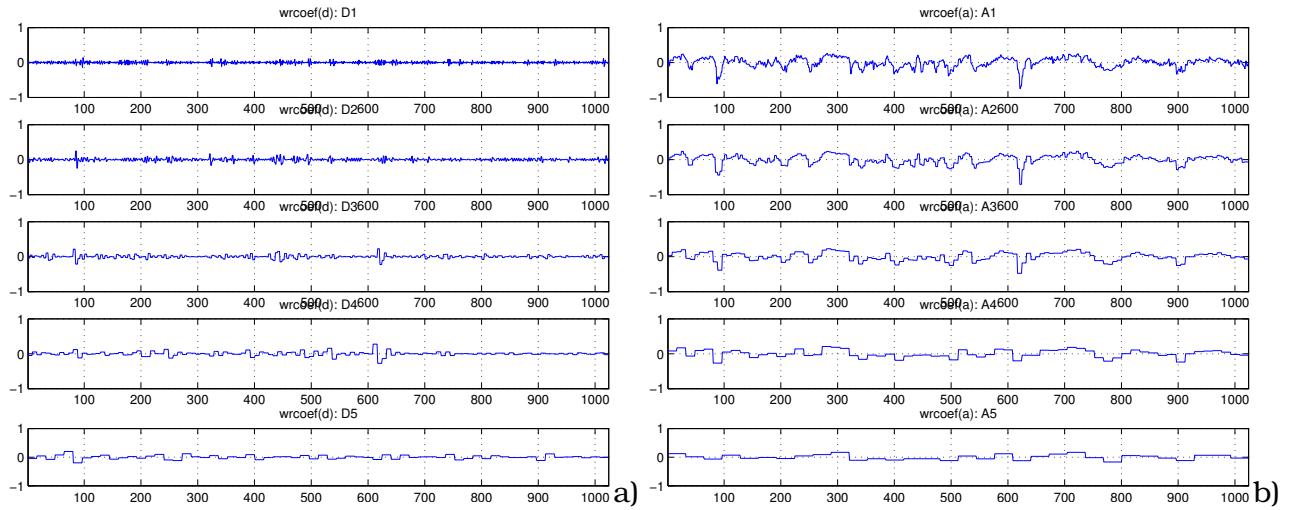


Figure 5.3: wDecRec() Haar, N=5: a) Decomposition b) HP components c) LP components

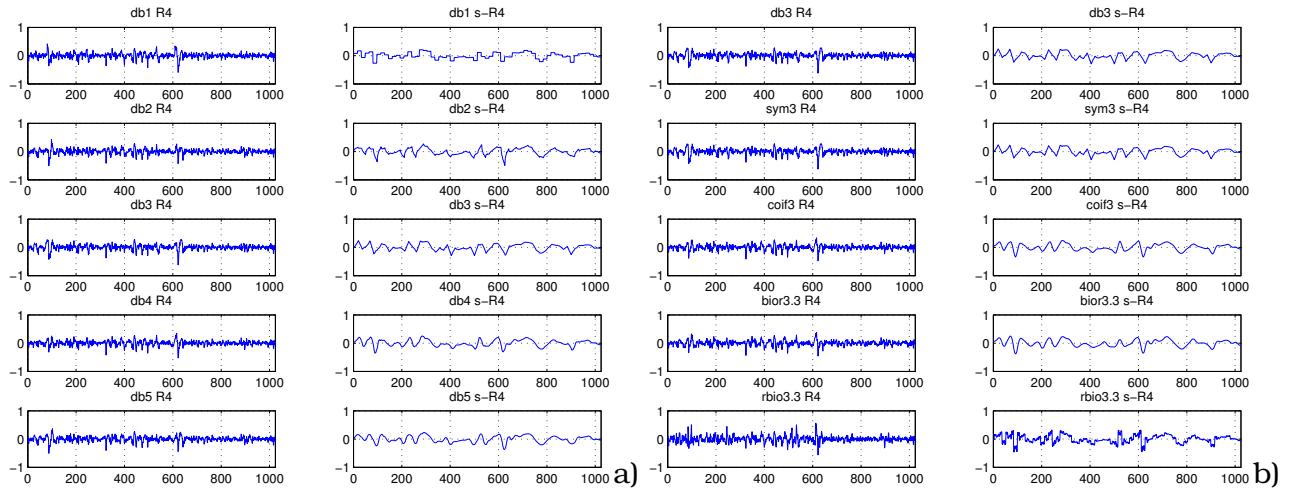


Figure 5.4: RS-1D-Rec(d1-d4), N=4 a) db1-db5 b) db3,...,rbio3.3

2. Thresholding: For each level a threshold applied to the detail coefficients. To minimize influences of the signal itself, the threshold often is based on a robust estimator of noise variance. Then noise is suppressed by setting to zero decomposition coefs below the thresh-old.
3. Reconstruction: The signal is reconstructed from the original approximation coefficients at level N and the modified detail coefficients.

For many applications noise is modeled as Gaussian distributed with white spectrum  $W[n] = \sigma e[n]$ .

$$X[n] = f[n] + W[n] \quad (5.16)$$

The undisturbed signal  $f[n]$  is estimated as  $F[n] = D(X[n])$ , where D is constructed to minimize the error  $f - F$ , often using the least squares distance for Gaussian error models. The risk of the estimation is calculated as [Luo and Zhang, 2012]

$$r(D, f) = E\{||f - F||^2\} \quad (5.17)$$

Several estimators exist, among these the 'minimax', 'universial' and 'Stein's unbiased risk' (SURE). Coefficients smaller than the threshold are regarded as noise, and are usually set zero.

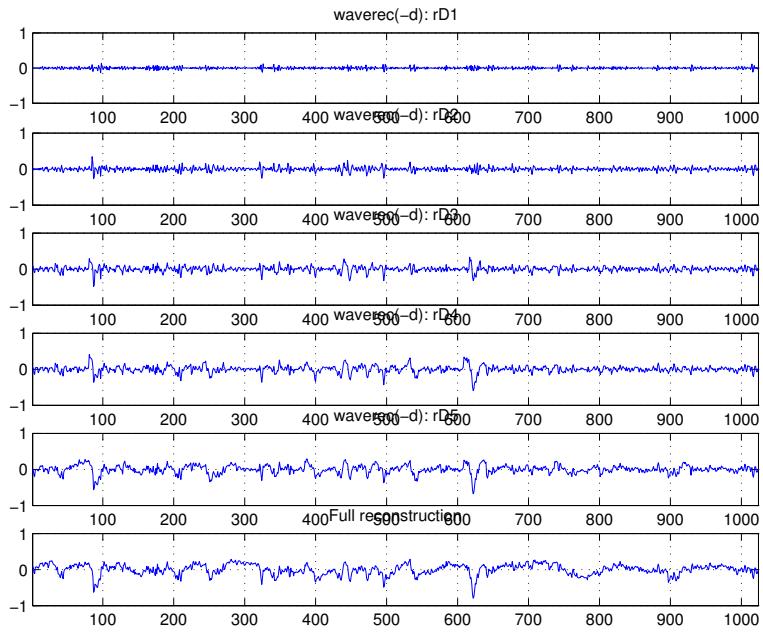


Figure 5.5: wDecRec(): Haar, N=5: Reconstruction from increasing number of HP components

Values above the threshold are either retained unscaled ('hard thresholding') or rescaled ('soft thresholding'), see Fig. 5.7 [Luo and Zhang, 2012]. However, because of the error induced by hard-thresholding, oscillations or small ripples are created near irregular points.

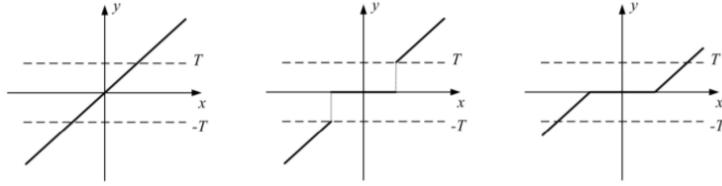


Figure 5.7: Denoising Threshold: signal, hard and soft threshold

Across the multi resolution analysis levels, a constant threshold is used for white noise, often estimated from the finest scale. This approach reduces the influence of signal and edge effects in deeper decomposition levels [Luo and Zhang, 2012]. Level dependent thresholds are used for more general noise patterns..

### 5.6.1 1D RS signal

Matlab offers the wden function for automatic denoising, and wdencmp for customized denoising and compression. In automatic mode the threshold levels of the multi-scale detail coefficients are based on a global strategy (same or different threshold for all scales) and a risk estimator (e.g. universal or SURE). Detail coefficients of a specific wavelet and decomposition depth are updated using thresholding strategy soft or hard. The function wdencmp together with the threshold manager wthrmngr offers more flexible options for the estimation of the threshold levels. Custom algorithms can be implemented by resorting to direct filtering of the multi-scale detail coefficients.

Fig. 5.8a) shows the distribution of the level 3 detail coefs for a haar level five decomposition after soft (left col) and hard thresholding (right col) for different estimators (top to bottom) and a single threshold level for all detail levels. The amount of denoising in this case is much lower for the rigsure and heursure estimators as compared to squarelog and minimax variants. The effects are evident in Fig. 5.8b) where the soft thresholded squarelog and minimax estimators lead to

substantial noise reduction. The least denoising effects are seen for the heard thresholding of the rigsure and heursure methods.

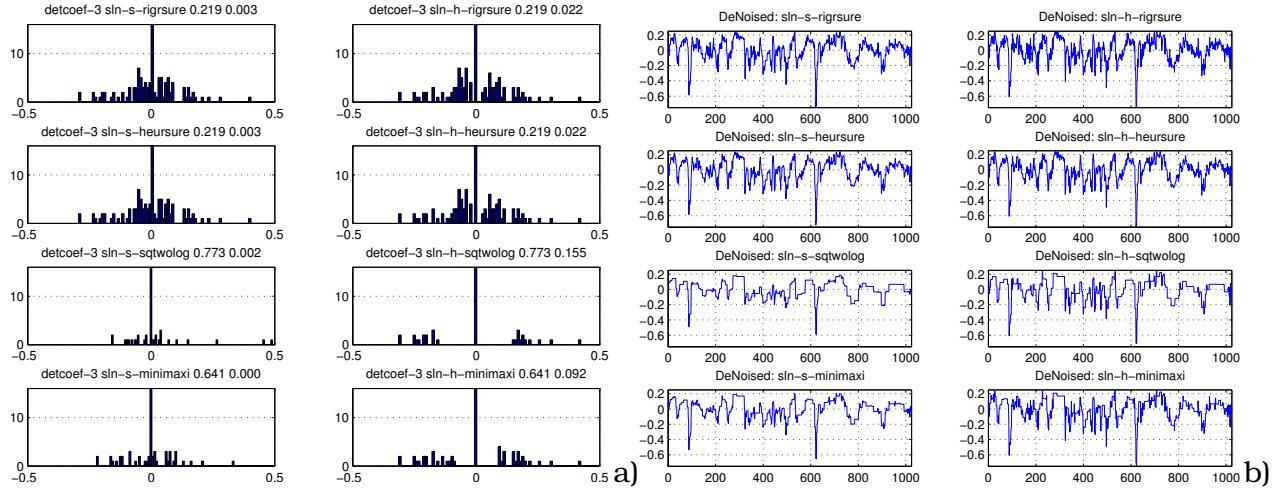


Figure 5.8: wDenoiseCompare0: Threshold method survey (haar wavelet)

## 6 Transformation of Images

Application of the DFT and the DWT to 2D signals runs the 1D variant of the transform independently across the signals in rows and cols. The 2D DFT first takes the DFT of all rows, and then calculates the col wise DFT of the result. The 2D DWT first applies the low and high pass filters to rows, and then col wise to the such obtained intermediate result.

### 6.1 2D Fourier Transform

Employing the super position principle, a digital image can be written as a linear combination of pulses distributed across the image plan, weighted by the value of the image function  $f(x, y)$  [Sonka et al., 2007]

$$f(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(a, b) \delta(x - a, y - b) da db \quad (6.1)$$

This expression can be regarded as a convolution of  $f(a, b)$  with the Dirac pulse. More generally, the convolution of a signal  $f$  with a signal  $h$  is given as

$$(f * h)(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(a, b) h(x - a, y - b) da db = \quad (6.2)$$

$$= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x - a, y - b) h(a, b) da db = (h * f)(x, y) \quad (6.3)$$

For images, that is, 2D discrete systems, the convolution with mask  $h$  (the 'convolution kernel') is written as a sum over its elements [Sonka et al., 2007]

$$f(i, j) = \sum_{m, n \in O} h(i - m, j - n) g(m, n) \quad (6.4)$$

where  $O$  is the neighborhood of pixels around  $h(i, j)$  covering  $g$ .

The 2D DFT is given as [Sonka et al., 2007]

$$F(u, v) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \exp\left(-2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)\right) = \quad (6.5)$$

$$= \frac{1}{M} \sum_{m=0}^{M-1} \left[ \frac{1}{N} \sum_{n=0}^{N-1} \exp\left(-\frac{2\pi i nv}{N}\right) f(m, n) \right] \exp\left(-\frac{2\pi i mu}{M}\right)$$

$$f(m, n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) \exp\left(2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)\right) \quad (6.6)$$

This shows that the 2D DFT can be calculated in a two step procedure that first calculates the 1D DFT of rows and then processes the result over cols.

Given a typical intensity resolution of 8 bits in the space domain, the same number of intensities are present in the frequency domain. As the values usually span several orders of magnitude, the spectrum is often visualized as  $\log(|F(u, v)|)$ . If  $f(x, y)$  is real valued (which is always the case for standard images), then  $F(u, v) = F^*(-u, -v)$ . Thus only the first quadrant can be used without loss of generality. The (periodic) four frequency quadrants are often aligned such that the origin being zero is at the center.

## 6.2 DFT of sample RS image

Figures 6.1, 6.3 and 6.2 demonstrate the FFT components of a sample image, the waves corresponding to five consecutive pixels, and the reconstruction from a slightly manipulated FFT spectrum. The quadrants of the spectra have been rearranged for DC in the center. Five pixels, that is, component waves, are selected from the central area of the power of the spectrum, and their representation in space domain reconstructed (see Fig. 6.3).

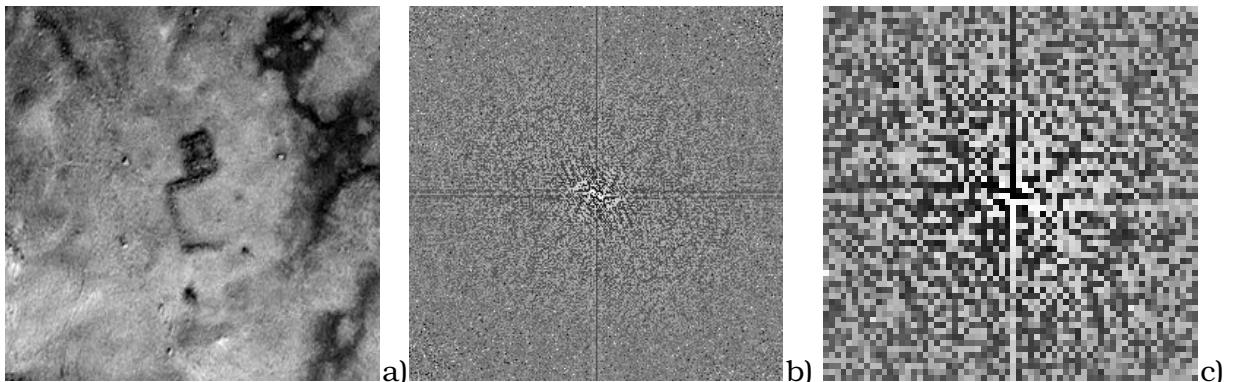


Figure 6.1: a) Original b)Real(FFT2) c) Zoom of imag(FFT2)

These five pixels are set to zero in the FFT (see Fig. 6.2b) and the original reconstructed (see Fig. 6.2c). The difference between original and reconstruction from the manipulated spectrum is shown in 6.3f.

## 6.3 2D Wavelets

Wavelet functions in two dimensions are constructed from one dimensional scaling  $\varphi(x)$  and wavelet  $\psi(x)$  functions that are directionally combined such that horizontal, vertical and diagonal components are being extracted [Lee, 2008].

$$\varphi(x, y) = \varphi(x)\varphi(y) \quad (6.7)$$

$$\psi^H(x, y) = \psi(x)\varphi(y) \quad (6.8)$$

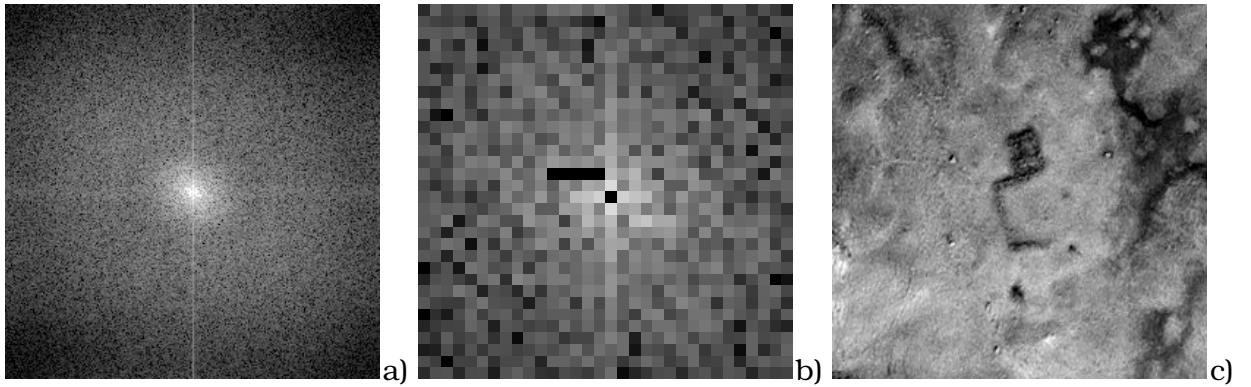


Figure 6.2: a) Power(FFT2) b) Zoom of Power(FFT2) c) IFFT with five pixels set to zero

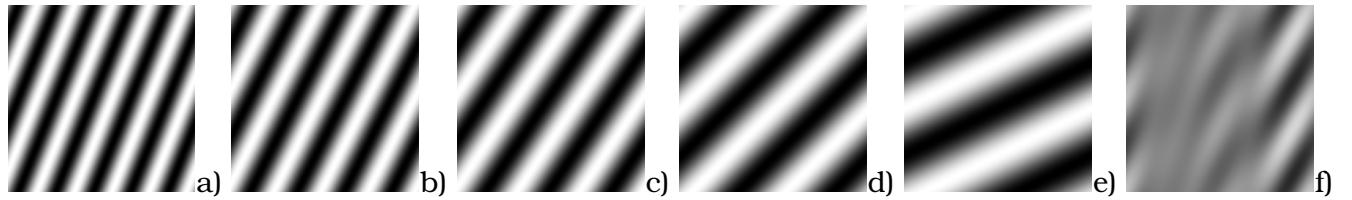


Figure 6.3: a)-e) Waves corresponding to five selected pixels of FFT2 f) Scaled superposition of a)-e)

$$\psi^V(x, y) = \varphi(x)\psi(y) \quad (6.9)$$

$$\psi^D(x, y) = \psi(x)\psi(y) \quad (6.10)$$

The implementation of the DWT is usually based on a high and a low pass filter that are being applied to the rows and cols in turn, producing the four transforms given in Eqs. (6.7) to (6.10). Scaling is inferred by dyadic decimation on rows and cols. Figs. 6.4 [Toufik and Mokhtar, 2012] and 6.5 [Lee, 2008] show one stage of the 2D filter banks and the size and content of two subsequent decomposition steps, respectively.

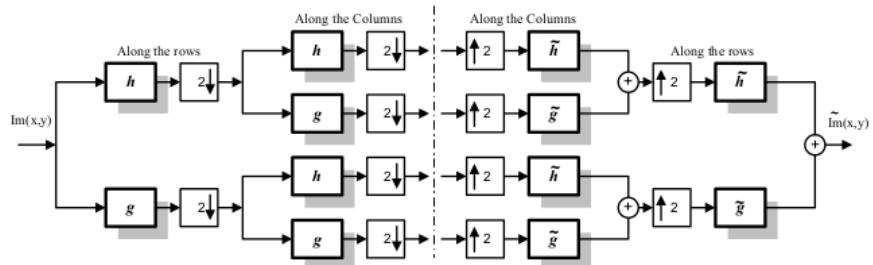


Figure 6.4: 2D wavelet filter banks

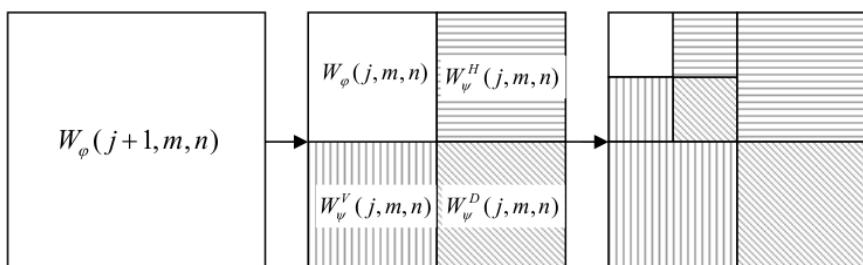


Figure 6.5: 2D decimated wavelet decomposition

## 6.4 DWT of sample RS image

This section extends on the 1D examples given in Sections 5.4 and 5.5. The decomposition and reconstruction products of several different wavelets are calculated and coefficients suitably rescaled for visualization (by setting the BW points relative to the mad() of the intensity distribution). The plots are produced by methods implemented in the 'dspCers' package (for details see Section 7).

Fig. 6.6 shows the decomposition products of a five level decomposition with the 'haar=db1' wavelet. Components from top to bottom are the approximation, horizontal, vertical and diagonal, from left to right the decomposition levels. A first inspection indicates that relevant detail is extracted in the first four detail levels.

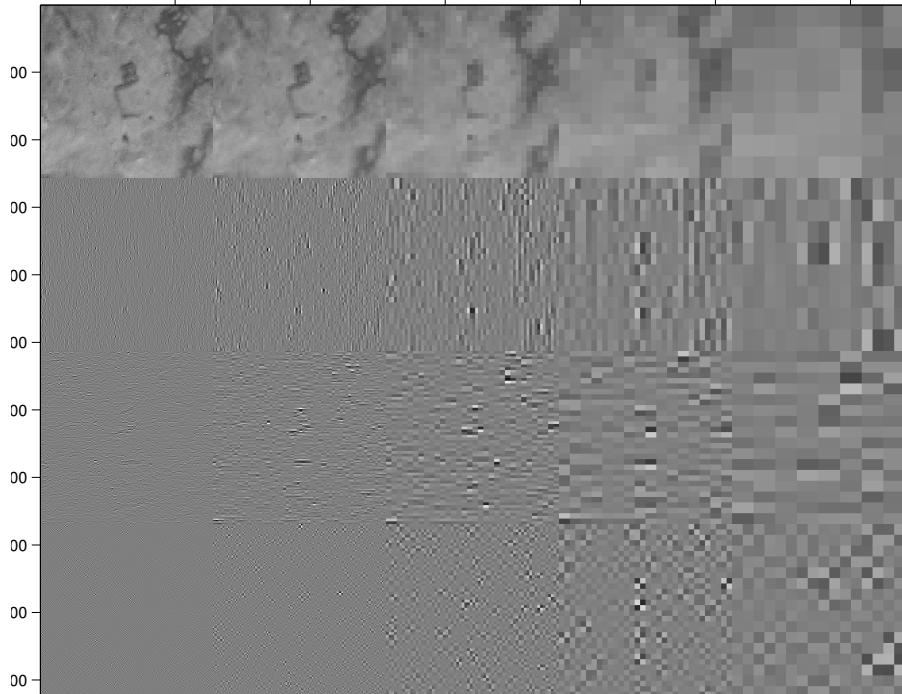


Figure 6.6: wimExplore: db1, N=5: A,H,V,D coefs

Fig. 6.7 compares the approximation coefs of the db1 to db7 (in rows) at levels one to five (in cols). For each detail level, all three components (H, V, D) are used. Differences become more prominent at deeper decomposition levels. Across levels, most detail is lost for db1, while, at first glance, not a lot is to be gained with more than three to five vanishing points (db3 to db5).

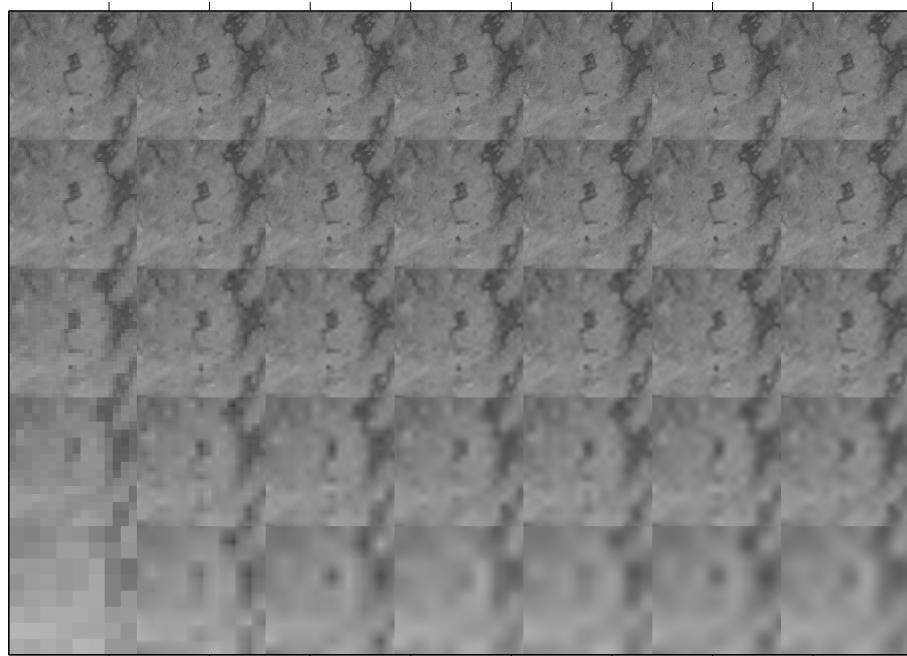


Figure 6.7: wimCompare: db1-db7, N=5: Approximation Levels 1-5

Fig. 6.8 shows the reconstructions from single detail level for levels 1 to 5 and for db1 to db7. Compliant with previous findings, levels two to four show most relevant information, and differences due to number of vanishing points rather subtle above values of maybe four. There are some hints of slight advantages for wavelets with an even number of vanishing points (e.g. db4 on levels 3 and 4).

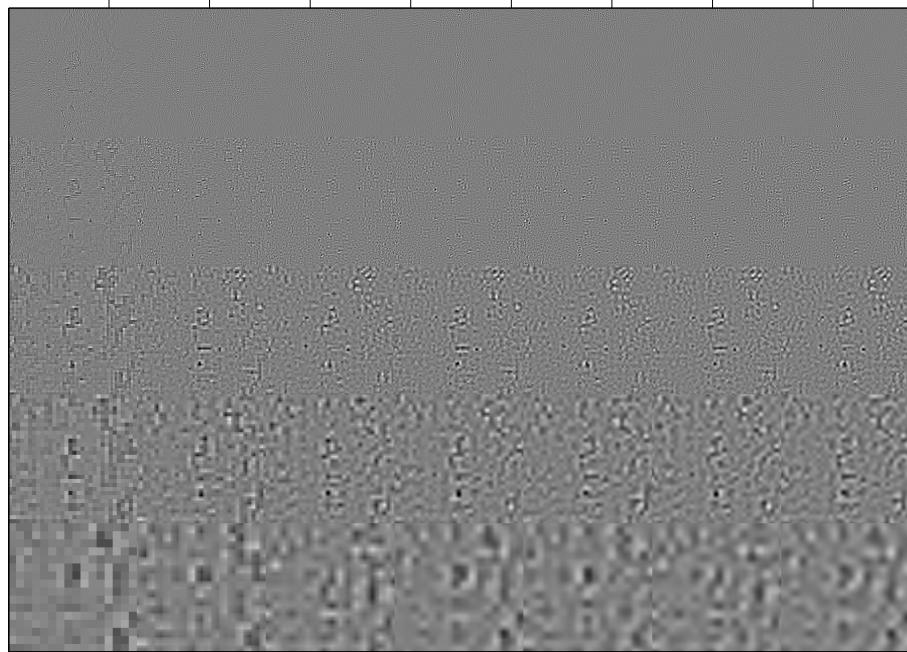


Figure 6.8: wimExplore: db1-db7, N=5: R(i,j)=waverec2(HVD(i=1-5),db1-db7)

Fig. 6.9 demonstrates reconstruction from selected detail levels of the 'haar' wavelet. All possible reconstructions involving groups of neighboring detail levels ( $\{1\}$ , ...,  $\{5\}$ ,  $\{1,2\}, \dots, \{4,5\}$ ,  $\{1,2,3\}, \dots, \{3,4,5\}$ , ...,  $\{1,2,3,4,5\}$ ) are calculated and plotted at coords (i=first level, j=last level). As expected the diagonal elements show that detail levels two to four contain most of the relevant information.

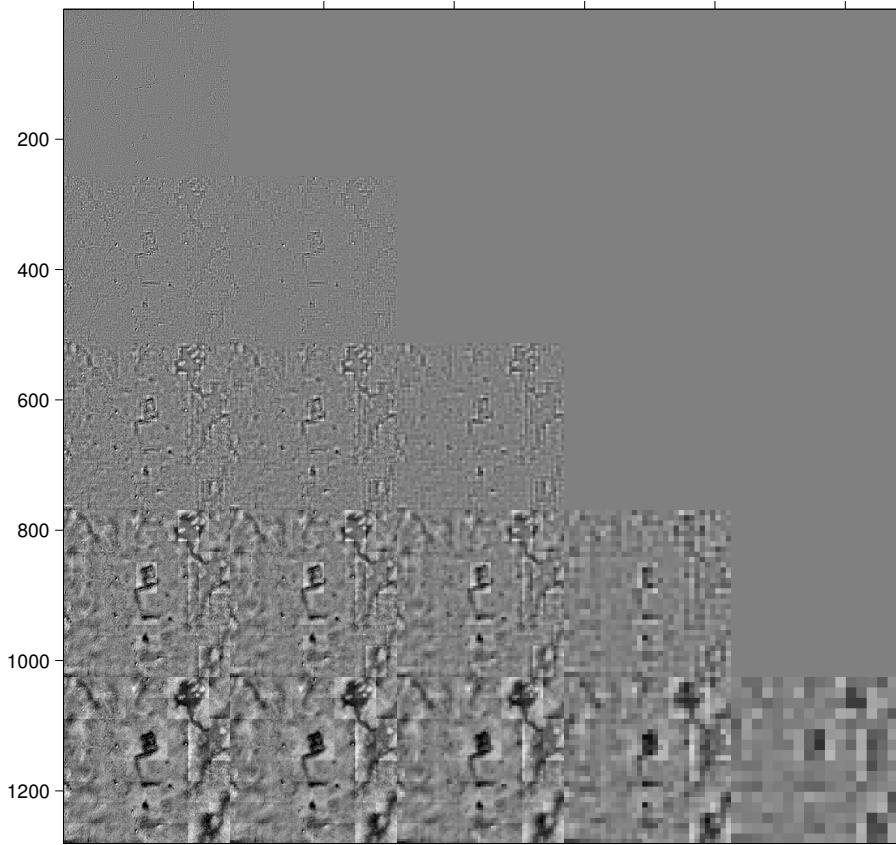


Figure 6.9: wimExplore: db1, N=5:  $R(i,j) = \text{waverec2}(\text{HVD}(i) \dots \text{HVD}(j), db1)$

## 7 Denoising and Local Contrast Enhancement

Based on the experience gained in Sections 5.5 and 6.4 this Section employs wavelets for denoising of 1D and 2D signals, and local contrast enhancement of images. Compression (e.g. JPEG2000), watermarking or blending are other applications that wavelets have been successfully applied to.

Smoothing is accomplished by various standard denoising techniques provided in the Matlab 'Wavelet' package, as well as two enhanced variants thereof.

Two methods for local contrast enhancement are introduced. The first combines denoising and contrast enhancement by effectively reversing the method suggested in [Velde, 1999], who enhance small coeffs while leaving bigger ones untouched. The method presented suppresses small detail coefficients by one of the denoising techniques, and enhances the remaining coefficients by a linear factor. The second method follows the ideas presented by [Unaldi et al., 2013], who propose iterative scaling of approximation coefficients based on the relation to neighboring scales.

All methods are implemented in Matlab package 'dspCers' (see Section Denoising and Local Contrast Enhancement for details).

### 7.1 Denoising and coefficient Enhancement

The 'hard' and 'soft' thresholding methods for signal denoising for a given coefficient  $c$  and threshold  $thr$  are given as

$$c' = \begin{cases} 0 & \text{for } c < thr \\ 0, c & \text{for } c > thr \end{cases} \quad (7.1)$$

$$c' = \begin{cases} 0 & \text{for } c < thr \\ 0, c - thr & \text{for } c > thr \end{cases} \quad (7.2)$$

respectively.

### 7.1.1 Thresholding methods 'equalMax' and 'equalEnergy'

These thresholding modes are implemented as static methods in `dspCers.wavelet.wimAbstract.wimDen` in addition to the Matlab Wavelet Package standard 'Soft' and 'Hard' variants, plus providing some additional enhancements.

- Method 'equalMax' combines the properties of the 'Soft' and 'Hard' methods by rescaling the 'Soft' coefs such that their max is identical to the 'Hard' coef's max.

$$c' = \begin{cases} 0 & \text{for } c < thr \\ 0, (c - thr) * (\frac{\max(c)}{\max(c) - thr}) & \text{for } c > thr \end{cases} \quad (7.3)$$

- Method 'equalEnergy' transfers the energy contained in the thresholded coefs into the non-thresholded coefs, that is, the energy of the noise into the signal, such that the total energy remains constant.

$$E_{Tot} = E_{Noise} + E_{Enh} = \sum_{k=1}^T c_k^2 + \sum_{k=T+1}^K c_k^2 = \sum_{k=T+1}^K (rc_k)^2 = r^2 \sum_{k=T+1}^K c_k^2 = r^2 E_{Enh} \quad (7.4)$$

$$r^2 = \frac{E_{Noise}}{E_{Enh}} + 1 \quad (7.5)$$

- The threshold may be specified as
  - an absolute value (obtained from e.g. any of the Matlab provided methods mentioned in Section 5.6), or
  - a multiple of the 'mad' of the coef distribution (that is, in terms of a robust measure of their spread)
- All settings, including the thresholding method, may be set globally or individually for each level.

### 7.1.2 Coefficient enhancement

Coefficients at each scale may be linearly emphasized by a global or level dependent factor. Thus the weight of individual scales can be manipulated and contributions from different frequency lines to the reconstruction controlled. The method may be applied on top of denoising or on the raw decomposition coefficients. It is easily extended to non-linear scaling functions.

## 7.2 Center-Surround detail enhancement

[Unaldi et al., 2013] have suggested a wavelet based contrast enhancement method that takes coefficient relations between consecutive scale levels into consideration, thereby obtaining a center/surround type of contrast enhancement. Their approach is based on an undecimated DWT (UDWT) derived from the 'a trous' algorithm [Shensa, 1992].

### 7.2.1 UDWT by 'A trous' method

While the standard DWT uses decimation of the signal to achieve scaling, the 'a trous' algorithm ('with holes') dilates the filter by filling in zeros ('holes') [Shensa, 1992], [González-Audícana et al., 2005]. For example for  $j=1$  [Unaldi et al., 2013]

$$h^{(1)} = \{h(-k), 0, h(-k+1), \dots, 0, h(0), 0, h(1), 0, \dots, h(k)\} \quad (7.6)$$

Thus an UDWT is formulated as a convolution with a dyadically upscaled filter kernel. Reconstruction is realized with corresponding filters by taking [Unaldi et al., 2013]

$$c_j = \frac{1}{2}(\tilde{h} * c_{j+1} + \tilde{g} * w_{j+1}) \quad (7.7)$$

They show that the reconstruction filters for the non-bi-orthogonal filter banks they use are just the identity operation such that co-additions of all scales yields the original. The contrast enhancement algorithm they propose scales the approximation coefficients to [Unaldi et al., 2013]

$$c_j^+ = c_j \frac{c_j}{c_{j+1}} \quad \text{for } \frac{c_j}{c_{j+1}} < 1, \quad c_j \text{ else} \quad (7.8)$$

thereby taking the relation of intensities of consecutive scales in a 'center-surround' type of mode into account.

### 7.2.2 Consecutive scale relation enhancement

From this formulation it is not entirely clear how the center-surround enhancements are to be carried over to the reconstruction across scales. Also, it is not clear why coefficient modification is only applied in case the amplitude on the detailed scale is less than that of the next coarser level. Moreover, the relation  $c_j/c_{j+1}$  emphasizes rescaling in low amplitude regions of  $c_{j+1}$  (e.g.  $0.9/0.8 = 1.125$ ,  $0.2/0.1 = 2$ ), which may or not be desirable.

The implementation of the Center-Surround contrast enhancement in package `dspCers.wavelet` is based on the identity property of the reconstruction filters

$$c_0 = c_J + \sum_{j=1}^J w_j \quad (7.9)$$

implicating

$$w_j = c_{j-1} - c_j \quad (7.10)$$

and reformulation of the approximation coefficient manipulation given in Eq. (7.8) in terms of details

$$w_j^+ = c_{j-1}^+ - c_j = c_{j-1} \frac{c_{j-1}}{c_j} - c_j \quad (7.11)$$

for a reconstruction from the enhanced detail scales

$$c_0^+ = c_J + \sum_{j=1}^J w_j^+ \quad (7.12)$$

Furthermore the center-surround enhancement contribution can made scaleable by formulating (along ideas taken from unsharp masking)

$$w_j^{++} = w_j + k(w_j^+ - w_j) \quad (7.13)$$

By making *k* scale level dependent, frequency/size arguments can be incorporated.

### 7.3 Package *dspCers.wavelet*

The functionality presented in Sections 7.1. and 7.2. above is implemented in package 'dspCers', usage examples are given in 'dspCersExamples.m'. The package consists of one abstract class *wimAbstract*, that defines some static functions and interfaces. The classes *wimEnhance*, *wimExplore* and *wimCompare* all implement and extend the *wimAbstract* class.

#### 7.3.1 *dspCers.wavelet.wimEnhance*

This class provides the *wimDenEnhMatrix*, *wimCoefDenEnh* and *wimCoefCenSur* methods. It is suited for standard usage, featuring

- Controlled memory consumption applicable to images of many mPixels
- Option to include/exclude the approximation coefs at the coarsest level from the reconstruction
- Named parameters and sensible defaults

#### 7.3.2 *dspCers.wavelet.wimExplore* and *wimCompare*

These classes provide methods to survey and visualize the effects of several different wavelet parameters (including the type of wavelet, thresholds or number of levels involved). They calculate and store an extensive set of derived entities that take up a lot of memory and should thus only be used with small images (below 1 mPixel). These methods have been extensively used in Section 6.

## 8 Experiments and Results

This Section reports on typical results of denoising and local contrast enhancement by methods outlined in Section Denoising and Local Contrast Enhancement and as implemented in Matlab package 'dspCers'.

### 8.1 *wimCoefDenEnh*: Denoising

Fig. 8.1b) shows the original sample image that exhibits traces of remains of a cattle enclosure in a grassland area in the Silvretta alps. The image in Fig. 8.1a) is denoised by thresholding method 'soft' on decomposition detail levels one and two with thresholds corresponding to two and one sigma in units of mad, respectively. Fig. 8.1c) employs thresholding method 'equalEnergy' with otherwise identical parameters. Thresholding on levels one and two reflects the assumption that noise is mostly comprised of imaging pixel noise and small scale grassland intensity variations.

```
wimEnhance: Image size=[256 256] intensity=[0.173 0.796]
Wavelet: wName=db4 N=5
Method: Fig. 8.1a): 's'. Fig. 8.1c): 'e'
MSD: 2.00 0.50 0.00 0.00 0.00
```

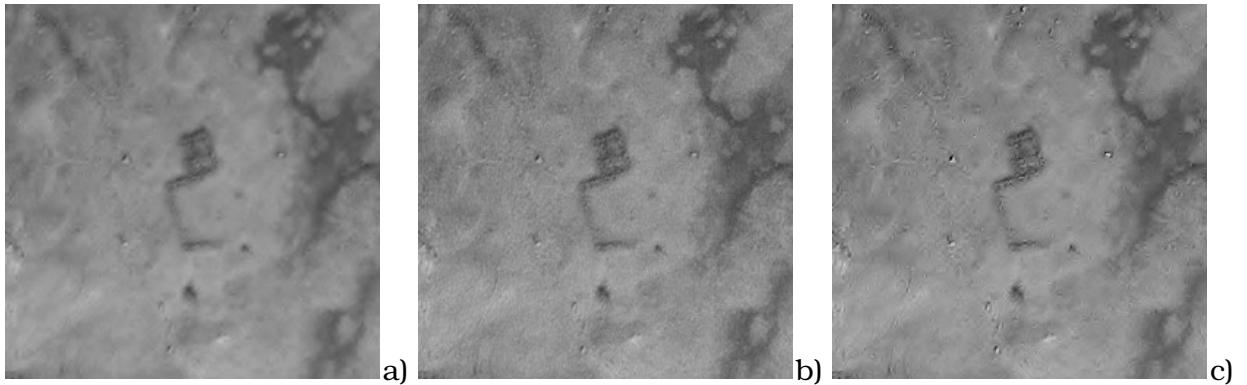


Figure 8.1: 2D denosing with Matlab and own methods

## 8.2 **wimCoefDenEnh: scale level contrast enhancement**

Local contrast enhancement by linear rescaling of detail coefficients is demonstrated for the db1 and db4 wavelets. Results for db1 clearly show the zone of influence at each scale level. Fig. 8.2a shows the original (BW points not set, as opposed to example for DFT), Fig. 8.2b rescaling of mainly detail levels two to four, and Fig. 8.2c strong enhancement of detail coefficients at all decomposition scales.

```
wimEnhance: Image size=[256 256] intensity=[0.173 0.796]
Wavelet: wName=db1 N=4
Method: 'e'
Fig. 8.2b)
MSD: 1.00 0.50 0.25 0.10
ENH: 1.00 2.00 3.00 2.00
Fig. 8.2c)
MSD: 1.00 1.00 1.00 1.00
ENH: 3.00 3.00 3.00 3.00
keepApp: false
```

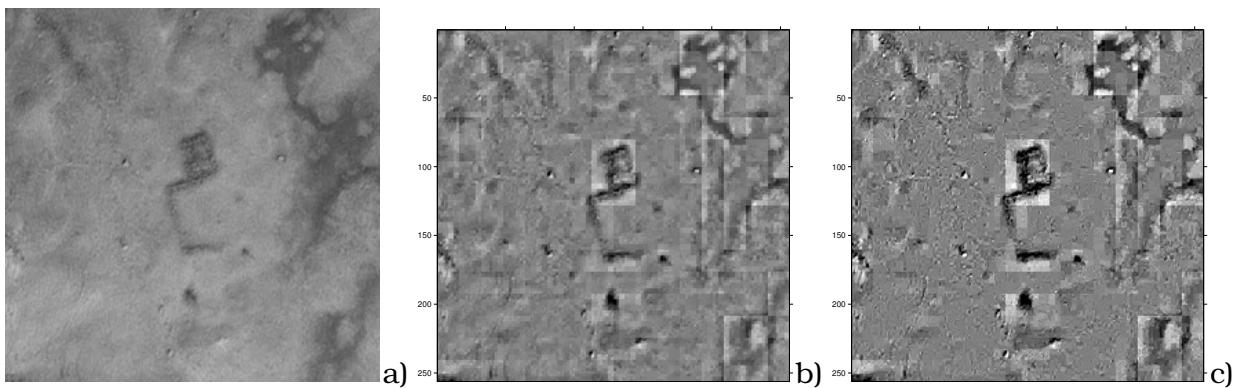


Figure 8.2: wimEnhance: db1

Fig. 8.3 demonstrates results for similar enhancement parameters for the db4 wavelet. Fig. 8.3a leaves the approximation coefs intact, while Figs. 8.3b and 8.3c set to these coefs to zero. The reduction in large scale intensity variations is evident from the comparison between Figs. 8.3a and 8.3b at otherwise identical parameters.

```
wimEnhance: Image size=[256 256] intensity=[0.173 0.796]
```

```

Wavelet: wName=db4 N=4
Method: 'e'
Fig. 8.3a)
MSD: 1.00 0.50 0.25 0.10
ENH: 1.00 2.00 3.00 2.00
keepApp: true
Fig. 8.3b)
MSD: 1.00 0.50 0.25 0.10
ENH: 1.00 2.00 3.00 2.00
keepApp: false
Fig. 8.3c)
MSD: 1.00 1.00 1.00 1.00
ENH: 3.00 3.00 3.00 3.00
keepApp: false

```

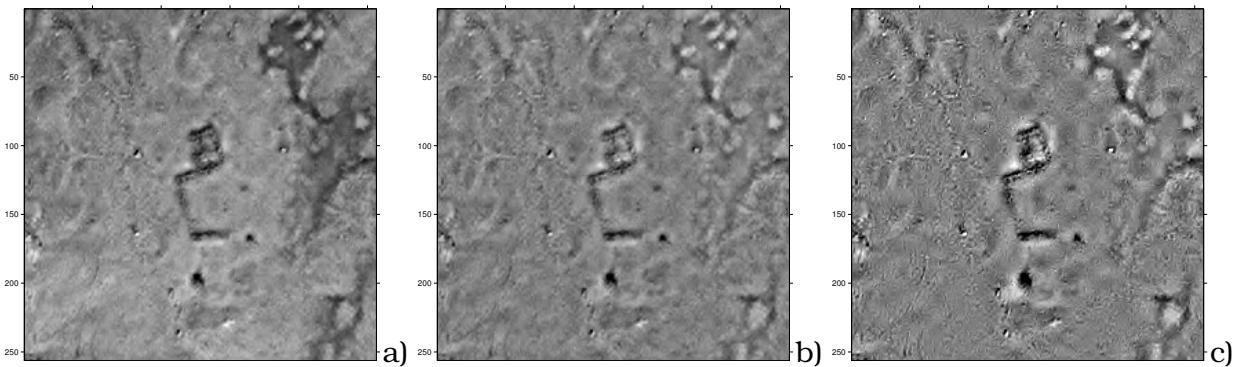


Figure 8.3: wimEnhance: db4

### 8.3 wimCoefCenSur: center-surround contrast enhancement

The calculation for results in Fig. 8.4 employs the inter-scale coefficient ratio based contrast enhancement method, implemented via the 'a trous' algorithm for an undecimated DWT. Fig. 8.4a shows the default enhancement as suggested by [REF], while Figs. 8.4b,c demonstrate the additional contrast emphasis as proposed in Section [REF] in this work. Levels two to four have been selected for parameter rescaling due to scales of relevant objects in the test image . The filter used ('CSfilter') is  $[1 \ 2 \ 1]/4$ . Enhancement factors ('CSfactor') are given below. Obviously edge effects are not handled correctly in the current implementation.

```

wimCenSur: Image size=[256 256] intensity=[0.173 0.796]
Wavelet: wName=db4 N=5
Method: e e e e e
MSD: 1.00 0.50 0.25 0.10 0.00
CSfilter: 0.250 0.500 0.250
Fig. 8.4a)
CSfactor: 1.000 1.000 1.000 1.000 1.000
Fig. 8.4b)
CSfactor: 1.000 2.000 2.000 2.000 1.000
Fig. 8.4c)
CSfactor: 1.000 5.000 5.000 5.000 1.000

```

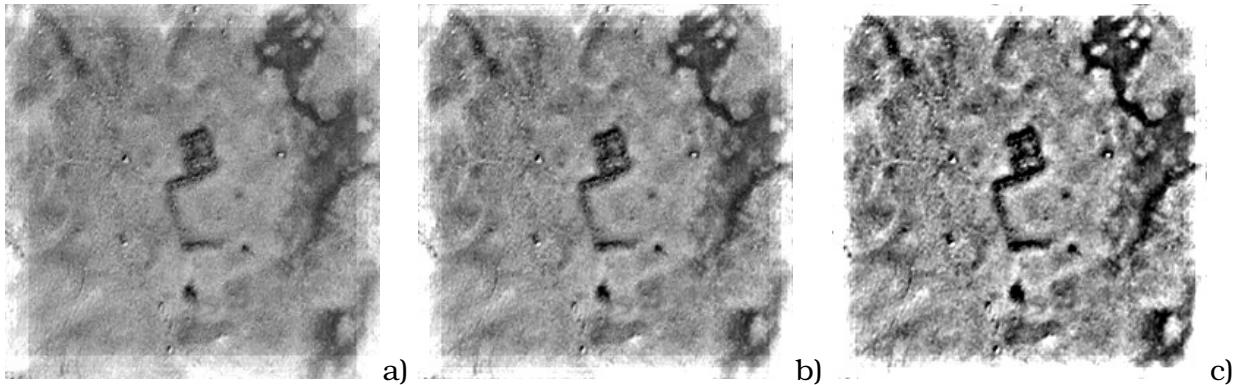


Figure 8.4: wimCenSur: db4

## 9 Discussion and Conclusion

This section summarizes results on wavelet based denoising and local contrast enhancement methods and experiments presented in Sections 6 and 8.

### 9.0.1 Denoising and coefficient scaling

While the denoising effect with method 'soft' in Fig. 8.1a is evident, the result suffers from a general loss of detail. Method 'equalEnergy' (see Fig. 8.1c) on the other side manages to contain reduction of noise to predominantly smooth areas, but exhibits small scale artifacts in regions of originally high contrast.

While tuning of parameters may alleviate unwanted effects, a more refined coefficient thresholding method with a smooth mapping function is called for.

Contrast enhancement usually tends to increase noise if not specifically handled. Thus the idea behind the detail coefficient manipulation based contrast enhancement is to first decrease noise by one of the thresholding methods, and then to enhance the remaining parameters in appropriate decomposition levels. As is known for the 'hard' threshold method, artifacts tend to appear even if non-thresholded coeffs are not rescaled. These effects become increasingly evident with more pronounced coef rescaling (see Fig. 8.3c).

As with the denoising, artifact suppression will profit from smoother mapping functions. Presumably an optimal mapping function will provide a smooth transition from coeff suppression for denoising to coeff enhancement for contrast enhancement.

### 9.0.2 Tunable center-surround enhancement

For the inter-level ratio contrast enhancement method, Figs. 8.4b and c suggest that this method is less prone to building artifacts even at high enhancement levels (as compared to Fig. 8.3c). Although only one simple decomposition UWT filter was tried (as opposed to the many experiments with different wavelets for the detail coef rescaling method), this method yields more promising results, in particular in combination with the tuneable contrast enhancement suggested in this work.

### 9.0.3 Summary example

Finally, Figs. 9.1 to 9.3 summarize typical results for denoising (Fig. 9.1b), local contrast enhancement based on individual detail scale coefficients (Fig. 9.2) and center-surround inter-scale level coefficient manipulation (Fig. 9.3). Figs. 9.2b and 9.3b take contrast enhancement to extreme levels in an attempt to make artifacts set in.

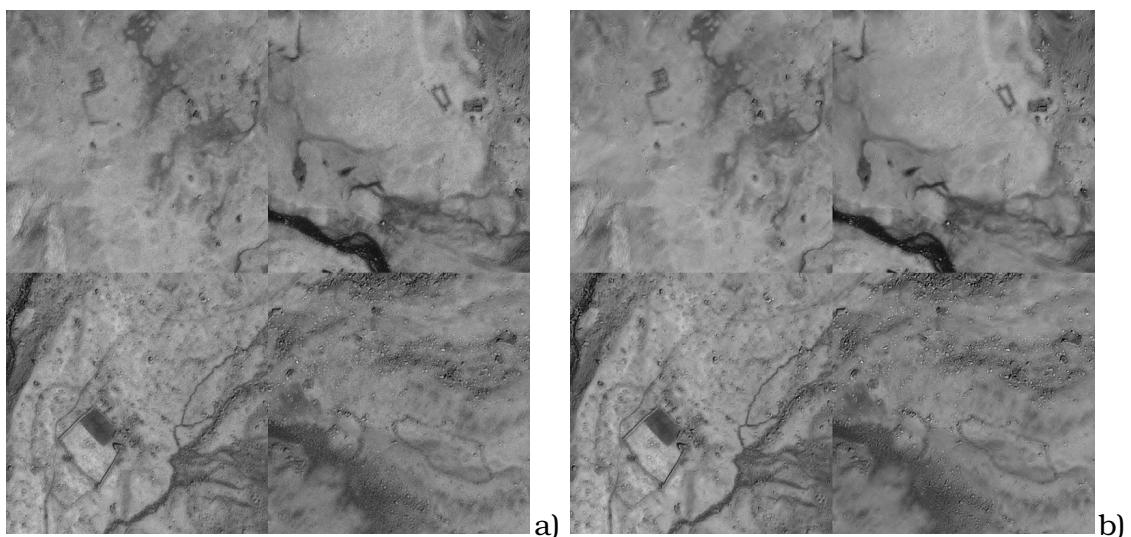


Figure 9.1: wimEnhance: denoise

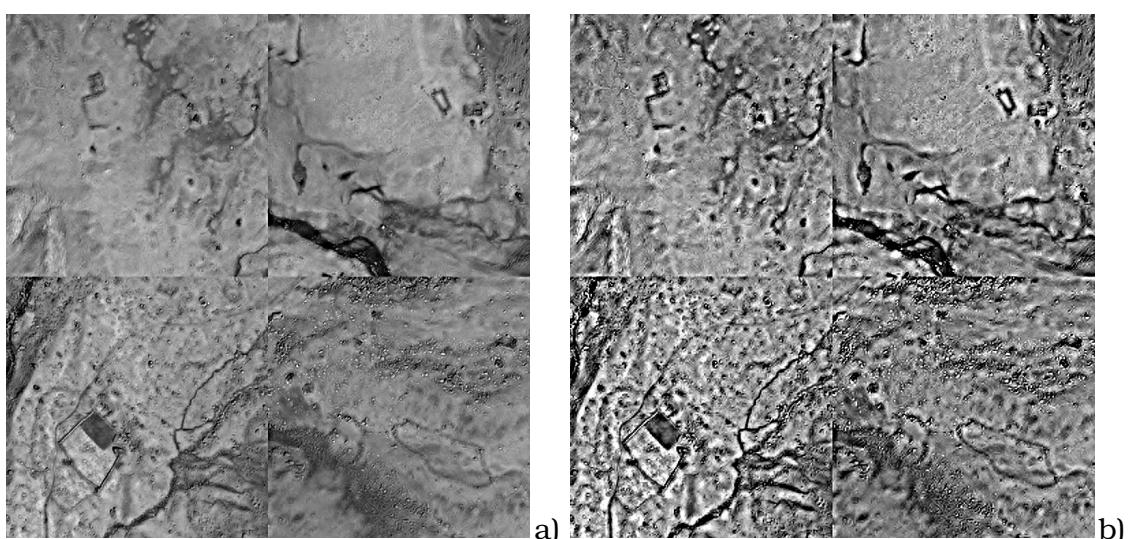


Figure 9.2: wimEnhance: enhance

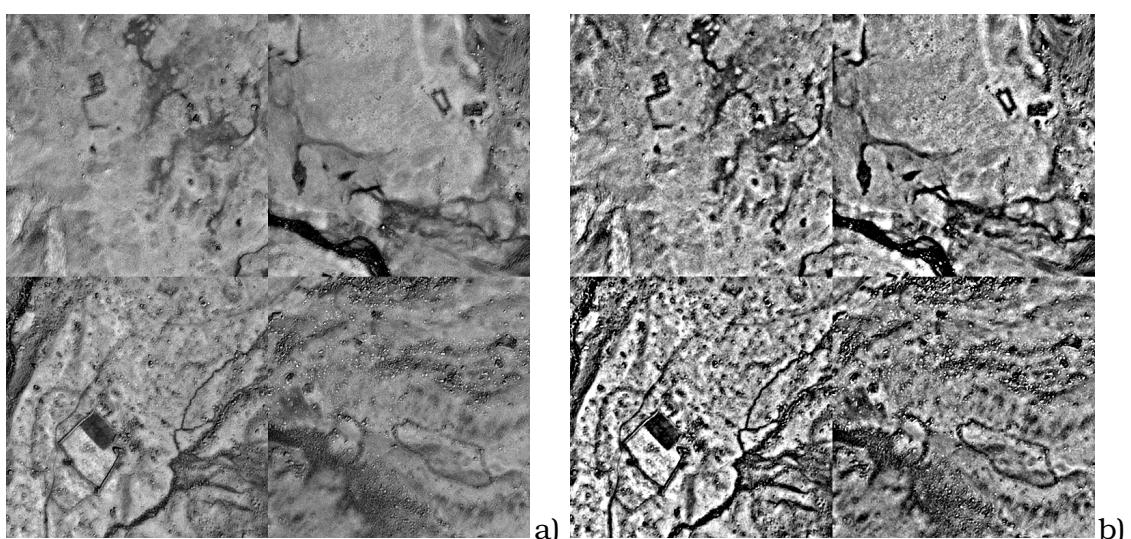


Figure 9.3: wimCenSur: enhance

## References

- Gerhard Doblinger. *Matlab-Programmierung in der digitalen Signalverarbeitung*. J.Schlemmbach Fachverlag, 2001. ISBN 3-935340-02-8.
- M. González-Audicana, X. Otazu, O. Fors, and A. Seco. Comparison between Mallat's and the 'à trous' discrete wavelet transform based algorithms for the fusion of multispectral and panchromatic images. *International Journal of Remote Sensing*, 26(3):595–614, February 2005. ISSN 0143-1161, 1366-5901. doi: 10.1080/01431160512331314056. URL <http://www.tandfonline.com/doi/abs/10.1080/01431160512331314056>.
- Tzu-Heng Henry Lee. Wavelet Analysis for Image Processing. *Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan, ROC*, 2008. URL [http://disp.ee.ntu.edu.tw/henry/wavelet\\_analysis.pdf](http://disp.ee.ntu.edu.tw/henry/wavelet_analysis.pdf).
- Guomin Luo and Daming Zhang. *Wavelet Denoising, Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, volume Chapter 4. In-Tech, 2012. ISBN 978-953-51-0494-0. URL <http://www.intechopen.com/books/advances-in-wavelet-theory-and-their-applications-in-engineering-physics-and-technology-wavelet-denoising>.
- R. C. P. Marques, C. M. Laprano, and F. N. S. Medeiros. Multiscale Denoising Algorithm Based on the à Trous Algorithm. In *XV Brazilian Symposium on Computer Graphics and Image Processing, 2002. Proceedings*, page 400, 2002. doi: 10.1109/SIBGRA.2002.1167172.
- Michel Misiti, Yves Misiti, Georges Oppenheim, and Jean-Michel Poggi. *Wavelets and their Applications*. Wiley-ISTE, 4 edition, May 2007. ISBN 978-1-905209-31-6.
- M. Shensa. The discrete wavelet transform: wedding the a trous and Mallat algorithms. *IEEE Transactions on Signal Processing*, 40(10):2464–2482, October 1992. ISSN 1053-587X. doi: 10.1109/78.157290.
- M. Sifuzzaman, M.R. Islam, and M.Z. Ali. Application of Wavelet Transform and its Advantages Compared to Fourier Transform. *Journal of Physical Sciences*, 13:121–134, 2009. ISSN 0972-8791. URL <http://www.vidyasagar.ac.in/journal>.
- Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image Processing, Analysis, and Machine Vision*. Cengage Learning, Toronto, 3 edition edition, March 2007. ISBN 9780495082521.
- Bouden Toufik and Nibouche Mokhtar. The Wavelet Transform for Image Processing Applications. In *Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, volume Chapter 17. INTECH Open Access Publisher, 2012. ISBN 9789535104940. URL <http://cdn.intechopen.com/pdfs/34963.pdf>.
- Numan Unaldi, Samil Temel, and Süleyman Demirci. Undecimated Wavelet Transform Based Contrast Enhancement. *International Journal of Computer, Control, Quantum and Information Engineering*, 7(9):571 – 574, 2013. ISSN 1307-6892. URL <http://waset.org/Publications?p=81>.
- K.V. Velde. Multiscale contrast enhancement for medical images. Internal report KUL/ESAT/PSI/9807, ESAT, Belgium, August 1998. URL [https://mirc.uzleuven.be/tools/download.php?root=MedicalImageComputing\\_UID=u7WU2tM1LvsBXuy1Wl2aPWaCO7DLTY](https://mirc.uzleuven.be/tools/download.php?root=MedicalImageComputing_UID=u7WU2tM1LvsBXuy1Wl2aPWaCO7DLTY).
- K.V. Velde. Multi-scale color image enhancement. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, volume 3, pages 584–587 vol.3, 1999. doi: 10.1109/ICIP.1999.817182.
- Martin Werner. *Digitale Signalverarbeitung mit MATLAB*. Vieweg+Teubner Verlag, 5 edition, 2012. ISBN 978-3-8348-1473-9.