Mathematical Foundations of Data Sciences

Gabriel Peyré
CNRS & DMA
École Normale Supérieure
gabriel.peyre@ens.fr
https://mathematical-tours.github.io
www.numerical-tours.com

November 15, 2020

Chapter 3

Wavelets

The reference for this chapter is [17].

3.1 Multi-resolution Approximation Spaces

A multiresolution approximation of $L^2(\mathbb{R})$ is a set of nested closed subspaces $(V_j)_j$

$$L^2(\mathbb{R}) \supset \ldots \supset V^{j-1} \supset V_j \supset V_{j+1} \supset \ldots \supset \{0\} \tag{3.1}$$

which must be related one from each other by dyadic scaling and must also be stable by dyadic translation

$$f \in V_j \iff f(\cdot/2) \in V_{j+1} \quad \text{and} \quad f \in V_j \iff \forall\, n \in \mathbb{Z},\ f(\cdot + n2^j) \in V_j$$

So large j corresponds to coarse approximation spaces, and 2^j is often call the "scale".

The limit on the left of (3.1) means that $\cup_j V_j$ is dense in $L^2(\mathbb{R})$, or equivalently that $P_{V_j}(f) \to f$ as $j \to -\infty$ where P_V is the orthogonal projector on V

$$P_V(f) = \operatorname*{argmin}_{f' \in V} \|f - f'\|.$$

The limit on the right of (3.1) means that $\cap_j V_j = \{0\}$, or equivalently that $P_{V_j}(f) \to 0$ as $j \to +\infty$.

The first example is piecewise constant functions on dyadic intervals

$$V_j = \left\{ f \in L^2(\mathbb{R}) \ ;\ \forall n, f \text{ is constant on } [2^j n, 2^j(n+1)[\right\}, \tag{3.2}$$

A second example is the space used for Shannon interpolation of bandlimitted signals

$$V_j = \left\{ f \ ;\ \mathrm{Supp}(\hat{f}) \subset [-2^{-j}\pi, 2^{-j}\pi] \right\} \tag{3.3}$$

which corresponds to function which can be exactly represented using samples $f(2^j n)$ (to be compared with piecewise constant signal on a grid with spacing 2^j). In this case, the orthogonal projection is the bandlimitting operator

$$P_{V_j}(f) = \mathcal{F}^{-1}(\hat{f} \odot 1_{[-2^{-j}\pi, 2^{-j}\pi]}).$$

Scaling functions. We also require that there exists a scaling function $\varphi \in L^2(\mathbb{R})$ so that

$$\{\varphi(\cdot - n)\}_n \text{ is an Hilertian orthonormal basis of } V_0.$$

By the dilation property, this implies that

$$\{\varphi_{j,n}\}_n \text{ is an Hilertian orthonormal basis of } V_j \quad \text{where} \quad \varphi_{j,n} \stackrel{\text{def.}}{=} \frac{1}{2^{j/2}} \varphi\left(\frac{\cdot - 2^j n}{2^j} \right).$$

The normalization is such to ensure $\|\varphi_{j,n}\| = 1$.

Note that one then has

$$P_{V_j}(f) = \sum_n \langle f, \varphi_{j,n}\rangle \varphi_{j,n}.$$

Figure 3.1 illustrates the translation and scaling effect.

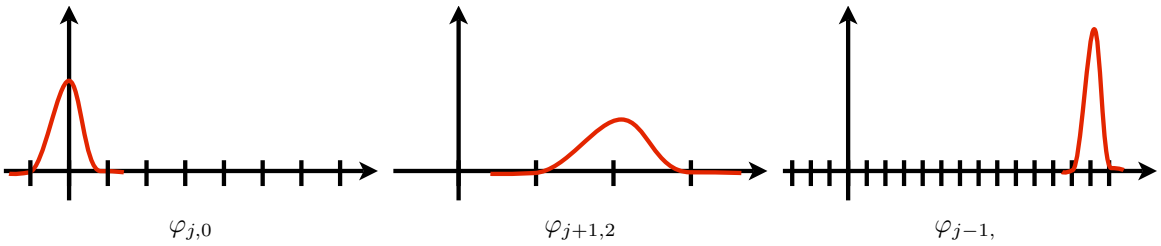

Figure 3.1: Translation and scaling to generate approximation spaces.

For the case of piecewise constant signals (3.2), one can use

$$\varphi = 1_{[0,1[} \quad \text{and} \quad \varphi_{j,n} = 2^{-j/2} 1_{[2^j n, 2^j(n+1)[}.$$

For the case of Shannon multiresolution (3.3), one can use $\varphi(t) = \sin(\pi t)/(\pi t)$ and one verifies

$$\langle f, \varphi_{j,n}\rangle = f(2^j n) \quad \text{and} \quad f = \sum_n f(2^j n)\varphi_{j,n}.$$

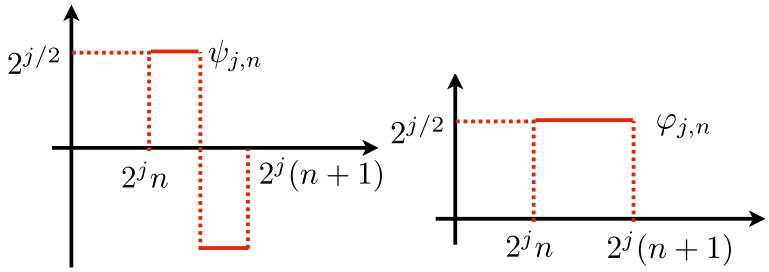

Figure 3.2: **BUG: the elevation should be $2^{-j/2}$ and not $2^{j/2}$** Haar scaling (left) and wavelet (right) functions.

Spectral orthogonalization. In many case of practical interest, the space V_j is describe by a translation-invariant basis which is not-orthogonal, $V_j = \mathrm{Span}(\theta(\cdot - n))_{n\in\mathbb{Z}}$. The following proposition shows how to orthogonalize it using the Fourier transform. Figure 3.3 shows how it leads to cardinal spline orthogonal functions.

Proposition 11. *For $\theta \in L^2(\mathbb{R})$ (assumed regular and with fast enough decay), $\{\theta(\cdot - n)\}_{n\in\mathbb{Z}}$ is orthonormal if and only if*

$$\forall \omega, \quad A(\omega) \stackrel{\text{def.}}{=} \sum_k |\hat\theta(\omega - 2\pi k)|^2 = 1.$$

If there exists $0 < a \leqslant b < +\infty$ such that $a \leqslant A(\omega) \leqslant b$, then φ defined by

$$\hat\varphi(\omega) = \frac{\hat\theta(\omega)}{\sqrt{A(\omega)}}$$

is such that $\{\varphi(\cdot - n)\}_{n\in\mathbb{Z}}$ is an Hilbertian basis of $\mathrm{Span}\{\theta(\cdot - n)\}_{n\in\mathbb{Z}}$.

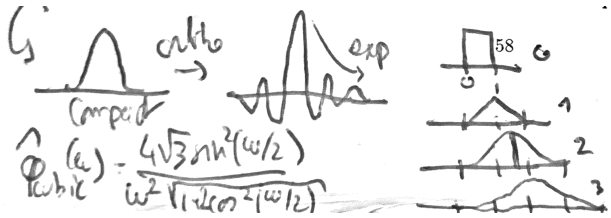

Figure 3.3: Orthogonalization of b-spline to defines cardinal orthogonal spline functions.

Proof. One has that $\{\theta(\cdot - n)\}_{n\in\mathbb{Z}}$ is orthonormal if and only if

$$\langle \theta, \theta(\cdot - n)\rangle = (\theta \star \bar{\theta})(n) = \delta_{0,n} \overset{\text{def.}}{=} \begin{cases} 0 & \text{if } n = 0, \\ 1 & \text{otherwise.} \end{cases}$$

where $\bar{\theta} = \theta(-\cdot)$ and δ_0 is the discrete Dirac vector. Recall the Poisson summation formula (1.8) for any function h

$$\sum_n h(\omega - 2n\pi) = \sum_n h(n)e^{-\mathrm{i}n\omega}$$

which here reads

$$\sum_n \mathcal{F}(\theta \star \bar{\theta})(\omega - 2\pi n) = \sum_n \delta_{0,n} e^{-\mathrm{i}n\omega} = 1.$$

We conclude with the Fourier-convolution formula (2.6) shows that $\mathcal{F}(\theta\star\bar{\theta})(\omega) = \hat{\theta}(\omega)\hat{\theta}(\omega)^* = |\hat{\theta}(\omega)|^2$ which leads to the desired formula, and it is if and only if. Normalizing by $1/\sqrt{A(\omega)}$, which is a bounded function, shows that $\hat{\varphi}$ satsifies $\sum_k |\hat{\varphi}(\omega - 2\pi k)|^2 = 1$. □

A typical example of application is spline (e.g. cubic ones) interpolations, which are generated by the box-spline function θ which is a piecewise polynomial with a compact support.

3.2 Multi-resolution Details Spaces

The details spaces are defined as orthogonal complement of $V_j \subset V_{j-1}$, which is legit because these are closed subspaces

$$\forall\, j, \quad W_j \text{ is such that } V_{j-1} = V_j \oplus^{\perp} W_j.$$

This leads to the following sequence of embedded spaces

$$\mathrm{L}^2(\mathbb{R}) \longrightarrow \cdots \searrow V_{j-1} \longrightarrow V_j \searrow V_{j+1} \longrightarrow \cdots \searrow \longrightarrow \{0\}$$
$$W_{j-1} \qquad W_j \qquad W_{j+1}$$

Once again, we suppose that W_0 has an Hilbertian ortho-basis of the form $\{\psi(\cdot - n)\}_n$, so that

$$\{\psi_{j,n}\}_n \text{ is an Hilertian orthonormal basis of } W_j \quad \text{where} \quad \psi_{j,n} \overset{\text{def.}}{=} \frac{1}{2^{j/2}}\psi\left(\frac{\cdot - 2^j n}{2^j}\right).$$

Due to the orthogonal complementarity property, one has

$$L^2(\mathbb{R}) = \bigoplus_{j=-\infty}^{j=+\infty} W_j = V_{j_0} \bigoplus_{j\leqslant j_0}^{j} V_j.$$

This means that for all $f \in L^2(\mathbb{R})$, one has the following convergence in $L^2(\mathbb{R})$,

$$f = \lim_{(j_-,j_+)\to(-\infty,+\infty)} \sum_{j=j_-}^{j_+} P_{W_j} f = \lim_{j_+\to+\infty} P_{V_{j_0}} f + \sum_{j=j_0}^{j_+} P_{W_j} f.$$

This decomposition shows that

$$\left\{\psi_{j,n} \;;\; (j,n) \in \mathbb{Z}^2\right\}$$

is an Hilbertian orthogonal basis of $L^2(\mathbb{R})$, which is called a wavelet basis. One also have a "truncated" ortho-basis

$$\{\psi_{j,n} \;;\; j \leqslant j_0, n \in \mathbb{Z}\} \cup \{\varphi_{j_0,n} \;;\; n \in \mathbb{Z}\}.$$

A (forward) Wavelet transform corresponds to the computation of all the inner products of some function f with the elements of these basis.

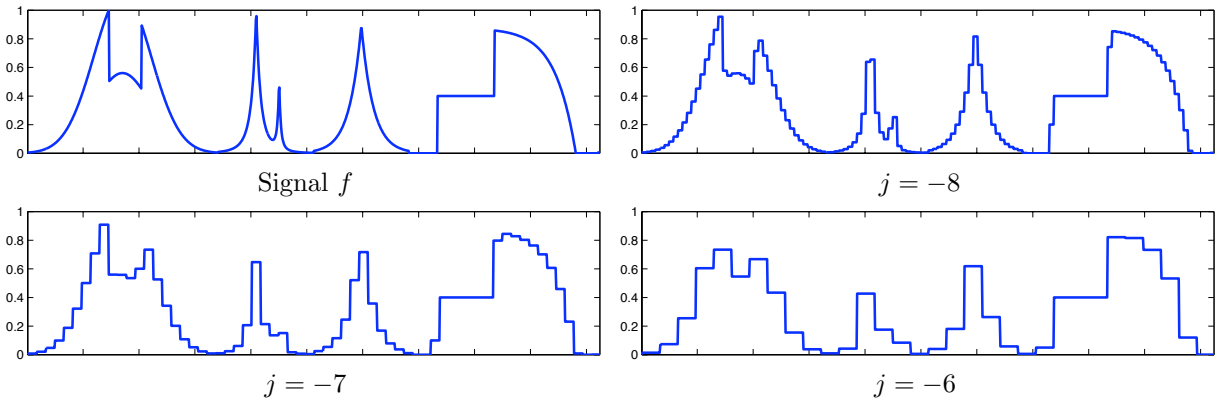

Figure 3.4: 1-D Haar multiresolution projection $P_{V_j} f$ of a function f.

Haar wavelets. For the Haar multiresolution (3.2), one has

$$W_j = \left\{ f \;;\; \forall n \in \mathbb{Z},\; f \text{ constant on } [2^{j+1}n, 2^{j+1}(n+1)) \quad \text{and} \quad \int_{n2^j}^{(n+1)2^j} f = 0 \right\}. \tag{3.4}$$

A possible choice for a mother wavelet function is

$$\psi(t) = \frac{1}{\sqrt{2}} \begin{cases} 1 & \text{for} \quad 0 \leqslant t < 1/2, \\ -1 & \text{for} \quad 1/2 \leqslant t < 1, \\ 0 & \text{otherwise,} \end{cases}$$

as shown on Figure 3.2, right.

Figure 3.5 shows examples of projections on details spaces, and how they can be derived from projection on approximation spaces.

Shannon and splines. Figure 3.6 shows that Shannon and splines corresponds to a hard and a soft segmentation of the frequency domain (Shannon being in some sense the high degree limit of splines).

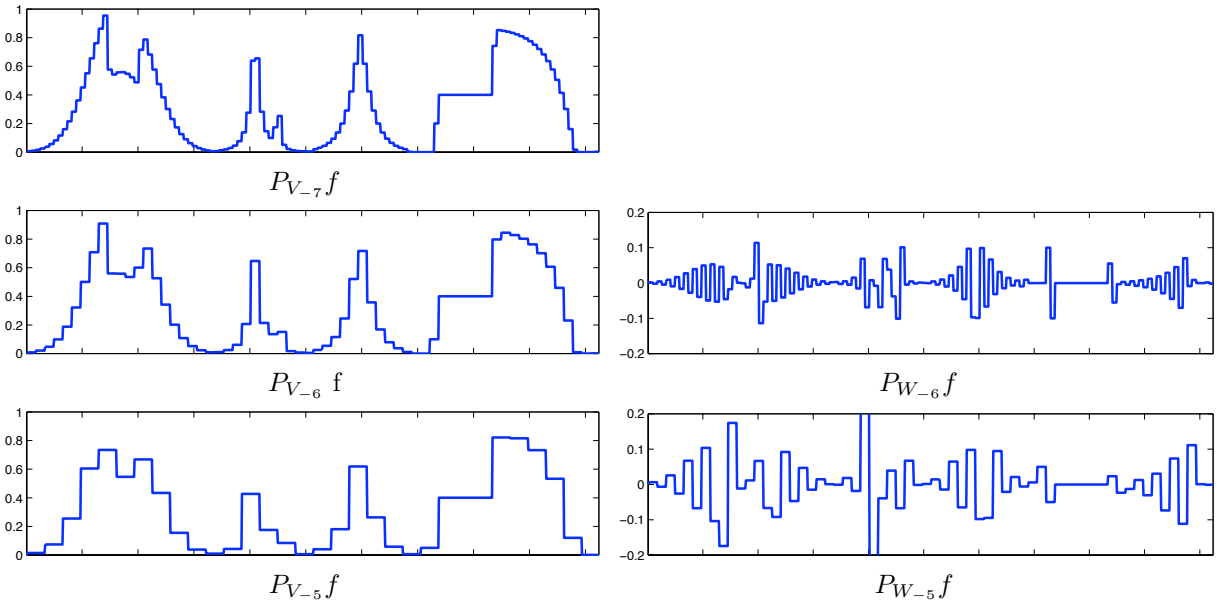

Figure 3.5: Projection on Haar approximation spaces (left) and detail spaces (right).

3.3 On Bounded Domains

On a periodic bounded domain $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ (note that we use here 1-periodicity, in contrast to the convention we used for Fourier series of 2π-periodicity), one obtains an orthogonal wavelet basis of $L^2(\mathbb{T})$ by periodizing the original wavelets, and also restricting the translation points $2^j n$ to be in $[0,1]$, i.e. $0 \leqslant n < 2^{-j}$. Similarly to (1.6), the periodization of a function $f \in L^1(\mathbb{R})$ is the function

$$f^P = \sum_{n \in \mathbb{Z}} f(\cdot - n) \in L^1(\mathbb{T}).$$

The wavelet basis is thus define as

$$\left\{ \psi^P_{j,n} \;;\; j \leqslant j_0, 0 \leqslant n < 2^{-j} \right\} \cup \left\{ \varphi^P_{j_0,n} \;;\; 0 \leqslant n < 2^{-j_0} \right\}.$$

and one verifies that it defines an Hilbertian ortho-basis of $L^2(\mathbb{T})$, see Figure 3.7. It is possible to define wavelet basis using Neumann (mirror) boundary conditions, but this is more involved.

3.4 Fast Wavelet Transform

3.4.1 Discretization

We now work over $\mathbb{R}/\mathbb{Z}$. The modeling hypothesis is that one has access to a discrete signal $a_J \in \mathbb{R}^N$ with $N = 2^{-J}$ at some fixed scale 2^J, and that this signal exactly matches the inner-product with the scaling functions, i.e.

$$\forall\, n, \in \{0, \ldots, N-1\}, \quad a_{J,n} = \langle f, \varphi^P_{J,n} \rangle \approx f(2^J n), \tag{3.5}$$

for some function of interest f we are sampling. This is equivalent to saying that the discretization process have exactly access to $P_{V_J} f$. This hypothesis is questionnable, and similar to the Shannon bandlimit assumption. In practice, the scaling functions $(\varphi_{J,n})_n$ are often quite close to the point-spread function of the acquisition device, so it is acceptable. One can however improves this by correcting the acquired values by the device to be closer to asumption (3.5).

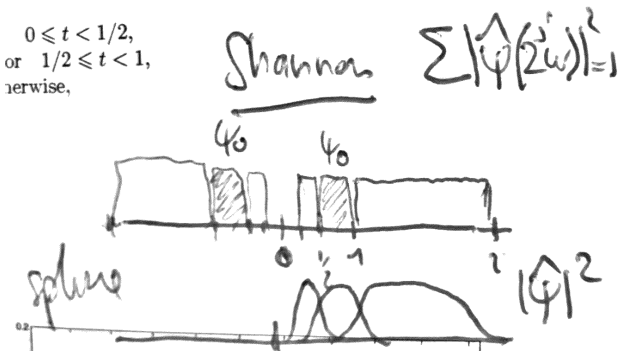

Figure 3.6: Spline and Shannon wavelet segmentation of the frequency axis.

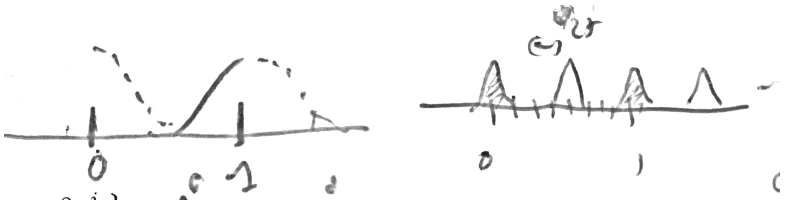

Figure 3.7: Periodizing basis functions. This leads to considering only $0 \leqslant n2^{-j}$, so that in practice one choose $j_0 = 0$.

The discrete wavelet transform then computes, from this input a_J, all the coefficients

$$\forall\, j \in \{J+1, J+2, \ldots, 0\}, \quad \forall\, n \in [\![0, 2^{-j}-1]\!], \quad a_{j,n} \stackrel{\text{def.}}{=} \langle f, \varphi^P_{j,n}\rangle, \quad \text{and} \quad d_{j,n} \stackrel{\text{def.}}{=} \langle f, \psi^P_{j,n}\rangle$$

in this order (increasing values of j). During the algorithm, the previously computed vector a_j can be discarded, and only the d_j are kept.

The forward discrete wavelet transform on a bounded domain is thus the orthogonal finite dimensional map

$$a_J \in \mathbb{R}^N \longmapsto \big\{d_{j,n}\ ;\ 0 \leqslant j < J, 0 \leqslant n < 2^{-j}\big\} \cup \{a_0 \in \mathbb{R}\}.$$

The inverse transform, which is thus the adjoint due to orthogonality, is the inverse map.

Figure 3.8 shows examples of wavelet coefficients. For each scale 2^j, there are 2^{-j} coefficients.

3.4.2 Forward Fast Wavelet Transform (FWT)

The algorithm proceeds by computing a series of simple operators

$$\forall\, j = J+1, \ldots, 0, \quad (a_j, d_j) = \mathcal{W}_j(a_{j-1}) \quad \text{where} \quad \mathcal{W}_j : \mathbb{R}^{2^{-j+1}} \to \mathbb{R}^{2^{-j}} \times \mathbb{R}^{2^{-j}} \tag{3.6}$$

The number of such steps is thus $|J| = \log_2(N)$. Each $\mathcal{W}_j$ is orthogonal since it corresponds to linear maps between coefficients in orthogonal bases.

In order to describe the algorithm that computes $\mathcal{W}_j$, we introduce the filters "filter" coefficients $f, g \in \mathbb{R}^{\mathbb{Z}}$

$$h_n \stackrel{\text{def.}}{=} \frac{1}{\sqrt{2}}\langle \varphi(\cdot/2), \varphi(\cdot - n)\rangle \quad \text{and} \quad g_n \stackrel{\text{def.}}{=} \frac{1}{\sqrt{2}}\langle \psi(\cdot/2), \varphi(\cdot - n)\rangle. \tag{3.7}$$

Figure 3.9 illustrates the computation of these weights for the Haar system.

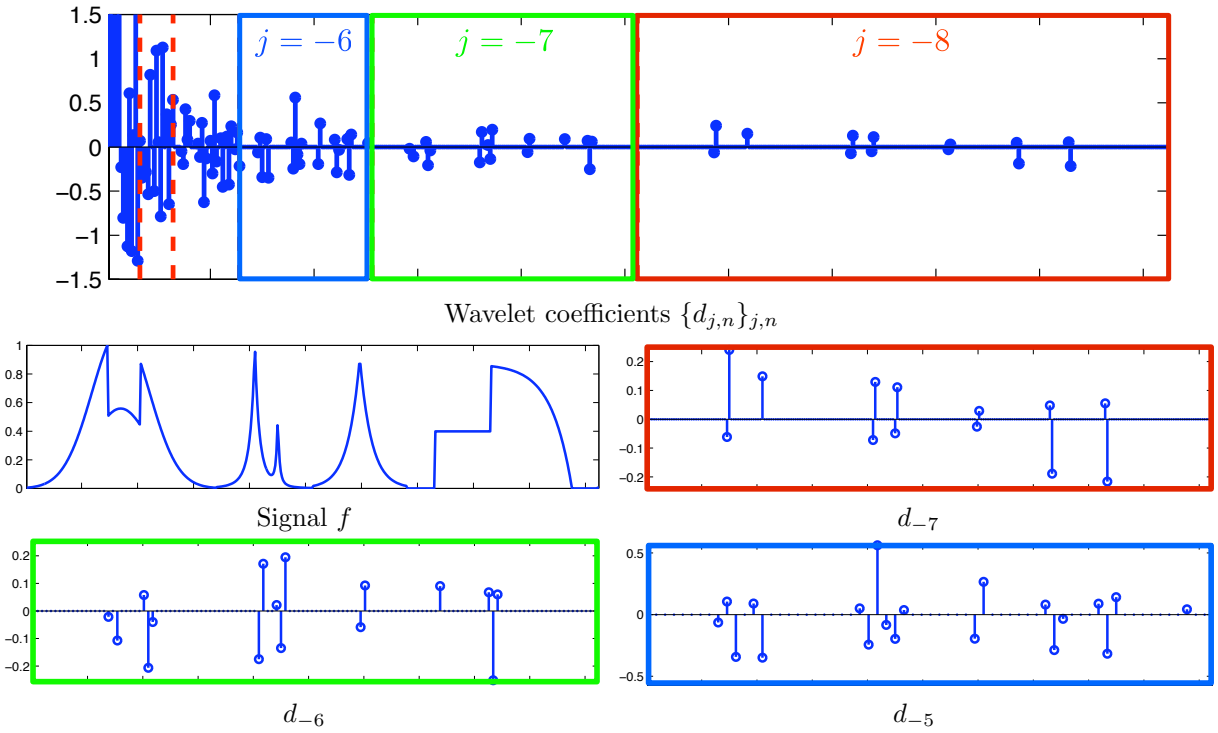

Figure 3.8: Wavelet coefficients. Top row: all the coefficients. Bottoms rows: zoom on the different scales

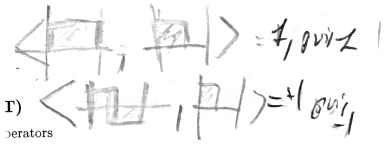

Figure 3.9: Haar wavelets weights as inner products.

We denote as $\downarrow_2 : \mathbb{R}^K \to \mathbb{R}^{K/2}$ the subsampling operator by a factor of 2, i.e.

$$u \downarrow_2 \stackrel{\text{def.}}{=} (u_0, u_2, u_4, \ldots, u_{K-2}, u_K).$$

In the following, we assume that these filters are decaying fast enough.

Proposition 12. *One has*

$$\mathcal{W}_j(a_j) = ((\bar{h} \star a_{j-1}) \downarrow_2, (\bar{g} \star a_{j-1}) \downarrow_2), \tag{3.8}$$

where $\star$ denotes periodic convolutions on $\mathbb{R}^{-j+1}$.

Proof. We note that $\varphi(\cdot/2)$ and $\psi(\cdot/2)$ are in $\mathcal{W}_j$, so one has the decompositions

$$\frac{1}{\sqrt{2}}\varphi(t/2) = \sum_n h_n \varphi(t-n) \quad \text{and} \quad \frac{1}{\sqrt{2}}\psi(t/2) = \sum_n g_n \varphi(t-n) \tag{3.9}$$

Doing the change of variable $t \mapsto \frac{t-2^j p}{2^{j-1}}$ in (3.9), one obtains

$$\frac{1}{\sqrt{2}}\varphi\left(\frac{t-2^j p}{2^j}\right) = \sum_n h_n \varphi\left(\frac{t}{2^{j-1}} - (n+2p)\right)$$

(similarly for ψ) and then doing the change $n \mapsto n-2p$, one obtains

$$\varphi_{j,p} = \sum_{n\in\mathbb{Z}} h_{n-2p}\varphi_{j-1,n} \quad \text{and} \quad \psi_{j,p} = \sum_{n\in\mathbb{Z}} g_{n-2p}\psi_{j-1,n}.$$

When working with periodized function $(\varphi^P_{j,n}, \psi^P_{j,p})$, this formula is still valid, but the summation over $n \in \mathbb{Z}$ should be done modulo 2^{-j+1}. Taking inner product of both size with respect to f (which is legit if h, g are decaying fast enough), one obtains the fundamental recursion fromula

$$a_{j,p} = \sum_{n\in\mathbb{Z}} h_{n-2p}a_{j-1,n} = (\bar{h} \star a_{j-1})_{2p} \quad \text{and} \quad d_{j,p} = \sum_{n\in\mathbb{Z}} g_{n-2p}a_{j-1,n} = (\bar{g} \star a_{j-1})_{2p} \tag{3.10}$$

where $\bar{u}_n \overset{\text{def.}}{=} u_{-n}$. One can show that this formula is still valid when working over a bounded interval $\mathbb{T} = \mathbb{R}/\mathbb{Z}$, but then $\star$ denotes the periodic convolution over $\mathbb{Z}/2^{-j+1}\mathbb{Z}$. □

Figure 3.10 shows two steps of application of these refinement relationships.

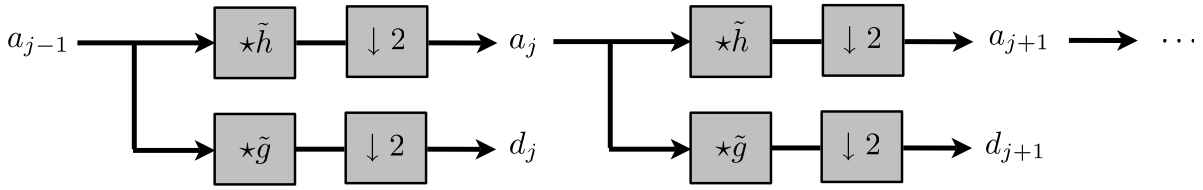

Figure 3.10: Forward filter bank decomposition.

The FWT thus operates as follow:

– Input: signal $f \in \mathbb{C}^N$.

– Initialization: $a_J = f$.

– For $j = J, \ldots, j_0 - 1$.

$$a_{j+1} = (a_j \star \tilde{h}) \downarrow 2 \quad \text{and} \quad d_{j+1} = (a_j \star \tilde{g}) \downarrow 2$$

– Output: the coefficients $\{d_j\}_{j_0\leqslant j<J} \cup \{a_{j_0}\}$.

If $|h|, |g| \leqslant C$ so that both filter are compactly supported, then computing each $\mathcal{W}_j$ is $(2C)2^{-j}$ operation, so that the complexity of the whole wavelet transform is

$$\sum_{j=J}^{1} (2C)2^{-j} = (2C)2^{-J} = 2CN.$$

This shows that the fast wavelet transform is a linear time algorithm. Figure 3.11 shows the process of extracting iteratively the wavelet coefficients. Figure 3.12 shows an example of computation, where at each iteration, the coefficients of a_j and d_j are added to the left of the output vector.

Fast Haar transform. For the Haar wavelets, one has

$$\varphi_{j,n} = \frac{1}{\sqrt{2}}(\varphi_{j-1,2n} + \varphi_{j-1,2n+1}),$$

$$\psi_{j,n} = \frac{1}{\sqrt{2}}(\varphi_{j-1,2n} - \varphi_{j-1,2n+1}).$$

This corresponds to the filters

$$h = [\ldots,\ 0,\ h[0] = \frac{1}{\sqrt{2}},\ \frac{1}{\sqrt{2}},\ 0, \ldots],$$

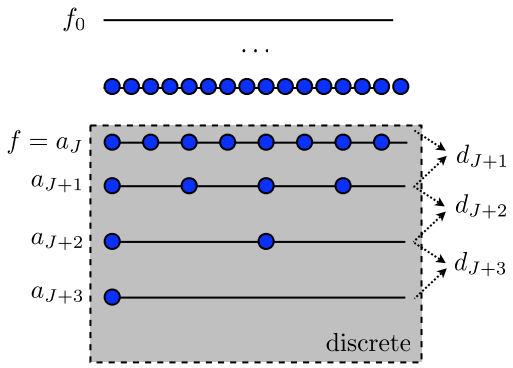

Figure 3.11: Pyramid computation of the coefficients.

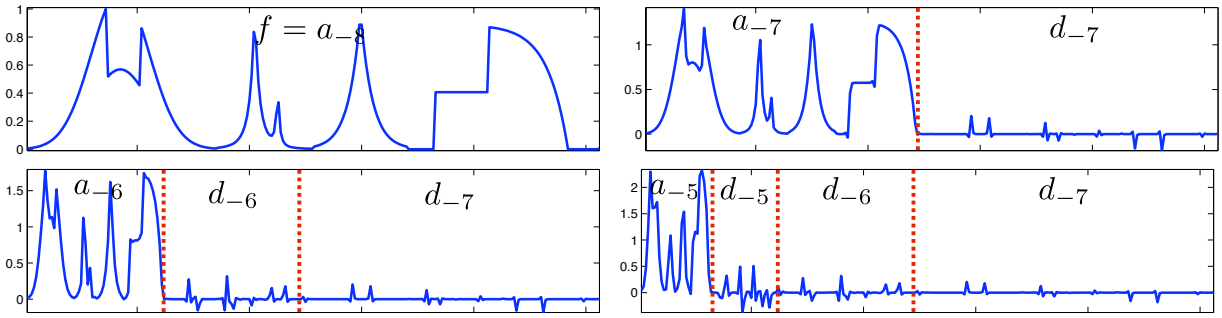

Figure 3.12: Wavelet decomposition algorithm.

$$g = [\ldots,\ 0,\ h[0] = \frac{1}{\sqrt{2}},\ -\frac{1}{\sqrt{2}},\ 0, \ldots].$$

The Haar wavelet transform algorithm thus processes by iterating averaging and differences:
– **Input:** signal $f \in \mathbb{C}^N$.
– **Initialization:** $a_J = f$.
– **For** $j = J, \ldots, j_0 - 1$.

$$a_{j+1,n} = \frac{1}{\sqrt{2}}(a_{j-1,2n} + a_{j-1,2n+1}) \quad \text{and} \quad d_{j+1,n} = \frac{1}{\sqrt{2}}(a_{j-1,2n} - a_{j-1,2n+1}).$$

– **Output:** the coefficients $\{d_j\}_{j_0 \leqslant j < J} \cup \{a_{j_0}\}$.

3.4.3 Inverse Fast Transform (iFWT)

The inverse algorithm proceeds by inverting each step (3.11)

$$\forall\, j = 0, \ldots, J+1, \quad a_{j-1} = \mathcal{W}_j^{-1}(a_j, d_j) = \mathcal{W}_j^*(a_j, d_j), \tag{3.11}$$

where $\mathcal{W}_j^*$ is the adjoint for the canonical inner product on $\mathbb{R}^{2^{-j+1}}$, i.e. when viewed as a matrix, the transpose.

We denote $\uparrow_2 : \mathbb{R}^{K/2} \to \mathbb{R}^K$ the up-sampling operator

$$a \uparrow_2 = (a_0, 0, a_1, 0, \ldots, 0, a_{K/2}, 0) \in \mathbb{R}^K.$$

Proposition 13. *One has*

$$\mathcal{W}_j^{-1}(a_j, d_j) = (a_j \uparrow_2) \star h + (d_j \uparrow_2) \star g.$$

Proof. Since $\mathcal{W}_j$ is orthogonal, $\mathcal{W}_j^{-1} = \mathcal{W}_j^*$. We write the whole transform as

$$\mathcal{W}_j = S_2 \circ C_{\bar{h},\bar{g}} \circ \mathcal{D} \quad \text{where} \quad \begin{cases} \mathcal{D}(a) = (a, a), \\ C_{\bar{h},\bar{g}}(a, b) = (\bar{h} \star a, \bar{g} \star a), \\ S_2(a, b) = (a \downarrow_2, b \downarrow_2). \end{cases}$$

One has the following adjoint operator

$$\mathcal{D}^*(a, b) = a + b, \quad C_{\bar{h},\bar{g}}^*(a, b) = (h \star a, g \star b), \quad \text{and} \quad S_2(a, b) = (a \uparrow_2, b \uparrow_2).$$

Indeed, let us check this for the convolution, assuming involved sequences are in ℓ_1 (they are actually finite sums when considering periodic signals),

$$\langle f \star \bar{h},\, g \rangle = \sum_n (f \star \bar{h})_n g_n = \sum_n \sum_k f_k h_{k-n} g_n = \sum_k f_k \sum_n h_{k-n} g_n = \langle f,\, h \star g \rangle,$$

for the copying

$$\langle \mathcal{D}(a),\, (u, v) \rangle = \langle a,\, u \rangle + \langle a,\, v \rangle = \langle a,\, u + v \rangle = \langle a,\, \mathcal{D}^*(u, v) \rangle,$$

and for the down-sampling

$$\langle f \downarrow_2,\, g \rangle = \sum_n f_{2n} g_n = \sum_n (f_{2n} g_n + f_{2n+1} 0) = \langle f,\, g \uparrow_2 \rangle.$$

This is shown using matrix notations in Figure 3.13. Putting everything together gives the desired formula. □

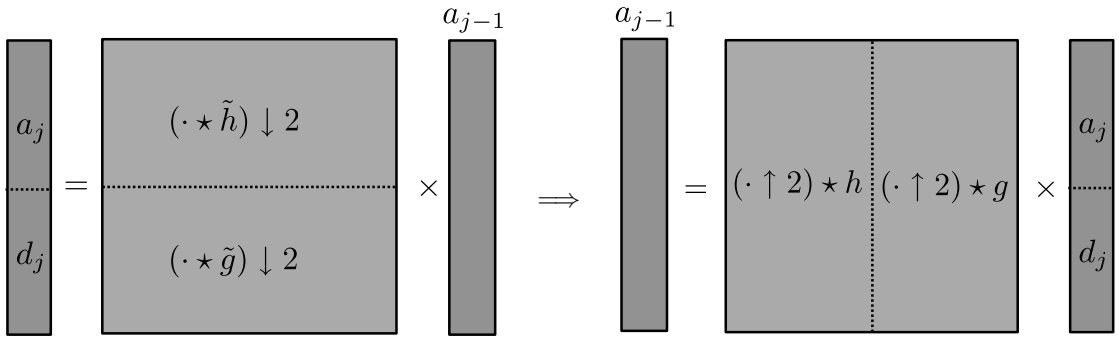

Figure 3.13: Wavelet inversion in matrix format.

The inverse Fast wavelet transform iteratively applies this elementary step

– Input: $\{d_j\}_{j_0 \leqslant j < J} \cup \{a_{j_0}\}$.

– For $j = j_0, \ldots, J + 1$.

$$a_{j-1} = (a_j \uparrow 2) \star h + (d_j \uparrow 2) \star g.$$

– Output: $f = a_J$.

This process is shown using a block diagram in Figure 3.14, which is the inverse of the block diagram 3.10.

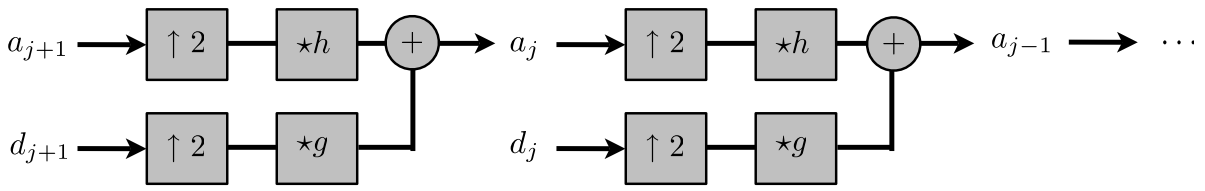

Figure 3.14: Backward filterbank recomposition algorithm.

3.5 2-D Wavelets

3.5.1 Anisotropic Wavelets

2-D anisotropic wavelets are defined using tensor product of Wavelet basis functions (2.12). The basis over $\mathbb{T}^2$ is thus of the form

$$\left\{\psi_{(j_1,j_2),(n_1,n_2)}\ ;\ (j_1,j_2)\in\mathbb{Z}^2, 0\leqslant n_1<2^{-j_1}, 0\leqslant n_2<2^{-j_2}\right\}$$

$$\text{where}\quad \psi_{(j_1,j_2),(n_1,n_2)}(x_1,x_2)=\psi_{j_1,n_1}(x_1)\psi_{j_2,n_2}(x_2). \tag{3.12}$$

The computation of the fast anisotropic Wavelet transform in 2-D is similar to the 2-D FFT detailed in Section 2.5.2. Viewing the input image $a_J\in\mathbb{R}^{2^{-J}\times 2^{-J}}$ as a matrix, one first apply the 1-D FWT to each row, and then to each column, resulting in a linear time $O(N)$ algorithm, where $N=2^{-2J}$.

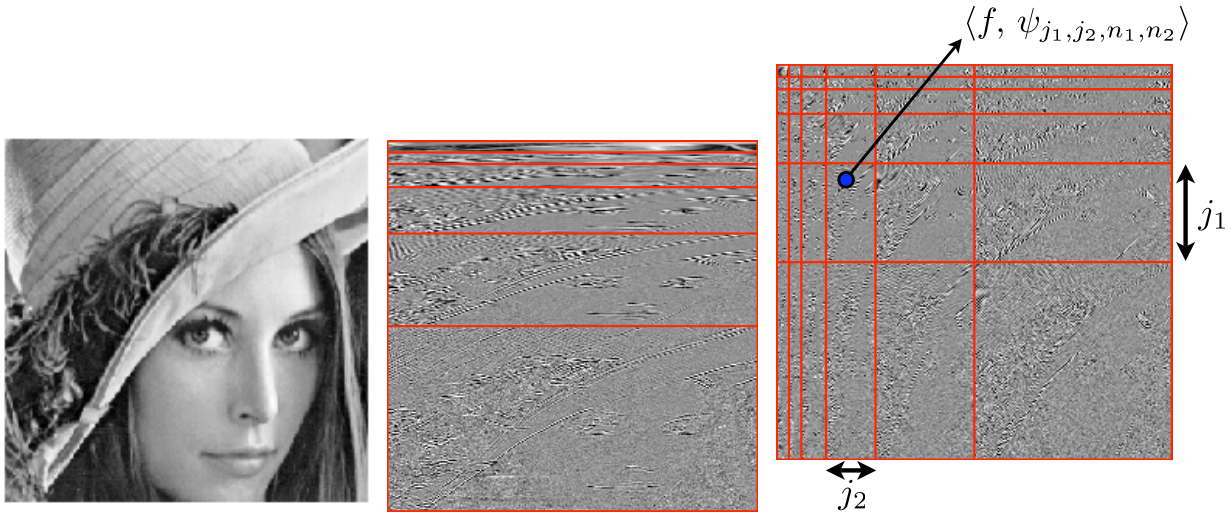

Image f Row transform Column transform.

Figure 3.15: Steps of the anisotropic wavelet transform.

3.5.2 Isotropic Wavelets

A major issue with these anisotropic wavelet (3.12) is that a function $\psi_{(j_1,j_2),(n_1,n_2)}$ is scaled independently in each direction, leads to functions concentrated along an axis-oriented rectangle of size $2^{-j_1}\times 2^{j_2}$. This is not a very good features (since natural images usually do not exhibit such an anisotropy) and typically leads to visually unpleasant artifacts when used for processing (denoising or compression).

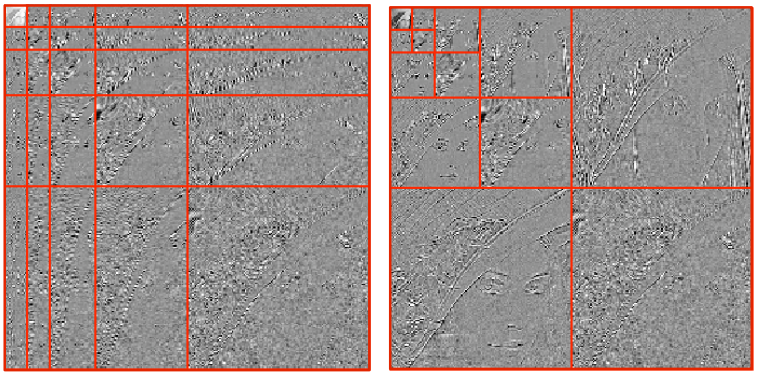

Figure 3.16: Anisotropic (left) versus isotropic (right) wavelet coefficients.

One rather use isotropic wavelet obtained by considering 2-D multi-resolution, obtained by tensor products of the 1-D approximation spaces

$$L^2(\mathbb{R}^2) \supset \ldots \supset V^{j-1} \otimes V^{j-1} \supset V_j \otimes V_j \supset V_{j+1} \otimes V_{j+1} \supset \ldots \supset \{0\}.$$

In the following, we denote

$$V_j^O \stackrel{\text{def.}}{=} V_j \otimes V_j$$

this isotropic 2-D multiresolution space.

Recall that the tensor product of two space $(V_1, V_2) \in L^2(\mathbb{R})^2$ is

$$V_1 \otimes V_2 = \text{Closure}\left(\text{Span}\left\{f_1(x_1)f_2(x_2) \in L^2(\mathbb{R}^2)\ ;\ f_1 \in V_1, f_2 \in V_2\right\}\right).$$

If $(\varphi_k^s)_k$ are Hilbertian bases for V_s, then one can show that $(\varphi_k^1(x_1)\varphi_k^2(x_2))_k$ is an Hilbertian basis for $V_1 \otimes V_2$.

One easily verify that one has the distributivity

$$(V_j \oplus^\perp W_J) \otimes (V_j \oplus^\perp W_J) = V_j^O \oplus^\perp W_j^V \oplus^\perp W_j^V \oplus^\perp W_j^V \quad \text{where} \quad \left\{ \begin{array}{l} W_j^V \stackrel{\text{def.}}{=} (V_j \otimes W_j), \\ W_j^H \stackrel{\text{def.}}{=} (W_j \otimes V_j), \\ W_j^D \stackrel{\text{def.}}{=} (W_j \otimes W_j). \end{array} \right.$$

Here the letters $\{V, H, D\}$ stands for *Vertical, Horizontal, Diagonal* detail spaces. This leads to the following diagram of embedded spaces

$$\begin{array}{ccccccccccc} \mathrm{L}^2(\mathbb{R}^2) & \rightarrow & \cdots & \rightarrow & V_{j-1} \otimes V_{j-1} & \rightarrow & V_j \otimes V_j & \rightarrow & V_{j+1} \otimes V_{j+1} & \rightarrow & \cdots \rightarrow \{0\} \\ & & & \searrow & W_{j-1}^{(2)} & \searrow & W_j^{(2)} & \searrow & W_{j+1}^{(2)} & \searrow & \end{array}$$

For $j \in \mathbb{Z}$, each of the three wavelet spaces is spanned with a wavelet, where basis elements are indexed by $n = (n_1, n_2) \in \mathbb{Z}$ (or in $\{0, \ldots, 2^{-j} - 1\}^2$ on the interval $\mathbb{T}$),

$$\forall\, \omega \in \{V, H, D\}, \quad W_j^\omega = \text{Span}\{\psi_{j,n_1,n_2}^\omega\}_{n_1,n_2}$$

where

$$\forall\, \omega \in \{V, H, D\}, \quad \psi_{j,n_1,n_2}^\omega(x) = \frac{1}{2^j}\psi^\omega\left(\frac{x_1 - 2^j n_1}{2^j}, \frac{x_2 - 2^j n_2}{2^j}\right)$$

and where the three mother wavelets are

$$\psi^H(x) = \psi(x_1)\varphi(x_2), \quad \psi^V(x) = \varphi(x_1)\psi(x_2), \quad \text{and} \quad \psi^D(x) = \psi(x_1)\psi(x_2).$$

Figure 3.17 displays an examples of these wavelets.

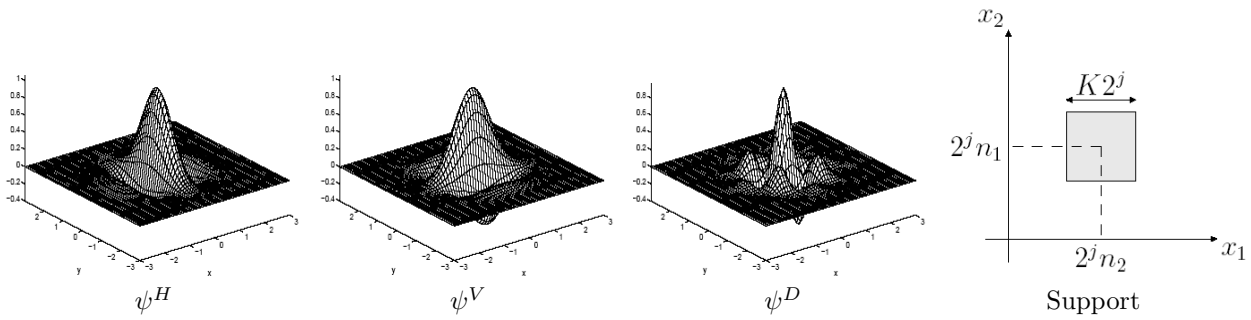

Figure 3.17: 2-D wavelets and their approximative support (right).

Haar 2-D multiresolution. For the Haar multiresolution, one obtains 2-D piecewise-constant Haar approximation. A function of $V_j \otimes V_j$ is constant on squares of size $2^j \times 2^j$. Figure 3.18 shows an example of projection of an image onto these 2-D Haar approximation spaces.

Figure 3.18: 2-D Haar approximation $P_{V_j^O} f$ for increasing j.

Discrete 2-D wavelet coefficients. Similarly to (3.5), we suppose that the sampling mechanism gives us access to inner product of the analog (continuous) signal f with the scaling function at scale $N = 2^{-J}$

$$\forall\, n, \in \{0, \ldots, N-1\}^2, \quad a_{J,n} = \langle f,\, \varphi^P_{J,n} \rangle$$

Discrete wavelet coefficients are defined as

$$\forall\, \omega \in \{V, H, D\},\ \forall\, J < j \leqslant 0,\ \forall\, 0 \leqslant n_1, n_2 < 2^{-j}, \quad d^\omega_{j,n} = \langle f,\, \psi^\omega_{j,n} \rangle.$$

(we use here periodized wavelets). Approximation coefficients are defined as

$$a_{j,n} = \langle f_0,\, \varphi^O_{j,n} \rangle.$$

Figure 3.19 shows examples of wavelet coefficients, that are packed in an image of N pixels. Figure 3.20 shows other examples of wavelet decompositions.

Forward 2-D wavelet transform basic step. A basic step of the computation of the 2-D wavelet transform computes detail coefficients and a low pass residual from the fine scale coefficients

$$a_{j-1} \longmapsto (a_j, d^H_j, d^V_j, d^D_j).$$

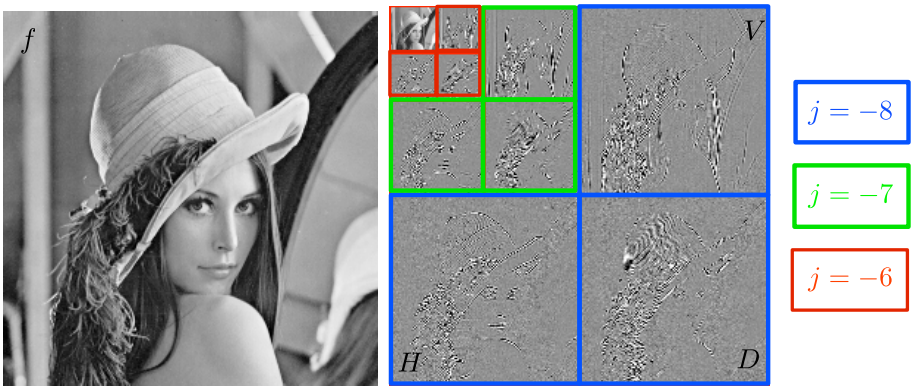

Figure 3.19: 2-D wavelet coefficients.

Similarly to the 1-D setting, this mapping is orthogonal, and is computed using the 1-D filtering and sub-sampling formula (3.8).

One first applies 1-D horizontal filtering and sub-sampling

$$\tilde{a}_j = (a_{j-1} \star^H \tilde{h}) \downarrow^H 2$$
$$\tilde{d}_j = (a_{j-1} \star^H \tilde{h}) \downarrow^H 2,$$

where $\star^H$ is the horizontal convolution, that applies the 1-D convolution to each column of a matrix

$$a \star^H b_{n_1,n_2} = \sum_{m_1=0}^{P-1} a_{n_1-m_1,n_2} b_{m_1}$$

where $a \in \mathbb{C}^{P \times P}$ and $b \in \mathbb{C}^P$ are matrix and vectors. The notation $\downarrow^H 2$ accounts for sub-sampling in the horizontal direction

$$(a \downarrow^H 2)_{n_1,n_2} = a_{2n_1,n_2}.$$

One then applies 1-D vertical filtering and sub-sampling to $\tilde{a}_j$ and $\tilde{d}_j$ to obtain

$$\begin{aligned} a_j &= (\tilde{a}_j \star^V \tilde{h}) \downarrow^V 2, & d_j^H &= (\tilde{d}_j \star^V \tilde{h}) \downarrow^V 2, \\ d_j^V &= (\tilde{a}_j \star^V \tilde{g}) \downarrow^V 2, & d_j^D &= (\tilde{d}_j \star^V \tilde{g}) \downarrow^V 2, \end{aligned}$$

where the vertical operators are defined similarly to horizontal operators but operating on rows.

These two forward steps are shown in block diagram in Figure 3.21. These steps can be applied in place, so that the coefficients are stored in an image of N pixels, as shown in Figure 3.22. This gives the traditional display of wavelet coefficients used in Figure 3.20.

Fast 2-D wavelet transform. The 2-D FWT algorithm iterates these steps through the scales:

– Input: signal $f \in \mathbb{C}^N$.

– Initialization: $a_J = f$.

– For $j = J, \ldots, j_0 - 1$.

$$\begin{aligned} \tilde{a}_j &= (a_{j-1} \star^H \tilde{h}) \downarrow^H 2, & d_j^V &= (\tilde{a}_j \star^V \tilde{g}) \downarrow^V 2, \\ \tilde{d}_j &= (a_{j-1} \star^H \tilde{h}) \downarrow^H 2, & d_j^H &= (\tilde{d}_j \star^V \tilde{h}) \downarrow^V 2, \\ a_j &= (\tilde{a}_j \star^V \tilde{h}) \downarrow^V 2, & d_j^D &= (\tilde{d}_j \star^V \tilde{g}) \downarrow^V 2. \end{aligned}$$

– Output: the coefficients $\{d_j^\omega\}_{j_0 \leqslant j < J, \omega} \cup \{a_{j_0}\}$.

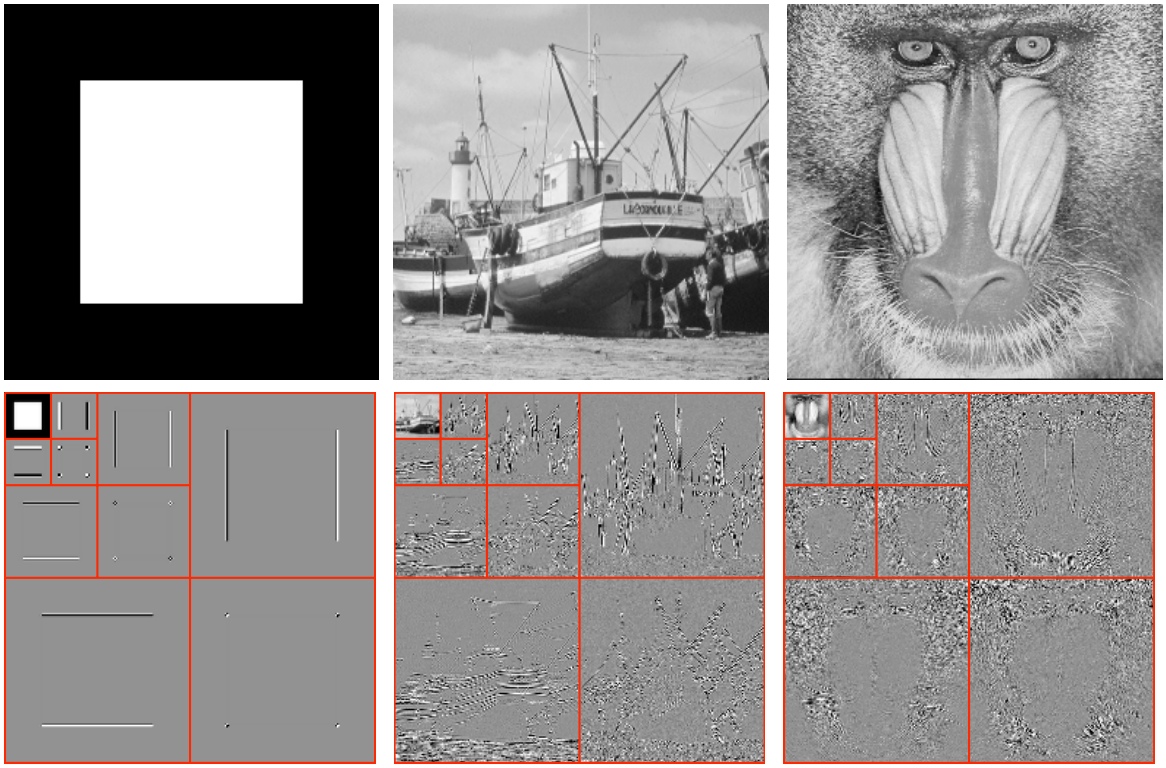

Figure 3.20: Examples of images (top row) and the corresponding wavelet coefficients (bottom row) .

Fast 2-D inverse wavelet transform. The inverse transform undo the horizontal and vertical filtering steps. The first step computes

$$\tilde{a}_j = (a_j \star^V h) \uparrow^V 2 + (d_j^V \star^V g) \uparrow^V 2,$$
$$\tilde{d}_j = (d_j^H \star^V h) \uparrow^V 2 + (d_j^D \star^V g) \uparrow^V 2,$$

where the vertical up-sampling is

$$(a \uparrow^V 2)_{n_1,n_2} = \begin{cases} a_{k,n_2} & \text{if} \quad n_1 = 2k, \\ 0 \quad \text{if} & n = 2k+1. \end{cases}$$

The second inverse step computes

$$a_{j-1} = (\tilde{a}_j \star^H h) \uparrow^H 2 + (\tilde{d}_j \star^H g) \uparrow^H 2.$$

Figure 3.23 shows in block diagram this inverse filter banks, that is the inverse of the diagram 3.21.

The inverse Fast wavelet transform iteratively applies these elementary steps

– Input: $\{d_j^\omega\}_{j_0 \leqslant j < J,\omega} \cup \{a_{j_0}\}$.

– For $j = j_0, \ldots, J+1$.

$$\tilde{a}_j = (a_j \star^V h) \uparrow^V 2 + (d_j^V \star^V g) \uparrow^V 2,$$
$$\tilde{d}_j = (d_j^H \star^V h) \uparrow^V 2 + (d_j^D \star^V g) \uparrow^V 2,$$
$$a_{j-1} = (\tilde{a}_j \star^H h) \uparrow^V 2 + (\tilde{d}_j \star^H g) \uparrow^V 2.$$

– Output: $f = a_J$.

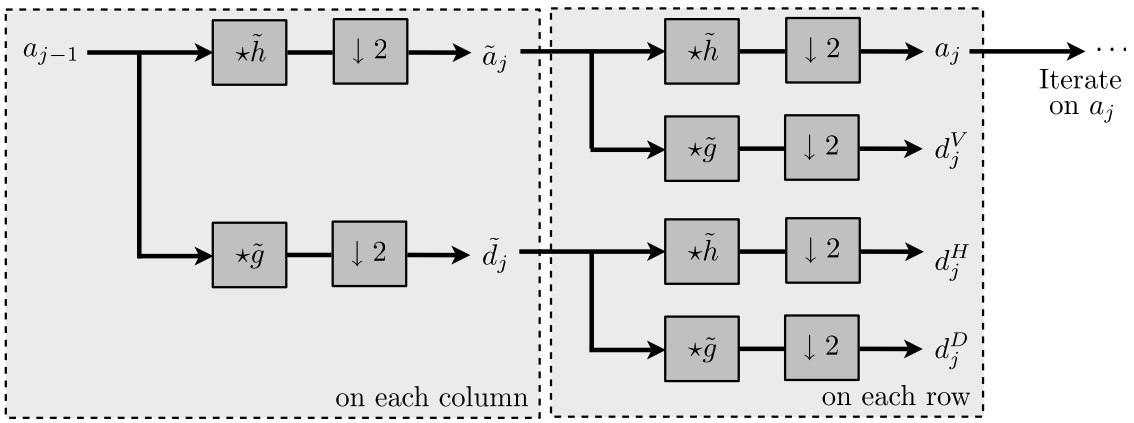

Figure 3.21: Forward 2-D filterbank step.

Figure 3.22: One step of the 2-D wavelet transform algorithm.

3.6 Wavelet Design

To be able to compute the wavelet coefficients using the FWT algorithm, it remains to know how to compute the scaling and wavelet functions. The FWT only makes use of the filters h and g, so instead of explicitly knowing the functions φ and ψ, one can only know these filters. Indeed, most of the known wavelets do not have explicit formula, and are implicitly defined through the cascade of the FWT algorithm.

This section shows what are the constraints h and g should satisfy, and gives practical examples. Furthermore, it shows that the knowledge of h determines g under the constraint of having quadrature filters, which is the most usual choice for wavelet analysis.

3.6.1 Low-pass Filter Constraints

We introduce the following three conditions on a filter h

$$\hat{h}(0) = \sqrt{2} \tag{C_1}$$

$$|\hat{h}(\omega)|^2 + |\hat{h}(\omega+\pi)|^2 = 2, \tag{C_2}$$

$$\inf_{\omega\in[-\pi/2,\pi/2]} |\hat{h}(\omega)| > 0. \tag{C^*}$$

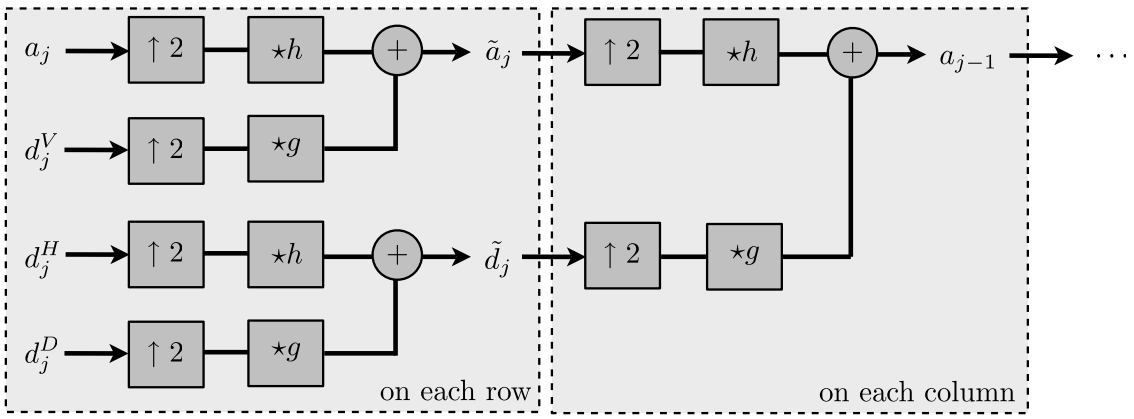

Figure 3.23: Backward 2-D filterbank step.

Here we are using the Fourier series associated to a filter $h \in \mathbb{R}^{\mathbb{Z}}$

$$\forall \omega \in \mathbb{R}/2\pi\mathbb{Z}, \quad \hat{h}(\omega) \stackrel{\text{def.}}{=} \sum_{n\in\mathbb{Z}} h_n e^{-\mathrm{i}n\omega}.$$

If $h \in \ell^1(\mathbb{Z})$, this defines a continuous periodic fonction $\hat{h} \in \mathcal{C}^0(\mathbb{R}/2\pi\mathbb{Z})$, and this definition can be extended to $h \in \ell^2(\mathbb{Z})$ and defines $\hat{h} \in L^2(\mathbb{R}/2\pi\mathbb{Z})$.

Theorem 3. *If φ defines multi-resolution approximation spaces, then (C_1) and (C_2) holds for h defined in (3.7). Conversely, if (C_1), (C_2) and (C^*) holds, then there exists a φ defining multi-resolution approximation spaces so that associated filter is h as defined in (3.7).*

Proof. We only prove the first statement of the theorem. The converse statement is much more difficult to prove.

We now prove condition (C_1). The refinement equation reads like a discrete-continuous convolution (or equivalently a convolution with a distribution)

$$\frac{1}{\sqrt{2}}\varphi\left(\frac{t}{2}\right) = \sum_{n\in\mathbb{Z}} h_n \varphi(t-n). \tag{3.13}$$

Denoting $h \star \varphi$ such a convolution, assuming $h \in \ell_1(\mathbb{Z})$ and $\varphi \in L^1(\mathbb{R})$, one check that one can apply Fubini and that $h \star \varphi \in L^1(\mathbb{R})$ and then

$$\begin{aligned}\mathcal{F}(\sum_{n\in\mathbb{Z}} h_n\varphi(t-n))(\omega) &= \int_{\mathbb{R}} \sum_{n\in\mathbb{Z}} h_n\varphi(t-n)e^{-\mathrm{i}\omega t}\mathrm{d}t = \sum_{n\in\mathbb{Z}} h_n \int_{\mathbb{R}} \varphi(t-n)e^{-\mathrm{i}\omega t}\mathrm{d}t \\ &= \sum_{n\in\mathbb{Z}} h_n e^{-\mathrm{i}n\omega} \int_{\mathbb{R}} \varphi(x)e^{-\mathrm{i}\omega x}\mathrm{d}x = \hat{\varphi}(\omega)\hat{h}(\omega)\end{aligned}$$

where we made the change of variable $x = t-n$. Note that here, $\hat{h}(\omega)$ is the 2π-periodic Fourier transform (i.e. Fourier series) of infinite filters defined in (3.6.1), whereas $\hat{\varphi}(\omega)$ is the Fourier transform of function. This is thus a product of a 2π-periodic function $\hat{h}$ and a non-periodic function $\hat{\varphi}$. We recall that $\mathcal{F}(f(\cdot/s)) = s\hat{f}(s\cdot)$. Over the Fourier domain, equation (3.13) thus reads

$$\hat{\varphi}(2\omega) = \frac{1}{\sqrt{2}}\hat{h}(\omega)\hat{\varphi}(\omega). \tag{3.14}$$

One can show that $\hat{\varphi}(0) \neq 0$ (actually, $|\hat{\varphi}(0)| = 1$), so that this relation implies the first condition (C_1).

We now prove condition (C_2). The orthogonality of $\varphi(\cdot - n)\}_n$ is rewritten using a continuous convolution as (see also Proposition 11)

$$\forall\, n \in \mathbb{Z}, \quad \varphi \star \bar{\varphi}(n) = \delta_0$$

where $\bar{\varphi}(x) = \varphi(-x)$, and thus over the Fourier domain, using (3.14) which shows $\hat{\varphi}(\omega) = \frac{1}{\sqrt{2}}\hat{h}(\omega/2)\hat{\varphi}(\omega/2)$

$$1 = \sum_k |\hat{\varphi}(\omega + 2k\pi)|^2 = \frac{1}{2}\sum_k |\hat{h}(\omega/2 + k\pi)|^2 |\hat{\varphi}(\omega/2 + k\pi)|^2.$$

Since $\hat{h}$ is 2π-periodic, one can split even and odd k and obtain

$$2 = |\hat{h}(\omega/2)|^2 \sum_k |\hat{\varphi}(\omega/2 + 2k\pi)|^2 + |\hat{h}(\omega/2 + \pi)|^2 \sum_k |\hat{\varphi}(\omega/2 + 2k\pi + \pi)|^2$$

This leads to condition (C_2). Re-using the fact that $\sum_k |\hat{\varphi}(\omega + 2k\pi)|^2 = 1$ for $\omega' = \omega/2$ in place of ω, one thus has

$$|\hat{h}(\omega')|^2 + |\hat{h}(\omega' + \pi)|^2 = 2.$$

We do not prove the converse statement, which requires to "create" a function φ from the filter h. The intuition is that iterating (3.14) leads informally to

$$\hat{\varphi}(\omega) = \prod_{k<0} \frac{\hat{h}(\omega/2^k)}{\sqrt{2}}. \tag{3.15}$$

Condition (C^*) can be shown to imply that this infinite product converge, and define a (non-periodic) function in $L^2(\mathbb{R})$. □

Note that for the converse statement of this theorem to holds, condition (C^*) imposes a control on the behavior of $\hat{h}$ near 0.

3.6.2 High-pass Filter Constraints

We now introduce the following two conditions on a pair of filter (g, h)

$$|\hat{g}(\omega)|^2 + |\hat{g}(\omega + \pi)|^2 = 2 \tag{C_3}$$

$$\hat{g}(\omega)\hat{h}(\omega)^* + \hat{g}(\omega + \pi)\hat{h}(\omega + \pi)^* = 0. \tag{C_4}$$

Figure 3.24 illustrates this constraint.

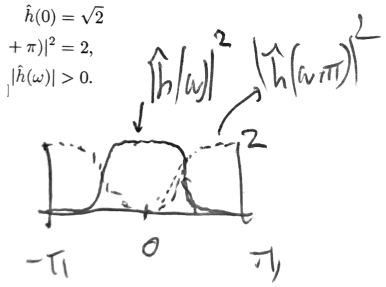

Figure 3.24: Low/high pass filter constraint.

Theorem 4. *If (φ, ψ) defines a multi-resolution analysis, then (C_3) and (C_4) holds for (h, g) defined in (3.7). Conversely, if (C_1) to (C_4) hold, then there exists a (φ, ψ) defining multi-resolution analysis so that associated filters are (h, g) as defined in (3.7). Furthermore,*

$$\hat{\psi}(\omega) = \frac{1}{\sqrt{2}} \hat{g}(\omega/2)\hat{\varphi}(\omega/2). \tag{3.16}$$

Proof. We prove condition (C_3). The refinement equation for the wavelet reads

$$\frac{1}{\sqrt{2}}\psi\left(\frac{t}{2}\right) = \sum_{n\in\mathbb{Z}} g_n \varphi(t-n)$$

and thus over the Fourier domain

$$\hat{\psi}(2\omega) = \frac{1}{\sqrt{2}} \hat{g}(\omega)\hat{\varphi}(\omega). \tag{3.17}$$

The orthogonality of $\{\psi(\cdot - n)\}_n$ is re-written

$$\forall\, n \in \mathbb{Z}, \quad \psi \star \bar{\psi}(n) = \delta_0$$

and thus over the Fourier domain (using Poisson formula, see also Proposition 11)

$$\sum_k |\hat{\psi}(\omega + 2k\pi)|^2 = 1.$$

Using the Fourier domain refinement equation (3.17), similarely to the proof of Theorem 3 for (C_1), this is equivalent to condition (C_3). Figure 3.25 shows the Fourier transform of two filters that satisfy this complementary condition.

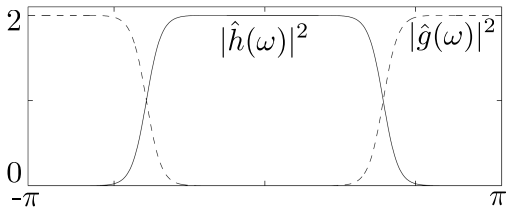

Figure 3.25: Complementarity between a low pass and a high pass wavelet filters h and g that satisfy condition (C_3).

We now prove condition (C_4). The orthogonality between $\{\psi(\cdot - n)\}_n$ and $\{\varphi(\cdot - n)\}_n$ is written as

$$\forall\, n \in \mathbb{Z}, \quad \psi \star \bar{\varphi}(n) = 0$$

and hence over the Fourier domain (using Poisson formula, similarly to Proposition 11)

$$\sum_k \hat{\psi}(\omega + 2k\pi)\hat{\varphi}^*(\omega + 2k\pi) = 0.$$

Using the Fourier domain refinement equations (3.14) and (3.17), this is equivalent to condition (C_4). □

Quadrature mirror filters. Quadrature mirror filters (QMF) defines g as a function of h so that the conditions of Theorem 4 are automatically satisfy. This choice is the natural choice to build wavelet filters, and is implicitly assumed in most constructions (other choices leading to the same wavelet function anyway, since it safisfies (3.16)).

Proposition 14. *For a filter $h \in \ell^2(\mathbb{Z})$ satisfying (C_1), defining $g \in \ell^2(\mathbb{Z})$ as*

$$\forall\, n \in \mathbb{Z}, \quad g_n = (-1)^{1-n} h_{1-n} \tag{3.18}$$

satisfies conditions (C_3) and (C_4).

Proof. One indeed has that

$$\hat{g}(\omega) = e^{-i\omega} \hat{h}(\omega + \pi)^*, \tag{3.19}$$

so that

$$|\hat{g}(\omega)|^2 + |\hat{g}(\omega+\pi)|^2 = |e^{-i\omega}\hat{h}(\omega+\pi)^*|^2 + |e^{-i(\omega+\pi)}\hat{h}(\omega+2\pi)^*|^2 = |\hat{h}(\omega+\pi)|^2 + |\hat{h}(\omega+2\pi)|^2 = 2.$$

where we used the fact that $\hat{h}$ is 2π-periodic, and also

$$\begin{aligned}\hat{g}(\omega)\hat{h}(\omega)^* + \hat{g}(\omega+\pi)\hat{h}(\omega+\pi)^* &= e^{-i\omega}\hat{h}(\omega+\pi)^*\hat{h}(\omega)^* + e^{-i(\omega+\pi)}\hat{h}(\omega+2\pi)^*\hat{h}(\omega+\pi)^* \\ &= (e^{-i\omega} + e^{-i(\omega+\pi)})\hat{h}(\omega+\pi)^*\hat{h}(\omega)^* = 0.\end{aligned}$$

□

3.6.3 Wavelet Design Constraints

According to the previous sections, the construction of a multi-resolution analysis (i.e. of functions (φ, ψ) is obtained by designing a filter h satisfying conditions (C_1) and (C_2). The function φ is obtained by an infinite cascade of filtering, or equivalently in the Fourier domain by (3.15), there is in general (put aside special case such as the Haar multiresolution) no closed form expression for φ. Once φ is defined, ψ is automatically defined by the relation (3.16) (and g can be defined as (3.19)).

There exists only one Fourier transform, but there is a large choice of different mother wavelet functions ψ. They are characterized by

– Size of the support.
– Number of oscillations (the so called number p of vanishing moments).
– Symmetry (only possible for non-orthogonal bases).
– Smoothness (number of derivatives).

We now detail how these constraints are integrated together with conditions (C_1)- (C_4).

Vanishing moments. A wavelet ψ has p vanishing moments if

$$\forall\, k \leqslant p-1, \quad \int_{\mathbb{R}} \psi(x)\, x^k \mathrm{d}x = 0. \tag{3.20}$$

This ensures that $\langle f, \psi_{j,n}\rangle$ is small if f is C^α, $\alpha < p$ on $\mathrm{Supp}(\psi_{j,n})$, which can be seen by doing a Taylor expansion of f around the point $2^j n$ on the support of the wavelet. This condition is equivalently expressed over Fourier as followed (see Fig. 3.26).

Proposition 15. *Assuming enough regularity of ψ, and using the QMF construction (3.19), it has p vanishing moments if and only if*

$$\forall\, k \leqslant p-1, \quad \frac{\mathrm{d}^k \hat{h}}{\mathrm{d}\omega^k}(\pi) = \frac{\mathrm{d}^k \hat{g}}{\mathrm{d}\omega^k}(0) = 0. \tag{3.21}$$

Proof. Since ψ is regular, one has that $\hat{\psi}^{(k)} = \mathcal{F}((-\mathrm{i}\cdot)^k \psi(\cdot))$, so that

$$(-\mathrm{i})^k \int_{\mathbb{R}} x^k \psi(x) \mathrm{d}x = \hat{\psi}^{(k)}(0).$$

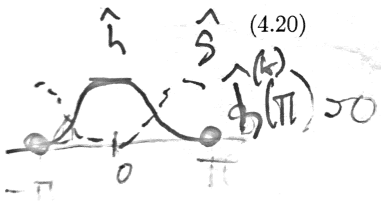

Figure 3.26: Wanishing moment condition as zero derivative conditions.

Relation (3.16) and (3.19) implies

$$\hat{\psi}(2\omega) = \hat{h}(\omega+\pi)^*\rho(\omega) \quad \text{where} \quad \rho(\omega) \stackrel{\text{def.}}{=} \frac{1}{\sqrt{2}} e^{-i\omega}\hat{\varphi}(\omega).$$

and differentiating this relation shows

$$2\hat{\psi}^{(1)}(2\omega) = \hat{h}(\omega+\pi)^*\rho^{(1)}(\omega) + \hat{h}^{(1)}(\omega+\pi)^*\rho(\omega)$$

which shows that, since $\rho(0) = \hat{\varphi}(0) \neq 0$, $\psi^{(1)}(0) = 0 \Leftrightarrow \hat{h}^{(1)}(\pi)^* = 0$. Recursing this argument and iterating the derrivative, one obtains that $\psi^{(k)}(0) = 0 \Leftrightarrow \hat{h}^{(k)}(\pi)^* = 0$ (assuming this hold for previous derivatives). □

Note that conditions (C_1) and (C_1) implies that $\hat{h}(\pi) = 0$, so that an admissible wavelet necessarily has 1 vanishing moment, i.e. $\int \psi = 0$. Condition (3.21) shows that having more vanishing moment is equivalent to having a Fourier transform $\hat{h}$ which is "flatter" arround $\omega = \pi$.

Support. Figure 3.27 shows the wavelet coefficients of a piecewise smooth signal. Coefficients of large magnitude are clustered near the singularities, because the wavelet ψ has enough vanishing moments.

To avoid that many wavelets create large coefficients near singularities, one should choose ψ with a small support. One can show that the size of the support of ψ is proportional to the size of the support of h. This requirement is however contradictory with the vanishing moment property (3.20). Indeed, one can prove that for an orthogonal wavelet basis with p vanishing moments

$$|\text{Supp}(\psi)| \geqslant 2p-1,$$

where $\sup(a)$ is the largest closed interval outside of which the function f is zero.

Chapter 4 studies in details the tradeoff of support size and vanishing moment to perform non-linear approximation of piecewise smooth signals.

Smoothness. In compression or denoising applications, an approximate signals is recovered from a partial set I_M of coefficients,

$$f_M = \sum_{(j,n)\in I_M} \langle f, \psi_{j,n}\rangle \psi_{j,n}.$$

This approximation f_M has the same smoothness as ψ.

To avoid visually unpleasant artifacts, one should thus choose a smooth wavelet function ψ. This is only for cosmetic reasons, since increasing smoothness does not leads to a better approximation. However, for most wavelet family, increasing the number of vanishing moments also increases the smoothness of the wavelets. This is for instance the case of the Daubechies family exposed in the next section.

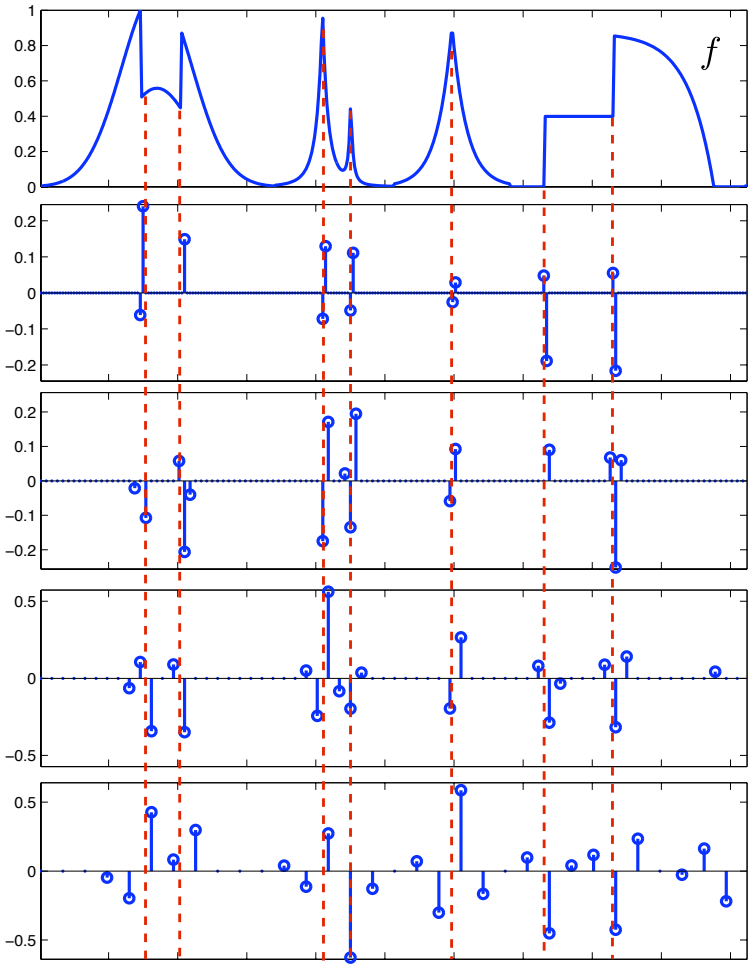

Figure 3.27: Location of large wavelet coefficients.

3.6.4 Daubechies Wavelets

To build a wavelet ψ with a fixed number p of vanishing moments, one designs the filter h, and use the quadrature mirror filter relation (3.19) to compute g. One thus look for h such that

$$|\hat{h}(\omega)|^2 + |\hat{h}(\omega+\pi)|^2 = 2, \quad \hat{h}(0) = \sqrt{2}, \quad \text{and} \quad \forall\, k < p, \ \frac{\mathrm{d}^k \hat{h}}{\mathrm{d}\omega^k}(\pi) = 0.$$

This corresponds to algebraic relationships between the coefficients of h, and it turns out that they can be solved explicitly using the Euclidean division algorithm for polynomials.

This leads to Daubechies wavelets with p vanishing moments, which are orthogonal wavelets with a minimum support length of $2p-1$.

For $p = 1$, it leads to the Haar wavelet, with

$$h = [h_0 = 0.7071; 0.7071].$$

For $p = 2$, one obtains the celebrated Daubechies 4 filter

$$h = [0.4830; h_0 = 0.8365; 0.2241; -0.1294],$$

and for $p = 3$,

$$h = [0; 0.3327; 0.8069; h_0 = 0.4599; -0.1350; -0.0854; 0.0352].$$

Wavelet display. Figure 3.28 shows examples of Daubechies mother wavelet functions with an increasing number of vanishing moments. These displays are obtained by computing in fact a discrete wavelet $\bar{\psi}_{j,n}$ defined in (**??**) for a very large number of samples N. This discrete wavelet is computed by applying the inverse wavelet transform to the coefficients $d_{j',n'} = \delta_{j-j'}\delta_{n-n'}$.

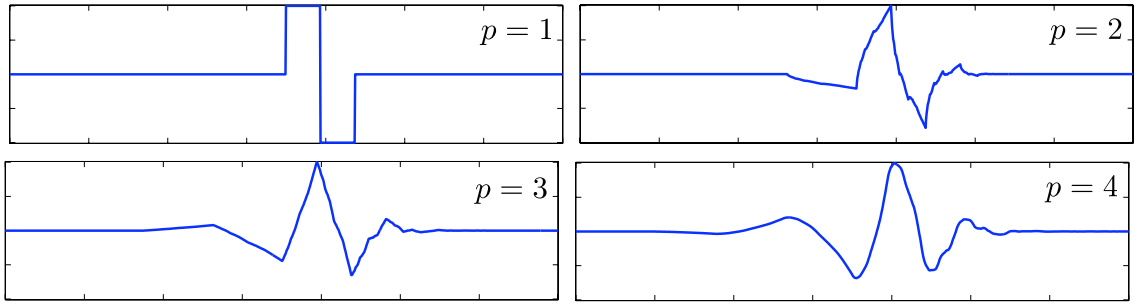

Figure 3.28: Examples of Daubechies mother wavelets ψ with an increasing number p of vanishing moments.

Bibliography

[1] Amir Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*. SIAM, 2014.

[2] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[3] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] E. Candès and D. Donoho. New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities. *Commun. on Pure and Appl. Math.*, 57(2):219–266, 2004.

[5] E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying. Fast discrete curvelet transforms. *SIAM Multiscale Modeling and Simulation*, 5:861–899, 2005.

[6] A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.*, 20:89–97, 2004.

[7] Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9(263-340):227, 2010.

[8] Antonin Chambolle and Thomas Pock. An introduction to continuous optimization for imaging. *Acta Numerica*, 25:161–319, 2016.

[9] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1999.

[10] Philippe G Ciarlet. Introduction à l'analyse numérique matricielle et à l'optimisation. 1982.

[11] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *SIAM Multiscale Modeling and Simulation*, 4(4), 2005.

[12] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. on Pure and Appl. Math.*, 57:1413–1541, 2004.

[13] D. Donoho and I. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, Dec 1994.

[14] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.

[15] M. Figueiredo and R. Nowak. An EM Algorithm for Wavelet-Based Image Restoration. *IEEE Trans. Image Proc.*, 12(8):906–916, 2003.

[16] Simon Foucart and Holger Rauhut. *A mathematical introduction to compressive sensing*, volume 1. Birkhäuser Basel, 2013.

[17] Stephane Mallat. *A wavelet tour of signal processing: the sparse way*. Academic press, 2008.

[18] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Commun. on Pure and Appl. Math.*, 42:577–685, 1989.

[19] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239, 2014.

[20] Gabriel Peyré. *L'algèbre discrète de la transformée de Fourier*. Ellipses, 2004.

[21] J. Portilla, V. Strela, M.J. Wainwright, and Simoncelli E.P. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Proc.*, 12(11):1338–1351, November 2003.

[22] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268, 1992.

[23] Otmar Scherzer, Markus Grasmair, Harald Grossauer, Markus Haltmeier, Frank Lenzen, and L Sirovich. *Variational methods in imaging*. Springer, 2009.

[24] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[25] Jean-Luc Starck, Fionn Murtagh, and Jalal Fadili. *Sparse image and signal processing: Wavelets and related geometric multiscale analysis*. Cambridge university press, 2015.