CHAPTER V

INTRODUCTION TO WAVELETS

Wavelets have generated tremendous interest in both theoretical and applied areas, especially within the latter half of this decade. Wavelet theory can be viewed as a synthesis of ideas originating in engineering (sub-band coding in part with quadrature mirror filters), physics (coherent states, renormalization group), and pure and applied mathematics (Calderón-Zygmund operators) [Dau92]. Historically, fundamental mathematical concepts of wavelet theory can be traced back to Fourier's work of 1807, Haar's algorithmic structure of 1909, with most of the development more directly related to wavelet theory occurring in the 1930s and 1960s [Mey93]. Wavelet theory has recently received wide acclaim due to the amalgamation of the diverse yet related analytical techniques into one elegant, coherent framework. Numerous researchers contributed to this effort. Grossmann, Morlet, Daubechies, Meyer, and Chui [Dau88, Mey93, Chu92] have greatly influenced the development of the mathematical theory, while Mallat and Meyer are credited with the introduction of multiresolution analysis in the wavelet context [Mal89a, Mal89b, Mey93]. Most of the analytical techniques developed herein for image/video processing and eye movement modeling are based on Mallat's multiresolution results, as they relate to pyramidal image processing techniques pioneered by Burt [Bur81, BA83b], and Mallat et al.'s singularity detection theory in the wavelet domain [MH91, MH92, MZ92a]. This section starts with the review of fundamental concepts of wavelet theory closely following Chui's derivations, then presents the theory of multiresolution analysis, wavelet filters and the discrete wavelet transform, and concludes with three applications of wavelet analysis, namely:

1. multiscale sharp variation (edge) detection in spatiotemporal data,
2. anisotropic multidimensional discrete wavelet analysis, and
3. multiresolution image representation through MIP mapping.

5.1 Fundamentals

The central idea behind wavelet analysis is the use of compactly supported basis functions which are used to approximate arbitrary signals. In essence, a wavelet basis is a generalized, functional extension of a vector basis. This section examines the fundamental concept of a basis and its use in the expression of arbitrary vectors and functions. The idea of a basis is studied in the domains of linear algebra, Fourier series, and wavelet series. Conventions for mathematical expressions are shown in Table 2.

TABLE 2
Notational conventions.

$c_1, c_2, \ldots, c_n$	Constants, scalars.
$\mathbf{v}$	Vector.
$(v_1, v_2, \ldots, v_n)$	Vector components.
$\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$	Vector set.
$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{k=1}^{n} u_k v_k = u_1 v_1 + \cdots + u_n v_n$	Inner product of vectors $\mathbf{u}, \mathbf{v}$.
$\lVert \mathbf{v} \rVert = \langle \mathbf{v}, \mathbf{v} \rangle^{1/2} = \sqrt{v_1^2 + \cdots + v_n^2}$	Vector norm.
V_n	Vector space (n-dimensional), i.e., set of all n-dimensional vectors.
$\mathbf{Z}$	Set of integers.
$\mathbf{R}$	Set of real numbers.
$\delta_{j,l} = \begin{cases} 1 & \text{for } j = l; \\ 0 & \text{for } j \neq l, \quad j, l \in \mathbf{Z} \end{cases}$	Kronecker delta.
E^n	Euclidean (n-dimensional) space.
$L^2(0, 2\pi)$	Vector space of 2π-periodic square-integrable one-dimensional functions $f(x)$.
$L^2(\mathbf{R})$	Vector space of measurable, square-integrable one-dimensional functions $f(x)$.
$L^2(\mathbf{R}^2)$	Vector space of measurable, square-integrable two-dimensional functions $f(x,y)$.
$\langle f, g \rangle = \int_{-\infty}^{\infty} f(x) \overline{g(x)}\, dx$	Inner product of $f, g \in L^2(\mathbf{R})$.
$\langle f, g \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \overline{g(x,y)}\, dx\, dy$	Inner product of $f, g \in L^2(\mathbf{R}^2)$.
$\lVert f \rVert = \langle f, f \rangle^{1/2}$	Norm of $f \in L^2(\mathbf{R})$ or $f \in L^2(\mathbf{R}^2)$, where the corresponding inner product is assumed.
f^j, g^l	Representations of functions f, g at j^{th} and l^{th} levels of resolution, respectively.
$\{h_k\}, \{g_m\}$	Digital filter sequences.
H, G	Digital filters.
$\mathbf{H}, \mathbf{G}$	Matrices (usually representing multidimensional digital filters).

5.1.1 Linear Algebra and Vector Spaces

Recall, from linear algebra, a vector $\mathbf{v}$ in n-dimensional Euclidean space E^n is defined as an ordered n-tuple $(v_1, v_2, \ldots, v_n)$ of real numbers, recognized as *vector components*. Vectors are *linearly independent* when the equation

$$\sum_{k=1}^{n} c_k \mathbf{v}_k = 0, \quad k \in \mathbf{Z}$$

can only hold if $c_1 = c_2 = \cdots = c_n = 0$. A vector *basis* in n-dimensional space is defined as any set of linearly independent vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$. Linear independence guarantees that any vector $\mathbf{v}$ in n-space can be expressed uniquely as a linear combination of the basis vectors, i.e.,

$$\begin{aligned} \mathbf{v} &= c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n \\ &= \sum_{k=1}^{n} c_k \mathbf{v}_k, \quad k \in \mathbf{Z}. \end{aligned}$$

The vectors are *orthogonal* (perpendicular) if

$$\langle \mathbf{v}_l, \mathbf{v}_m \rangle = \sum_{k=1}^{n} \mathbf{v}_{l_k} \mathbf{v}_{m_k} = v_{l_1} v_{m_1} + v_{l_2} v_{m_2} + \cdots + v_{l_n} v_{m_n} = 0, \quad \forall l, m,\ l \neq m,\ k, l, m \in \mathbf{Z},$$

where $\langle \mathbf{v}_l, \mathbf{v}_m \rangle$ denotes the vector *inner* (or scalar or dot) product. Every orthogonal system of n vectors forms a basis for the set of all vectors in n-space, V^n, although the orthogonality condition is not strictly necessary.[1] The orthogonal system of n vectors is *orthonormal* if each of the vectors has unit norm,

$$\|\mathbf{v}_k\| = \langle \mathbf{v}_k, \mathbf{v}_k \rangle^{1/2} = 1.$$

The theory of a vector space of infinite dimension is closely related to the theories of a function space, Fourier and wavelet series. The binding thread among these theories is the expression of an arbitrary function $f(x)$ by a series expansion using a set of basis functions $\{\psi_k(x)\}$ such that $f(x) = \sum_k c_k \psi_k(x)$.

5.1.2 Function Spaces

Wavelet theory is concerned with a particular functional vector space, namely the space $L^2(\mathbf{R})$ of all real, (Lebesgue) measurable, square integrable functions defined on the real line $\mathbf{R}$. The space $L^2(\mathbf{R})$ is a vector (Hilbert) space in which wavelet functions are typically constructed to serve as basis functions. Due to its pertinence to wavelet theory, the definition and properties of $L^2(\mathbf{R})$ are briefly discussed here. The present outline closely follows the very readable text by Holland [Hol90] where the reader is referred for further clarifications. References to particular sections and pages are given where appropriate.

---

[1] Given n linearly independent vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, it is always possible to construct an orthogonal system of n vectors $\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$, each of which is a linear combination of $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ (Gram-Schmidt orthogonalization process) [Kap84, p.55].

The terms *functional vector space, vector space of functions*, and *function space*, refer to a set of functions possessing the same formal properties as a vector space of n-tuples in linear algebra, i.e., closure under sum and closure under scalar multiplication. That is, a set V of functions forms a vector space if for any functions f,g in V, $f+g$, and cf are also in V. The set W is a (functional) subspace of V if W is a vector space in its own right [Hol90, pp.22-23]. Note that the linear algebra concepts of linear dependence and independence carry over seamlessly to functional vector spaces. The function space V is called an *inner product space* if a scalar-valued expression called the *inner product*, denoted $\langle f,g\rangle$, can be defined for all $f,g \in V$, satisfying the following three conditions:

1. linear in the first variable, conjugate linear in the second (conjugate bilinear):

$$\langle af+bg,h\rangle = a\langle f,h\rangle + b\langle g,h\rangle, \text{ and } \langle f,ag+bh\rangle = \overline{a}\langle f,g\rangle + \overline{b}\langle f,h\rangle; \tag{5.1}$$

2. Hermitian symmetric:

$$\langle f,g\rangle = \overline{\langle g,f\rangle} \;\; \forall f,g \in V; \tag{5.2}$$

3. positive definite:

$$\langle f,f\rangle \geq 0 \;\; \forall f \in V, \text{ and } \; \langle f,f\rangle = 0 \text{ implies } f=0, \tag{5.3}$$

where the symbol ($\bar{\cdot}$) denotes complex conjugation. In the case of real-valued functions and real scalars, complex conjugation has no effect, i.e., $\overline{a}=a$ for all scalars a. In this case, condition (5.1) states that the inner product is linear in each variable separately, condition (5.2) states that $\langle f,g\rangle = \langle g,f\rangle$, $\forall f,g \in V$, and condition (5.3) states that f,g are "essentially" equal if, in the sense of the inner product, $\langle f,g\rangle = 0$, and f is "essentially" zero if $\langle f,f\rangle = 0$. The last condition is somewhat subtle in that for $f(x)$ to be "essentially" zero (zero "almost everywhere") does not necessarily mean that $f(x)$ has to be zero at every point, only that whatever is used to define the inner product (e.g., an integral) evaluates to zero [Hol90, §3.6].

The above conditions define the abstract scalar value of an inner product axiomatically. In the case of the real-valued functional space $L^2(\mathbf{R})$, the inner product is specified as:

$$\langle f,g\rangle = \int_{-\infty}^{\infty} f(x)\,\overline{g(x)}\,dx. \tag{5.4}$$

Noting that conjugation has no effect on real-valued functions, the inner product satisfies the above properties (see [Hol90, pp.108-111]).

1. Bilinearity:

$$\begin{aligned}
\langle af+bg,h\rangle &= \int_{-\infty}^{\infty} (af+bg)h\,dx \\
&= \int_{-\infty}^{\infty} afh+bgh\,dx \\
&= a\int_{-\infty}^{\infty} fh\,dx + b\int_{-\infty}^{\infty} gh\,dx \\
&= a\langle f,h\rangle + b\langle g,h\rangle,
\end{aligned}$$

with a similar argument for the second variable.

2. Hermitian symmetry:

$$\langle f,g\rangle = \int_{-\infty}^{\infty} f\,g\,dx \;=\; \int_{-\infty}^{\infty} g\,f\,dx \;=\; \langle g,f\rangle.$$

3. Positive definiteness:

$$\langle f,f\rangle = \int_{-\infty}^{\infty} (f(x))^2\,dx \geq 0,$$

because $(f(x))^2 \geq 0$ since $f(x)$ is real, and

$$\langle f,f\rangle = \int_{-\infty}^{\infty} (f(x))^2\,dx \;=\; 0$$

implies $f(x)$ is “essentially” zero, meaning that $f(x)$ integrates to zero over the real line $\mathbf{R}$. The subtlety of this property can be illustrated by the function

$$f(x) = \begin{cases} 1 & \text{if } x = 0, \\ 0 & \text{otherwise,} \end{cases}$$

which has $\langle f,f\rangle = \int_{-\infty}^{\infty} |f(x)|^2\,dx = 0$ although $f(x)$ is not zero at every point. In fact, any function $f(x)$ that is zero except at a finite number of points has a zero integral.

The positive definite condition provides a definition of function equality: using the principle that absolute convergence implies convergence (so permitting the inspection of the integral of $|f(x)\overline{g(x)}|$ instead of $f(x)\overline{g(x)}$) and the fact that $|\overline{z}| = |z|$ for any complex number z, functions f and g are “essentially” equal if $\int_{-\infty}^{\infty} |f(x) - g(x)|\,dx = 0$ and f is “essentially” zero if $\int_{-\infty}^{\infty} |f(x)|^2\,dx = 0$.

Any function f is said to belong to the space $L^2(\mathbf{R})$ if it satisfies:

$$\int_{-\infty}^{\infty} |f(x)|^2\,dx < \infty, \tag{5.5}$$

where the integral is not required to have any particular value, only that it be finite [Hol90, pp.138-139].[2] The main issue in testing whether a given $f(x)$ belongs to $L^2(\mathbf{R})$ is to consider whether or not the integral in (5.5) is convergent or divergent. If it is not divergent, then $f(x)$ is said to be in $L^2(\mathbf{R})$. With the above definitions

[2]Technically, attention is usually restricted to Lebesgue measurable functions where all integrals should be interpreted as Lebesgue integrals. Under Lebesgue theory of integration, $f(x)$ is differentiable “almost everywhere” meaning that $f(x)$ is the Lebesgue integral of its derivative $f'(x)$, i.e.,

$$f(x) = \int_a^x f'(t)\,dt + f(a).$$

Such functions are called *absolutely continuous*. Although Lebesgue theory is beyond the scope of the present discussion (see [Hol90, p.165,p.253] for an introduction), it is tacitly assumed, without loss of generality, that all functions in $L^2(\mathbf{R})$ are absolutely continuous. This assumption only excludes functions where integration by parts may fail, i.e., functions of the Cantor-Lebesgue type, and practically restricts the discussion to functions that are piecewise continuous [Chu92, p.1].

of the space $L^2(\mathbf{R})$, and of the inner product for $L^2(\mathbf{R})$ in (5.4), it is clear that $L^2(\mathbf{R})$ is an inner product vector space (see [Hol90, pp.141-143]).

Since $L^2(\mathbf{R})$ is an inner product space, all properties associated with an inner product hold in $L^2(\mathbf{R})$. The properties particularly applicable to wavelet theory are listed below (see also [Chu92, p.4]):

- *Norm*:
$$\|f\| = \langle f, f\rangle^{1/2} = \left[\int_{-\infty}^{\infty} f(x)\overline{f(x)}\,dx\right]^{1/2} = \left[\int_{-\infty}^{\infty} |f(x)|^2\,dx\right]^{1/2}$$
- *Mean-square (distance) metric*:
$$\|f-g\| = \left[\int_{-\infty}^{\infty} |f(x)-g(x)|^2\,dx\right]^{1/2}$$
- *Schwarz' inequality*:
$$|\langle f,g\rangle| \leq \|f\|\,\|g\| \Rightarrow \left|\int_{-\infty}^{\infty} f(x)\overline{g(x)}\,dx\right| \leq \left[\int_{-\infty}^{\infty} |f(x)|^2\,dx\right]^{1/2}\left[\int_{-\infty}^{\infty} |g(x)|^2\,dx\right]^{1/2}$$

The above definition of $L^2(\mathbf{R})$ spaces naturally extends to the general class of inner product spaces, the L^2 spaces. L^2 spaces are denoted by
$$L^2(a,b), \quad -\infty \leq a < b \leq +\infty$$
where any function f is said to belong to the space $L^2(a,b)$ if it satisfies:
$$\int_a^b |f(x)|^2\,dx < \infty.$$
The inner product space $L^2(\mathbf{R})$ is a particular instance of the class of L^2 spaces with $a = -\infty$, $b = \infty$. The formal properties of a vector space and inner product, as well as the properties associated with the inner product, defined for $L^2(\mathbf{R})$ above, extend analogously to the general class of L^2 spaces. All L^2 spaces are Hilbert spaces since they satisfy the property of metric completeness [Hol90, p.143]. Although the notion of a Hilbert space is somewhat superfluous in the context of wavelet theory, some of the concepts and definitions of a Hilbert space do apply and are worth mentioning. In particular, the properties of *denseness*, *separability*, and *completeness*, required for the definition of the Hilbert space, are given below. The reader is referred to [Hol90, §3.10] for the complete account.

- *Denseness*: A subspace W of an abstract inner product space V is *dense* if, given any $v \in V$ there exists an $w \in W$ such that $\|v-w\| < \varepsilon$, for any small ε. This property states that W is dense in V if elements of W can be found as close as desired to any element of V.
- *Separability*: An inner product space V is *separable* if it contains a sequence of elements $w_1, w_2, \ldots$ that span a dense subspace of V. All finite-dimensional vector spaces V are separable since a basis can be found for V where the subspace spanned by the basis is V itself. Note that V is trivially dense in itself. All L^2 spaces are separable. As a consequence of this property, any separable inner product space has an orthogonal basis.

- *Completeness*: The notion of completeness is defined in terms of what is called a Cauchy sequence. A sequence of elements $v_k,\ k = 1,2,\ldots$ in an inner product space is a *Cauchy sequence* if: given any small positive number ε, an integer N (generally dependent on ε) can be found such that $\|v_k - v_l\| < \varepsilon$ whenever both $k,l \geq N$. Loosely speaking, a Cauchy sequence is one whose terms eventually cluster together. An inner product space V is said to be *complete* if, given any Cauchy sequence $v_k,\ k = 1,2,\ldots \in V$, there exists a $v \in V$ such that the sequence v_k converges to v. Roughly, the notion of completeness states that every Cauchy sequence in V must converge to an element of V.
- *Hilbert space*: The *Hilbert space* is a separable real (or complex) inner product space that is complete in the metric derived from its inner product.

Although concepts such as denseness and separability appear throughout the wavelet literature, arguably the most useful property (or at least notation) of wavelet theory is the idea of the inner product $\langle\cdot,\cdot\rangle$ of $L^2(\mathbf{R})$, and by analogous extension, of n-dimensional L^2 spaces denoted by $L^2(\mathbf{R}^n)$. This is due to the fact that wavelets are functions generating a basis in $L^2(\mathbf{R}^n)$ which, as explained below, is defined in terms of the inner product.

5.1.3 Fourier Series

Shifting from n-dimensional Euclidean space E^n to the space of 2π-periodic square-integrable functions $L^2(0,2\pi)$, a measurable function f is defined on the interval $(0,2\pi)$ as

$$\int_0^{2\pi} |f(x)|^2\,dx < \infty.$$

It may be assumed that f is a piecewise continuous function, extended periodically to the real line $\mathbf{R} = (-\infty,\infty)$ in $L^2(0,2\pi)$ by $f(x) = f(x-2\pi), \forall x$ [Chu92, p.1]. Any f in $L^2(0,2\pi)$ has a Fourier series expansion:

$$f(x) = \sum_{k=-\infty}^{\infty} c_k e^{ikx},$$

where $i = \sqrt{-1}$ and the Fourier coefficients c_k of f are defined by

$$c_k = \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{-ikx}\,dx.$$

Omitting the convergence considerations of Fourier series (detailed in [Kap84, Chu92]), the important feature to note here is that the function e^{ix} forms an orthonormal basis of $L^2(0,2\pi)$ which itself is a vector space. The original (2π-periodic square-integrable) function f is decomposed into (infinitely many) mutually orthogonal components $c_k e^{ikx}$ by the generation of the orthonormal basis $\{w_k\}$ from the *dilation* of the basis function $w(x) = e^{ix}$, i.e., $w_k(x) = w(kx)$, over all integers k.[3] Since the *sinusoidal wave* e^{ix} is the only function required to generate all 2π-periodic square-summable functions, every function in $L^2(0,2\pi)$ is composed of waves of various frequencies.

[3]The Fourier basis is often referred to as a basis of sines and cosines due to the identity: $e^{ix} = cos x + i sin x$.

5.1.4 Wavelet Series

Considering the space $L^2(\mathbf{R})$ of (Lebesgue) measurable functions f defined on the real line $\mathbf{R}$, again a single basis function is sought which can be made to express any function f in $L^2(\mathbf{R})$. The function space $L^2(\mathbf{R})$ differs from $L^2(0,2\pi)$ in that the local average values of every function must attenuate to zero at $\pm\infty$. The sinusoidal wave functions $w_k(x)$ do not belong to $L^2(\mathbf{R})$, and cannot be used directly to generate a basis in $L^2(\mathbf{R})$. Instead, "short-term" (quickly decaying) waves, known as *wavelets*, are required. In order to cover the entire space, these *compactly supported* wavelet functions must be shifted (translated) in space. The region where the function is nonzero is said to be its *support*, hence wavelets are nonzero in limited (compact) regions. The set of wavelet functions are formed by dilations and translations of a single function $\psi(x)$ called the "mother wavelet", "basic wavelet", or "analyzing wavelet" [RBC+92]. The term *wavelet* refers to wavelet functions of the form

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi(\frac{x-b}{a}), \quad a > 0, b \in \mathbf{R}, \tag{5.6}$$

where dilations and translations are governed by parameters a, b, respectively. Every wavelet ψ generates a series representation of $f \in L^2(\mathbf{R})$:

$$f(x) = \sum_{a,b=-\infty}^{\infty} c_{a,b}\psi_{a,b}, \quad a > 0, b \in \mathbf{R},$$

with wavelet coefficients $\{c_{a,b}\}$ given by the integral transform W_ψ:

$$\begin{aligned} c_{a,b} &= \{W_\psi f(x)\}(a,b) \\ &= \frac{1}{\sqrt{a}}\int_{-\infty}^{\infty} f(x)\overline{\psi(\frac{x-b}{a})}dx, \quad f \in L^2(\mathbf{R}),\ a > 0, b \in \mathbf{R}, \\ &= \langle f, \psi_{a,b}\rangle. \end{aligned} \tag{5.7}$$

The linear transformation W_ψ is called the *integral wavelet transform*, or simply *wavelet transform*, relative to ψ.[4]

For reasons concerning sampling theory and computational efficiency, the parameters a, b are chosen so that frequency space is partitioned into consecutive frequency bands (or "octaves") by a *binary* dilation, and space is covered by a *dyadic* translation, i.e.,

$$a = 2^{-j}; \ \ b = \frac{k}{2^j}, \quad j,k \in \mathbf{Z}.$$

[4] The resemblance of the wavelet function to the *ket* of quantum mechanics is not accidental. Both the wavelet and the ket are used to represent vector bases. The vector basis *bra*, associated with the ket, corresponds to the scaling function ϕ in the wavelet context, an essential component of multiresolution analysis, described in §5.4. Further similarities between the two domains include the inner product $\langle\cdot,\cdot\rangle$ which is used in place of the Dirac notation $\langle\cdot|\cdot\rangle$. The projection operator $P_\psi = |\psi\rangle\langle\psi|$ used in quantum mechanics is not expressly used in the wavelet literature although the wavelet transform itself defines the projection of an arbitrary function onto the wavelet basis generated by ψ. The tensor product of two vector bases defined in quantum mechanics as $|\phi\rangle \otimes |\psi\rangle$ is significant in the wavelet domain insofar as it is used to construct multidimensional wavelet bases (see §5.5,§5.7) [CDL77].

The *dyadic wavelet* can now be expressed in terms of dilation and translation parameters j,k,

$$\psi_{j,k}(x) \quad = \quad 2^{j/2}\psi(2^j x - k), \quad j,k \in \mathbf{Z}. \tag{5.8}$$

Henceforth expression (5.8) is used to define both *dyadic wavelets* and *wavelets* although it should be noted that a dyadic wavelet is technically distinguished from the basic wavelet, as defined in (5.6), not only by the binary dilation and dyadic translation, but also by a "stability condition" imposed on the basic wavelet (see [Chu92, p.11]).

The dyadic wavelet $\psi_{j,k}$ generates a dyadic series representation of $f \in L^2(\mathbf{R})$:

$$f(x) = \sum_{j,k=-\infty}^{\infty} c_{j,k}\psi_{j,k}, \quad j,k \in \mathbf{Z}, \tag{5.9}$$

with wavelet coefficients $\{c_{j,k}\}$ given by the integral transform:

$$\begin{aligned} c_{j,k} &= \{W_\psi f(x)\}(j,k) \\ &= 2^{j/2}\int_{-\infty}^{\infty} f(x)\overline{\psi(2^j x - k)}dx, \quad f \in L^2(\mathbf{R}),\ j,k \in \mathbf{Z}, \\ &= \langle f, \psi_{j,k}\rangle. \end{aligned} \tag{5.10}$$

That is, the $(j,k)^{th}$ wavelet coefficient of f is given by the integral wavelet transformation of f at dyadic position $b = k/2^j$ with binary dilation $a = 2^{-j}$. Provided the wavelet ψ is orthogonal, the same wavelet ψ is used to generate the wavelet series (5.9) and to define the integral wavelet transform (5.10) [Chu92, p.5]. Orthogonal and other classifications of wavelets are discussed in §5.2.

5.1.5 From Vectors to Wavelets

The common goal among the above three methodologies is the representation of an arbitrary function (or generalized vector) $f(x)$ by a linear combination of basis elements, i.e.,

$$f(x) = \sum_{k=-\infty}^{\infty} c_k\psi_k(x).$$

For vectors, a set of basis vectors is used, i.e., $\{\psi_k(x)\} = \{\mathbf{v}_k\}$. In the Fourier domain, the linear combination is formed from the frequency dilation of the basis function $\{\psi_k(x)\} = \{e^{ikx}\}$. In the wavelet domain, the linear combination is formed from the frequency dilation and spatial translation of the basis function $\{\psi_k(x)\} = \{\psi_{j,k}(x)\}$.[5] Note that for wavelets, j,k are the dilation and translation parameters, respectively,

[5]The precise representation of $f(x)$ is an l^2-linear combination [Chu92, p.3], where l^2 denotes the space of all square-summable bi-infinite sequences; that is, $\{c_k\} \in l^2$ if and only if

$$\sum_{k=-\infty}^{\infty} |c_k|^2 < \infty.$$

whereas in the the Fourier domain these parameters are reversed, that is, k is the frequency dilation parameter, and there is no explicit translation parameter, i.e., $j = 1$, since the Fourier basis functions are infinite in extent [GB92]. The dilation of the basis function in both cases generates the representation of $f(x)$ at multiple frequencies.

The primary distinction between the Fourier and wavelet representations is that the Fourier series uses one basis function at multiple frequencies over all space. The wavelet representation also uses one basis function at multiple frequencies, but to cover all space, it uses many shifted versions of the compactly supported basis function (the mother wavelet), each over a limited spatial region. That is, each wavelet is *localized* in space. Furthermore, the wavelet basis $\{\psi_{j,k}(x)\}$ analyzes a function over a consecutive distribution of frequency bands governed by the parameter j. This frequency distribution results in a hierarchical partitioning of the function by a flexible *space-frequency window* which automatically narrows at high frequencies and widens at low frequencies.[6] The dimensions of the window on the space-frequency grid are governed by j,k with constant area $4\Delta_x\Delta_\omega$, where Δ_x denotes the spatial extent and Δ_ω the frequency extent. The Heisenberg uncertainty principle states that the area of the space-frequency windows (also called *Heisenberg boxes*) can be no greater than 2. The automatic dilation of the wavelet Heisenberg boxes maintains constant area, but as the boxes shrink in space they stretch over frequency. This characteristic of the wavelet representation is known as the wavelets' *zooming property* and it is the wavelets' paramount advantage over traditional Fourier techniques for signal analysis. For further detail and precise definition of the wavelet space-frequency window see [JS94a, Chu92].

The benefit gained by the wavelets' flexible space-frequency window can be illustrated by the following abstract example of burst signal detection. Because the Fourier representation integrates the basis function over all space, i.e., over the entire signal, the frequency content is recorded over all space. If a high-frequency burst is present in the signal, Fourier analysis will only report that such a high frequency component is present *somewhere* in the signal. The wavelet representation, on the other hand, due to its hierarchical frequency partitioning, enables the detection *and* localization of transient signal components such as bursts. In an effort to provide similar functionality, short-term Fourier approaches (such as the Short-Time Fourier Transform, or STFT) use spatially localized basis functions, but still suffer from a fixed space-frequency window (see [Chu92, §1.2] for details). A schematic of the STFT and wavelet space-frequency tiling is shown in Figure 14.

[6] The space-frequency window is also known as the time-frequency window. There is no real distinction between space and time except within the context of the analysis. Typically time refers to time-varying signals such as speech, whereas space may refer to the spatial (x,y) location of an image pixel.

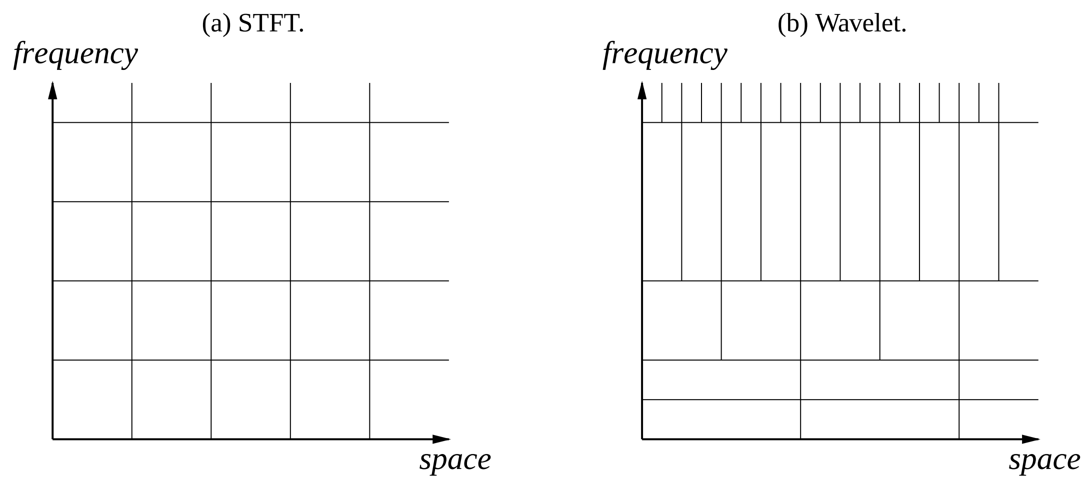


Fig. 14. Space-frequency tiling of the STFT and Wavelet representations. Adapted from [Bar94, p.9 (Fig. 2.1)].

5.2 Wavelet Functions

Denoting the closure of the linear span of $\{\psi_{j,k} : k \in \mathbf{Z}\}$ by W_j for each $j \in \mathbf{Z}$, i.e.,

$$W_j = clos_{L^2(\mathbf{R})}\langle \psi_{j,k} : k \in \mathbf{Z}\rangle,$$

$L^2(\mathbf{R})$ can be decomposed as a *direct sum* of the spaces W_j:

(5.11) $$L^2(\mathbf{R}) = \dot{\sum_{j\in\mathbf{Z}}} W_j = \cdots \dot{+} W_{-1} \dot{+} W_0 \dot{+} W_1 \dot{+} \cdots,$$

where $\dot{+}$ indicates "direct sum", in the sense that every function $f \in L^2(\mathbf{R})$ has a unique decomposition:

(5.12) $$f(x) = \cdots + g^{-1}(x) + g^0(x) + g^1(x) + \cdots,$$

where $g^j \in W_j$ and $g^l \in W_l$. Any wavelet generates a direct sum decomposition of $L^2(\mathbf{R})$.

5.2.1 Bi-orthogonal Wavelets

Every wavelet $\psi \in L^2(\mathbf{R})$, as defined by (5.8), has a *dual* $\tilde{\psi} \in L^2(\mathbf{R})$ defined by

$$\tilde{\psi}_{l,m}(x) = 2^{l/2}\tilde{\psi}(2^l x - m), \quad l,m \in \mathbf{Z}.$$

If the bases $\{\psi_{j,k}\}$ and $\{\tilde{\psi}_{l,m}\}$ generated by the dual wavelets ψ and $\tilde{\psi}$ satisfy

(5.13) $$\langle \psi_{j,k}, \tilde{\psi}_{l,m}\rangle = \delta_{j,l} \cdot \delta_{k,m}, \quad j,k,l,m \in \mathbf{Z},$$

i.e., the bases establish inter-scale ($\delta_{j,l}$) and intra-scale ($\delta_{k,m}$) orthogonality, then $(\psi, \tilde{\psi})$ form a pair of *bi-orthogonal wavelets*, and every $f \in L^2(\mathbf{R})$ can be written as a wavelet series

(5.14) $$f(x) \quad = \quad \sum_{j,k\in\mathbf{Z}} d_{j,k}\tilde{\psi}_{j,k}(x)$$

$$= \sum_{j,k \in \mathbf{Z}} d_{j,k} \psi_{j,k}(x), \tag{5.15}$$

where, analogous to Fourier coefficients, wavelet coefficients are given by

$$\begin{aligned} d_{j,k} &= \langle f, \psi_{j,k} \rangle \quad \text{in (5.14);} \\ &= \langle f, \widetilde{\psi}_{j,k} \rangle \quad \text{in (5.15).} \end{aligned}$$

If ψ and $\widetilde{\psi}$ constitute a bi-orthogonal wavelet pair, then they generate two subspaces $\{W_j\}, \{\widetilde{W}_j\}$ of $L^2(\mathbf{R})$ where the subspaces are not generally mutually orthogonal,

$$W_j \not\perp W_l, \text{ and } \widetilde{W}_j \not\perp \widetilde{W}_l, \quad j \neq l, \tag{5.16}$$

but instead are orthogonal in the dual sense,

$$W_j \perp \widetilde{W}_l, \quad j \neq l. \tag{5.17}$$

Since both $\{\psi_{j,k}\}$ and $\{\widetilde{\psi}_{l,m}\}$ are bases of $L^2(\mathbf{R})$, the space can be decomposed by either basis, i.e.,

$$\begin{aligned} L^2(\mathbf{R}) &= \dot{\sum_{j \in \mathbf{Z}}} W_j = \cdots \dot{+} W_{-1} \dot{+} W_0 \dot{+} W_1 \dot{+} \cdots \\ &= \dot{\sum_{j \in \mathbf{Z}}} \widetilde{W}_j = \cdots \dot{+} \widetilde{W}_{-1} \dot{+} \widetilde{W}_0 \dot{+} \widetilde{W}_1 \dot{+} \cdots. \end{aligned} \tag{5.18}$$

Equations (5.14) and (5.15) effectively state that any function in $L^2(\mathbf{R})$ projected onto one basis can be recovered by expansion in the other. In contrast to an orthogonal wavelet basis (see below), the bi-orthogonal system permits greater freedom in the construction of wavelet filters (see §5.6.4). For details pertaining to the convergence of the series, see [Chu92, p.5 and §3.6].

5.2.2 Orthogonal Wavelets

A function $\psi \in L^2(\mathbf{R})$ is called an *orthogonal wavelet* if the family $\{\psi_{j,k}\}$ forms an orthonormal basis of $L^2(\mathbf{R})$,

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = \delta_{j,l} \cdot \delta_{k,m}, \quad j,k,l,m \in \mathbf{Z}. \tag{5.19}$$

An orthogonal wavelet is self-dual with $\psi = \widetilde{\psi}$ generating $\{\psi_{j,k}\}$ so that every $f \in L^2(\mathbf{R})$ can be represented by the wavelet series

$$f(x) = \sum_{j,k \in \mathbf{Z}} d_{j,k} \psi_{j,k}(x),$$

with wavelet coefficients

$$d_{j,k} = \langle f, \psi_{j,k} \rangle.$$

Given an orthogonal wavelet ψ, the subspaces $\{W_j\}$ of $L^2(\mathbf{R})$ generated by the wavelet are mutually orthogonal,

$$W_j \perp W_l, \quad j \neq l. \tag{5.20}$$

The function decompositions, as given by (5.12), are also orthogonal, i.e.,

$$\langle g^j, g^l \rangle = 0, \quad j \neq l,$$

and the direct sum of subspaces (5.11) becomes an *orthogonal sum*:

$$L^2(\mathbf{R}) = \bigoplus_{j\in\mathbf{Z}} W_j = \cdots \oplus W_{-1} \oplus W_0 \oplus W_1 \oplus \cdots, \tag{5.21}$$

where $\oplus$ indicates "orthogonal sum" (see [Chu92, pp.14-15] for details).

5.2.3 Semi-orthogonal Wavelets

A function $\psi \in L^2(\mathbf{R})$ is called a *semi-orthogonal wavelet* if the generated basis $\{\psi_{j,k}\}$ satisfies

$$\langle \psi_{j,k}, \psi_{l,m} \rangle = 0, \quad j \neq l, \quad j,k,l,m \in \mathbf{Z}, \tag{5.22}$$

where $\langle \psi_{j,k}, \psi_{j,m} \rangle$ may be non-zero. That is, the wavelet does not necessarily provide intra-scale orthogonality. The distinction between orthogonal and semi-orthogonal wavelets is analogous to orthogonal and orthonormal vector bases. Semi-orthogonal wavelets, also known as "pre-wavelets", can produce orthogonal wavelets through an orthogonalization procedure [RBC+92, p.8]. Every semi-orthogonal wavelet generates an (inter-scale) orthogonal decomposition (but not necessarily an orthonormal one), and every orthogonal wavelet is also a semi-orthogonal wavelet since (5.19) guarantees (5.22).

Although semi-orthogonal wavelets are not generally fully orthonormal, the subspaces they generate are mutually orthogonal, i.e., the condition expressed by (5.20) holds, and the function decompositions, as given by (5.12), are also orthogonal, i.e.,

$$\langle g^j, g^l \rangle = 0, \quad j \neq l, \quad j,l \in \mathbf{Z}.$$

5.2.4 Non-orthogonal Wavelets

A wavelet ψ is called *non-orthogonal* if it is not a semi-orthogonal wavelet. Bi-orthogonal wavelets are generally non-orthogonal, meaning that the resulting bases typically lack both inter-scale and intra-scale orthogonality [RBC+92, p.8].

5.3 Wavelet Maxima and Multiscale Edges

A critical consideration in almost any signal analysis task is the detection of sharp variation points. The wavelet transform is closely related to multiscale edge detection employed in computer vision [MZ92a]. Mallat et al. have developed an adaptive sampling technique to locate signal sharp variation points by detecting local maxima of the wavelet transform modulus. The method is equivalent to the Canny edge detector [Can86]. Several papers and book chapters by Mallat can be found on this topic, including [MZ92a, MZ92b, MH92, FM92, Mal91]. Because this edge detection technique is directly applicable to eye movement modeling and video analysis it is summarized here, closely following Mallat's derivations. Where appropriate, the reference to the relevant source is provided.

The wavelet transform of f at scale j and position x, given in (5.10), defines the convolution product

$$\{W_\psi f(x)\}(j) = f * \psi_j(x),$$

where the translation parameter k is made implicit. The *dyadic wavelet transform* is defined as the sequence of functions

$$\mathbf{W} f = \left[\{W_\psi f(x)\}(j)\right]_{j \in \mathbf{Z}},$$

where $\mathbf{W}$ is the dyadic wavelet transform operator. Assuming a twice-differentiable smoothing function $\theta(x)$ exists, whose integral is equal to 1 and that converges to 0 at infinity, e.g., a Gaussian, define the first- and second-order derivatives of $\theta(x)$:

$$\psi'(x) = \frac{d\theta(x)}{dx}, \text{ and, } \psi''(x) = \frac{d^2\theta(x)}{dx^2}.$$

The functions ψ' and ψ'' are by definition wavelets since their integral is equal to 0. Denoting the wavelet transforms of $f(x)$ relative to ψ', ψ'' as,

$$\{W_{\psi'} f(x)\}(j) = f * \psi_j{}'(x), \text{ and } \{W_{\psi''} f(x)\}(j) = f * \psi_j{}''(x),$$

$\{W_{\psi'} f(x)\}(j)$, $\{W_{\psi''} f(x)\}(j)$ are the first and second derivative of the signal smoothed at scale j [MZ92a]:

$$\{W_{\psi'} f(x)\}(j) = f * (j\frac{d\theta_j}{dx})(x) = j\frac{d}{dx}(f * \theta_j)(x), \text{ and}$$
$$\{W_{\psi''} f(x)\}(j) = f * (j^2\frac{d^2\theta_j}{dx^2})(x) = j^2\frac{d^2}{dx^2}(f * \theta_j)(x).$$

The local extrema of $\{W_{\psi'} f(x)\}(j)$ correspond to the zero crossings of $\{W_{\psi''} f(x)\}(j)$ and to the inflection points of $f * \theta_j(x)$. In the particular case where $\theta(x)$ is a Gaussian, the zero-crossing detection is equivalent to a Marr-Hildreth edge detection [Mar80], and the extrema detection corresponds to Canny edge detection [Can86]. The wavelet approach follows the latter, relying on $\{W_{\psi'} f(x)\}(j)$ to distinguish between sharp and slow variation points of $f * \theta_j(x)$, which is often difficult using a second derivative operator.

Sharp variation points are detected by finding the local maxima of the modulus $|\{W_{\psi'} f(x)\}(j)|$. At each scale j, local modulus maxima are located by finding the points where $|\{W_{\psi'} f(x)\}(j)|$ is larger than its two closest neighbor values, and strictly larger than at least one of them [MH92]. That is, a modulus maxima is located at scale j and location (x_0) if:

$$|\{W_{\psi'} f(x_0-1)\}(j)| \leq |\{W_{\psi'} f(x_0)\}(j)| \geq |\{W_{\psi'} f(x_0+1)\}(j)|, \text{ and} \tag{5.23}$$

$$\begin{cases} |\{W_{\psi'} f(x_0)\}(j)| > |\{W_{\psi'} f(x_0-1)\}(j)|, & \text{or} \\ |\{W_{\psi'} f(x_0)\}(j)| > |\{W_{\psi'} f(x_0+1)\}(j)|. \end{cases} \tag{5.24}$$

The modulus maxima of the wavelet transform at scale j and location (x_0) is a strict local maxima of the modulus on the right or the left of location x_0.

The local maxima detection is extendible to multiple dimensions if there exists a smoothing function θ, which converges to 0 at infinity yet totally integrates to 1. In two dimensions, the image function $f(x,y)$ is smoothed at different scales j by convolution with the two-dimensional smoothing function $\theta_j(x,y)$. Computing the gradient vector, $\nabla(f*\theta_j)(x,y)$, edges are defined as points (x_0,y_0) where the modulus of the gradient vector is maximum in the direction of the gradient in the image plane. Introducing two 2D wavelet functions,

$$\psi_x(x,y) = \frac{\partial\theta(x,y)}{\partial x} \text{ and } \psi_y(x,y) = \frac{\partial\theta(x,y)}{\partial y},$$

two components of the wavelet transform of $f(x,y) \in L^2(\mathbf{R}^2)$ at scale j are defined with implicit translation parameter k:

$$\{Wf(x,y)\}_x(j) = \{W_{\psi_x} f(x,y)\}(j) = f * \psi_{x_j}(x,y), \text{ and}$$
$$\{Wf(x,y)\}_y(j) = \{W_{\psi_y} f(x,y)\}(j) = f * \psi_{y_j}(x,y).$$

Edge points can be located from the two components $\{Wf(x,y)\}_x(j)$, $\{Wf(x,y)\}_y(j)$ since

$$\begin{pmatrix} \{Wf(x,y)\}_x(j) \\ \{Wf(x,y)\}_y(j) \end{pmatrix} = j \begin{pmatrix} \frac{\partial}{\partial x}(f*\theta_j)(x,y) \\ \frac{\partial}{\partial y}(f*\theta_j)(x,y) \end{pmatrix} = j\nabla(f*\theta_j)(x,y).$$

Sharp variation points are detected analogously to the 1D case, where the modulus at scale j and position (x,y), denoted by $\{Mf(x,y)\}(j)$, is proportional to:

$$|\{Wf(x,y)\}(j)| \propto \{Mf(x,y)\}(j) = \sqrt{|\{Wf(x,y)\}_x(j)|^2 + |\{Wf(x,y)\}_y(j)|^2}. \tag{5.25}$$

At each scale j, the local modulus maxima are again found by comparing the point $\{Mf(x,y)\}(j)$ with its two neighbors as in (5.23) and (5.24), except now neighboring points must be examined along the direction of the gradient vector. The angle of the gradient vector with horizontal direction at scale j and position (x,y), denoted by $\{Af(x,y)\}(j)$ is given by [MZ92b]:

$$
\begin{aligned}
\{Af(x,y)\}(j) &= \arg(\{Wf(x,y)\}_x(j) + i\{Wf(x,y)\}_y(j)) \\
&= \tan^{-1}\left(\frac{\{Wf(x,y)\}_y(j)}{\{Wf(x,y)\}_x(j)}\right).
\end{aligned} \tag{5.26}
$$

In three dimensions, the volume function $f(x,y,t)$ is smoothed at different scales j by convolution with the three-dimensional smoothing function $\theta_j(x,y,t)$. Computing the gradient vector, $\nabla(f*\theta_j)(x,y,t)$, edges are defined as points (x_0,y_0,t_0) where the modulus of the gradient vector is maximum in the direction of the gradient in the volume. Introducing three 3D wavelet functions,

$$
\psi_x(x,y,t) = \frac{\partial\theta(x,y,t)}{\partial x},\ \psi_y(x,y,t) = \frac{\partial\theta(x,y,t)}{\partial y},\ \text{and}\ \psi_t(x,y,t) = \frac{\partial\theta(x,y,t)}{\partial t},
$$

three components of the wavelet transform of $f(x,y,t) \in L^2(\mathbf{R}^3)$ at scale j are defined with implicit translation parameter k:

$$
\begin{aligned}
&\{Wf(x,y,t)\}_x(j) = \{W_{\psi_x} f(x,y,t)\}(j) = f*\psi_{x\,j}(x,y,t), \\
&\{Wf(x,y,t)\}_y(j) = \{W_{\psi_y} f(x,y,t)\}(j) = f*\psi_{y\,j}(x,y,t),\ \text{and} \\
&\{Wf(x,y,t)\}_t(j) = \{W_{\psi_t} f(x,y,t)\}(j) = f*\psi_{t\,j}(x,y,t).
\end{aligned}
$$

Edge points can be located from the above three components since

$$
\begin{pmatrix} \{Wf(x,y,t)\}_x(j) \\ \{Wf(x,y,t)\}_y(j) \\ \{Wf(x,y,t)\}_t(j) \end{pmatrix} = j \begin{pmatrix} \frac{\partial}{\partial x}(f*\theta_j)(x,y,t) \\ \frac{\partial}{\partial y}(f*\theta_j)(x,y,t) \\ \frac{\partial}{\partial t}(f*\theta_j)(x,y,t) \end{pmatrix} = j\nabla(f*\theta_j)(x,y,t).
$$

Sharp variation points are detected analogously to the 1D case, where the modulus at scale j and position (x,y,t), denoted by $\{Mf(x,y,t)\}(j)$, is proportional to:

$$
\begin{aligned}
&|\{Wf(x,y,t)\}(j)| \propto \{Mf(x,y,t)\}(j) = \\
&\sqrt{|\{Wf(x,y,t)\}_x(j)|^2 + |\{Wf(x,y,t)\}_y(j)|^2 + |\{Wf(x,y,t)\}_t(j)|^2}.
\end{aligned} \tag{5.27}
$$

At each scale j, the local modulus maxima are again found by comparing the point $\{Mf(x,y,t)\}(j)$ with its two neighbors along the direction of the gradient vector. The angle of the gradient vector is now determined by three planar angles at scale j and position (x,y,t), denoted by $\{Af(x,y,t)\}_{\angle ab}(j)$ where $\angle ab$ specifies the directional plane, given by:

$$
\{Af(x,y,t)\}_{\angle xy}(j) = \tan^{-1}\left(\frac{\{Wf(x,y,t)\}_y(j)}{\{Wf(x,y,t)\}_x(j)}\right), \tag{5.28}
$$

$$
\{Af(x,y,t)\}_{\angle xt}(j) = \tan^{-1}\left(\frac{\{Wf(x,y,t)\}_t(j)}{\{Wf(x,y,t)\}_x(j)}\right), \tag{5.29}
$$

$$
\{Af(x,y,t)\}_{\angle yt}(j) = \tan^{-1}\left(\frac{\{Wf(x,y,t)\}_y(j)}{\{Wf(x,y,t)\}_t(j)}\right). \tag{5.30}
$$

5.4 Multiresolution Analysis

Multiresolution analysis (MRA), introduced by Meyer and Mallat [Mal89a], is an algorithmic framework for representing functions at hierarchical levels of scale (or resolution). The wavelet basis described above analyzes the underlying signal in terms of spatially-localized frequency components. Using the wavelet basis alone, reconstruction of the signal may be problematic. In order to reconstruct the original signal from its wavelet representation, the wavelet dual $\tilde{\psi}$ is used as the reconstruction kernel function in the inversion formula defined by Equation (5.14). In general, however, $\tilde{\psi}$ does not exist [Chu92, p.13]. Multiresolution analysis addresses this reconstruction problem by maintaining a scaled version of the signal at consecutive levels of resolution. The original signal can be faithfully reconstructed by successively combining the scaled signal with the wavelet coefficients at each level of resolution.

5.4.1 Scaling Functions

At the heart of multiresolution analysis is the notion of a *scaling function*, denoted by $\phi(x)$. The scaling function is very similar in nature to the wavelet in that it also generates a basis of $L^2(\mathbf{R})$. The scaling function is also a compactly supported function, defined as

$$\phi_{a,b}(x) = \frac{1}{\sqrt{a}}\phi(\frac{x-b}{a}), \quad a > 0, b \in \mathbf{R},$$

where again a, b are the dilation and translation parameters. As for the wavelet function, integral powers of 2 are used where the scaling function is obtained by a binary dilation (dilation by 2^j), and a dyadic translation (translation of $k/2^j$) of a single function ϕ. That is, a, b are chosen as for the wavelet function, and the scaling function becomes

$$\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k), \quad j, k \in \mathbf{Z}.$$

5.4.2 Scale Subspaces

Since the scaling function generates a basis of $L^2(\mathbf{R})$, it also generates subspaces $\{V_j\}$, just as the subspaces $\{W_j\}$ are generated by ψ above, i.e.,

$$V_j = clos_{L^2(\mathbf{R})}\langle \phi_{j,k} : k \in \mathbf{Z}\rangle,$$

with ϕ generating a reference subspace V_0, i.e.,

$$V_0 = clos_{L^2(\mathbf{R})}\langle \phi_{0,k} : k \in \mathbf{Z}\rangle.$$

In contrast to the sequence of orthogonal subspaces $\{W_j\}$ generated by an orthogonal ψ satisfying (5.20), the *nested* sequence of closed subspaces $\{V_j\}$ generated by the scaling function possess the following properties:

(5.31) 1. $\cdots \subset V_{-1} \subset V_0 \subset V_1 \cdots$ (*containment*);

(5.32) $\quad 2.\ \overline{\cup_{j\in\mathbf{Z}} V_j} = L^2(\mathbf{R}) \quad$ (*completeness*);

(5.33) $\quad 3.\ \cap_{j\in\mathbf{Z}} V_j = \{0\} \quad$ (*uniqueness*);

(5.34) $\quad 4.\ f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1}, j \in \mathbf{Z} \quad$ (*scalability*).

Property (5.31) states that the sequence of subspaces is nested; property (5.32) states that every function f in $L^2(\mathbf{R})$ can be approximated as closely as desired by its projections in V_j; property (5.33), on the other hand, states that, by decreasing j, the projections could have arbitrarily small energy [Chu92, p.16]; and property (5.34) is the multiresolution condition which states that as j increases, the spaces V_j correspond to "finer resolution": if the function f is in the basic multiresolution space V_0, then the finer resolution function $f(2^j\cdot) : x \mapsto f(2^j x)$ is in the space indexed by j [Fou95, p.43]. The scaling function is said to generate a multiresolution analysis if it generates a nested sequence of subspaces $\{V_j\}$ satisfying the above properties such that $\{\phi_{0,k}\}$ forms a basis of V_0.

5.4.3 Bi-orthogonal Multiresolution

Given a pair of scaling and wavelet functions (ϕ, ψ), neither of which necessarily forms an orthogonal basis, the goal is to specify dual functions $(\widetilde{\phi}, \widetilde{\psi})$ so that the original function f in $L^2(\mathbf{R})$ can be perfectly reconstructed. Recall that wavelets are bi-orthogonal if they satisfy condition (5.13) and generate dually orthogonal subspaces as expressed by Equations (5.16) and (5.17). Assuming that the scaling functions $(\phi, \widetilde{\phi})$ are dual as per (5.13), and imposing the following intra-scale orthogonality conditions:

(5.35) $$\langle \phi_{j,k}, \widetilde{\psi}_{j,l} \rangle = 0, \text{ and } \langle \widetilde{\phi}_{j,k}, \psi_{j,l} \rangle = 0, \quad j, k, l \in \mathbf{Z},$$

the double multiresolution generated by $(\phi, \widetilde{\phi})$ with two sequences of subspaces $\{V_j\}, \{\widetilde{V}_j\}$ then satisfies

$$V_j \perp \widetilde{W}_j \text{ and } \widetilde{V}_j \perp W_j, \quad j \in \mathbf{Z},$$

and $L^2(\mathbf{R})$ is decomposed as in (5.18), with

$$V_{j+1} = V_j \dot{+} W_j \text{ and } \widetilde{V}_{j+1} = \widetilde{V}_j \dot{+} \widetilde{W}_j, \quad j \in \mathbf{Z}.$$

The pairs (ϕ, ψ) and $(\widetilde{\phi}, \widetilde{\psi})$ are interchangeable in the sense that only one of the pairs needs to be specified. The second pair is derived from the first with ϕ connected to $\widetilde{\psi}$ and ψ connected to $\widetilde{\phi}$ (see §5.6.4) [Fou95, §II].

5.4.4 Orthogonal Multiresolution

If the scaling function ϕ can be chosen so that the set of translates $\{\phi_{0k}\} = \{\phi(x - k)\}$ forms an orthonormal basis and generates a set of multiresolution subspaces $\{V_j\}$, then an orthonormal wavelet basis can be constructed from ϕ [RBC+92]. Defining W_j as $V_j^{\perp}$, where the orthogonal complement is taken in V_{j+1}, so that

$$V_{j+1} = V_j \oplus W_j \text{ and } V_j \perp W_j,$$

$L^2(\mathbf{R})$ is decomposed as in (5.21). A function ψ is sought so that $\{\psi_{j,k}\}$ forms an orthonormal basis for $L^2(\mathbf{R})$, and subsequently $\{\psi_{j,k}\}$ is an orthonormal basis for W_j. Assuming that integer translates of ϕ generate an orthonormal basis for V_0 and there exist c_k such that

$$\phi(x) = \sum_{k \in \mathbf{Z}} c_k \phi(2x - k),$$

then $\psi(x)$ is given by

$$\psi(x) = \sum_{k \in \mathbf{Z}} (-1)^k c_{k+1} \phi(2x + k). \tag{5.36}$$

By the above construction and orthonormality of ϕ,

$$\langle \phi_{j,k}, \psi_{j,l} \rangle = 0, \quad j, k, l \in \mathbf{Z},$$

and ϕ, ψ are each self-dual, satisfying (5.35). That is, orthogonal multiresolution is a special case of bi-orthogonal multiresolution where $\phi = \widetilde{\phi}$ and $\psi = \widetilde{\psi}$.

5.5 Wavelet Decomposition and Reconstruction

Given the multiresolution framework, wavelet decomposition and reconstruction algorithms can be derived for any f in $L^2(\mathbf{R})$. Since $\widetilde{\phi} \in L^2(\mathbf{R})$ generates $\{\widetilde{V}_j\}$ and $\widetilde{\psi} \in L^2(\mathbf{R})$ generates $\{\widetilde{W}_j\}$, and by multiresolution property (5.32) above, every function f in $L^2(\mathbf{R})$ can be approximated by an $f^N \in \widetilde{V}_N$, for some $N \in \mathbf{Z}$. Consider $\widetilde{V}_N$ as the "sample space" and f^N the "data" (or measurement) of f on $\widetilde{V}_N$. Since

$$\begin{aligned} \widetilde{V}_N &= \widetilde{W}_{N-1} \dot{+} \widetilde{V}_{N-1} \\ &= \widetilde{W}_{N-1} \dot{+} \cdots \dot{+} \widetilde{W}_{N-M} \dot{+} \widetilde{V}_{N-M}, \end{aligned}$$

for any positive integer M, f^N has a unique decomposition:

$$f^N(x) = f^{N-1}(x) + g^{N-1}(x),$$

where $f^{N-1} \in \widetilde{V}_{N-1}$ and $g^{N-1} \in \widetilde{W}_{N-1}$. Recursively,

$$f^N(x) = g^{N-1}(x) + g^{N-2}(x) + \cdots + g^{N-M}(x) + f^{N-M}(x), \tag{5.37}$$

where

$$f^j(x) = \sum_k c_{j,k} \widetilde{\phi}(2^j x - k) \in \widetilde{V}_j \ : \ \mathbf{c}^j = \{c_{j,k}\}, \quad k \in \mathbf{Z}; \tag{5.38}$$

$$g^j(x) = \sum_k d_{j,k} \widetilde{\psi}(2^j x - k) \in \widetilde{W}_j \ : \ \mathbf{d}^j = \{d_{j,k}\}, \quad k \in \mathbf{Z}; \tag{5.39}$$

and

$$f^{N-M}(x) \in \widetilde{V}_{N-M}, \quad j = N-M, N-M+1, \ldots, N-1,$$

with the normalization factor $2^{j/2}$ folded into the series coefficients and M chosen so that f^{N-M} is sufficiently decomposed. The decomposition in (5.37) is uniquely determined by the sequences $\mathbf{c}^j$ and $\mathbf{d}^j$, in (5.38) and (5.39) [Chu92, pp.156-157], which are the scale and wavelet coefficients obtained from the multiresolution projection of f^j onto subspaces V_j, W_j as generated by ϕ, ψ, respectively:

$$c_{j,k} = \langle f^j, \phi_{j,k} \rangle, \quad d_{j,k} = \langle f^j, \psi_{j,k} \rangle. \tag{5.40}$$

Note that here, contrary to convention, ψ, not $\widetilde{\psi}$, is used as the analyzing wavelet, although by the *duality principle* [Chu92, p.156], the pairs $(\phi, \psi), (\widetilde{\phi}, \widetilde{\psi})$ are interchangeable for decomposition and reconstruction purposes. That is, incorporating Equations (5.38) and (5.39) into (5.37), the function f^{j+1} can be obtained from either combination of dual pairs by the following (bi-orthogonal) inversion formula [GB92, p.634]:

$$\begin{aligned} f^{j+1}(x) &= \sum_{j,k \in \mathbf{Z}} c_{j,k} \widetilde{\phi}_{j,k}(x) + \sum_{j,k \in \mathbf{Z}} d_{j,k} \widetilde{\psi}_{j,k}(x) && (5.41) \\ &= \sum_{j,k \in \mathbf{Z}} c_{j,k} \phi_{j,k}(x) + \sum_{j,k \in \mathbf{Z}} d_{j,k} \psi_{j,k}(x), && (5.42) \end{aligned}$$

with scale and wavelet coefficients

$$\begin{cases} c_{j,k} = \langle f^j, \phi_{j,k} \rangle, \quad d_{j,k} = \langle f^j, \psi_{j,k} \rangle & \text{in (5.41);} \\ c_{j,k} = \langle f^j, \widetilde{\phi}_{j,k} \rangle, \quad d_{j,k} = \langle f^j, \widetilde{\psi}_{j,k} \rangle & \text{in (5.42).} \end{cases}$$

In orthogonal MRA, with self-dual ϕ and ψ functions, Equations (5.41) and (5.42) condense into one (orthogonal) inversion formula:

$$f^{j+1}(x) = \sum_{j,k \in \mathbf{Z}} c_{j,k} \phi_{j,k}(x) + \sum_{j,k \in \mathbf{Z}} d_{j,k} \psi_{j,k}(x),$$

with coefficients as given by (5.40).

The most important property of the subspaces $\{V_j\}$ and $\{W_j\}$ (or $\{\widetilde{V}_j\}$ and $\{\widetilde{W}_j\}$, depending on which dual pair is used for decomposition), and hence multiresolution analysis in general, is that as $j \to -\infty$, more and more "variations" of the analyzed function are removed at each "rate of variation", or frequency band, j, and stored in W_j. The remaining coarser approximations to the function remain in V_j. The crux of the recursive nature of MRA is the decomposition of the coarse function at level j into the function's coarser approximation and stripped "variation" at level $j-1$, as projected onto V_{j-1} and W_{j-1}, respectively.[7] The

[7]Note that some authors use a convention of increasing subspaces [RBC+92]. Roughly speaking, in the Meyer convention (adopted here) the functions in V_j scale like 2^{-j}, whereas in the Daubechies convention they scale like 2^j. That is, in the Meyer convention, the decomposition level j is commensurate with the resolution of the function under study, i.e., level $j = 0$ represents the coarsest resolution. In the Daubechies

algorithmic approach for decomposing and reconstructing the function f^j between resolution levels is accomplished through the use of discrete sequences which approximate the scaling and wavelet functions ϕ, $\widetilde{\phi}$, ψ, $\widetilde{\psi}$.

Since both $\widetilde{\phi} \in \widetilde{V}_0$ and $\widetilde{\psi} \in \widetilde{W}_0$ are in $\widetilde{V}_1$, and since $\widetilde{V}_1$ is generated by $\widetilde{\phi}_{1,k}(x) = 2^{1/2}\widetilde{\phi}(2x-k),\ k \in \mathbf{Z}$, there exist two sequences denoted by $\{\widetilde{p}_k\}$ and $\{\widetilde{q}_k\}$ such that

$$\widetilde{\phi}(x) = \sum_k \widetilde{p}_k \widetilde{\phi}(2x-k); \tag{5.43}$$

$$\widetilde{\psi}(x) = \sum_k \widetilde{q}_k \widetilde{\phi}(2x-k), \tag{5.44}$$

for all $x \in \mathbf{R}$. These are the *two-scale*, *dilation*, or *refinement* relations of the scaling and wavelet functions, respectively. These relations imply that $\widetilde{\phi}(x)$ and $\widetilde{\psi}(x)$ must be generated by the finer scale functions $\widetilde{\phi}(2x-k)$, and lead to the decomposition algorithm.

Conversely, since both $\phi(2x)$ and $\phi(2x-1)$ are in V_1 and $V_1 = V_0 \dot{+} W_0$, there are two sequences denoted by $\{p_k\}$ and $\{q_k\}$, k in $\mathbf{Z}$, such that

$$\phi(2x-l) \quad = \quad \sum_k [p_{l-2k}\phi(x-k) + q_{l-2k}\psi(x-k)], \quad l \in \mathbf{Z}. \tag{5.45}$$

This is called the *decomposition relation* of ϕ and ψ. Mathematically, the decomposition relation roughly states that the function under analysis at a given resolution level (scale) can be decomposed into a coarser resolution approximation plus the stripped-off detail. Computationally, perhaps somewhat counterintuitively, the decomposition leads to the reconstruction algorithm. The two pairs of sequences $(\{\widetilde{p}_k\}, \{\widetilde{q}_k\})$ and $(\{p_k\}, \{q_k\})$, are unique once the normalization of ϕ is fixed (see [Chu92, §1.6] for details).

Representing f^j and g^j from (5.38) and (5.39) by the "digital" sequences $\mathbf{c}^j$ and $\mathbf{d}^j$, the following generalized (bi-orthogonal) decomposition and reconstruction algorithms emerge:

Decomposition:

$$c_k^{j-1} = \sum_l p_{l-2k} c_l^j; \quad d_k^{j-1} = \sum_l q_{l-2k} c_l^j, \tag{5.46}$$

Reconstruction:

$$c_k^j = \sum_l [\widetilde{p}_{k-2l} c_l^{j-1} + \widetilde{q}_{k-2l} d_l^{j-1}], \tag{5.47}$$

convention, the decomposition level j pertains to the number of decompositions applied to the function under study, i.e., level $j = 0$ represents the finest resolution since no decompositions have been applied to the function. Both conventions are equally informative since in the former the "current" resolution level can be used directly in estimating the extent of the function (i.e., the number of samples present in the scaled signal—this is particularly useful when dealing with images). The latter convention provides information in terms of number of decompositions applied to the function, which can be a valuable measure in a recursive implementation.

where $\{p_k\}$ and $\{q_k\}$ are *decomposition sequences*, while $\{\widetilde{p}_k\}$ and $\{\widetilde{q}_k\}$ are *reconstruction sequences*. These sequences correspond to digital filters in signal analysis.[8] Note that in the case of orthogonal MRA, the filters coincide, i.e., $\{p_k\} = \{\widetilde{p}_k\}$ and $\{q_k\} = \{\widetilde{q}_k\}$. The decomposition and reconstruction algorithms are shown schematically in Table 3.

TABLE 3
Schematic of wavelet decomposition and reconstruction.

$$
\begin{array}{ccccccccc}
 & & W_{N-1} & & & & W_{N-M+1} & & W_{N-M} \\
 & & \mathbf{d}^{N-1} & & & & \mathbf{d}^{N-M+1} & & \mathbf{d}^{N-M} \\
 & \nearrow & & \nearrow & & \nearrow & & \nearrow & \\
\mathbf{c}^{N} & \longrightarrow & \mathbf{c}^{N-1} & \longrightarrow & \cdots & \longrightarrow & \mathbf{c}^{N-M+1} & \longrightarrow & \mathbf{c}^{N-M} \\
V_N & & V_{N-1} & & & & V_{N-M+1} & & V_{N-M}
\end{array}
$$

(a) Decomposition

$$
\begin{array}{ccccccccc}
\widetilde{W}_{N-M} & & \widetilde{W}_{N-M+1} & & & & \widetilde{W}_{N-1} & & \\
\mathbf{d}^{N-M} & & \mathbf{d}^{N-M+1} & & & & \mathbf{d}^{N-1} & & \\
 & \searrow & & \searrow & & \searrow & & \searrow & \\
\mathbf{c}^{N-M} & \longrightarrow & \mathbf{c}^{N-M+1} & \longrightarrow & \cdots & \longrightarrow & \mathbf{c}^{N-1} & \longrightarrow & \mathbf{c}^{N} \\
\widetilde{V}_{N-M} & & \widetilde{V}_{N-M+1} & & & & \widetilde{V}_{N-1} & & \widetilde{V}_N
\end{array}
$$

(b) Reconstruction

The wavelet transform generalizes to multiple dimensions provided the scaling functions and wavelets generate multidimensional bases. In the particular two-dimensional case, there are two ways in which the 1D transform can be generalized, namely through the *standard* and *non-standard* decompositions.

The standard decomposition of a typical 2D function, i.e., an image, $f(x,y)$, is obtained by first applying the 1D wavelet transform to each row of (pixel) values, giving average (smoothed) values with detail coefficients for each row. The transformed rows are treated as 1D functions themselves and the 1D wavelet transform is

[8] Some authors prefer to concentrate on reconstruction filters as the “nice” filters and denote decomposition sequences by a special symbol. Because the decomposition is more pertinent to signal analysis, here the opposite convention is used where the “nice” filters are associated with decomposition and the distinguishing symbol $(\widetilde{\ })$ denotes reconstruction filters.

applied again on each column. The standard decomposition gives coefficients for a basis formed by the *standard construction* of wavelet basis functions, consisting of all possible tensor products of the one-dimensional basis functions,

$$\phi(x)\otimes\phi(x),\ \phi(x)\otimes\psi(x),\ \psi(x)\otimes\phi(x),\ \psi(x)\otimes\psi(x),$$

where $\phi(x)\otimes\phi(x)$ is the 2D scaling function and the rest are wavelets (see [Fou95, p.20] for details and examples).

The non-standard decomposition of a 2D function alternates between operations on rows and columns. That is, the decomposition is obtained by first applying the 1D wavelet transform to each row of (pixel) values *at one resolution level*, giving average (smoothed) values with detail coefficients for each row. The transformed rows are again treated as 1D functions and one level the 1D wavelet transform is applied again on each column. To complete the transform, the process is repeated recursively on the quadrant containing both row and column averages. The *non-standard construction* of a two-dimensional basis is similar to the standard construction, except that the tensor products are obtained using transposed versions of the 1D scaling and wavelet functions. That is, the two-dimensional scaling function is defined as

$$\phi\phi(x,y) \quad = \quad \phi(x)\otimes\phi^T(x),$$

and the three wavelet functions are:

$$\begin{aligned}
\phi\psi(x,y) &= \phi(x)\otimes\psi^T(x),\\
\psi\phi(x,y) &= \psi(x)\otimes\phi^T(x),\\
\psi\psi(x,y) &= \psi(x)\otimes\psi^T(x).
\end{aligned}$$

Both constructions will generate orthogonal 2D bases given orthogonal 1D functions [Fou95]. Examples of the non-standard decomposition and some of its properties are given in §5.7.

In three dimensions, the wavelet transform depends on three-dimensional scaling and wavelet bases functions. The standard decomposition of a typical 3D function, e.g., a video frame sequence, $f(x,y,t)$, is obtained by first applying the 1D wavelet transform on inter-frame pixels between two successive video frames at each resolution level. This gives the temporal decomposition of the video frames, analogous to the wavelet transform of one-dimensional signals. The first transformed frame contains the overall temporal average value, while the last frame contains the overall temporal difference of the original frames. The transformed frames are then treated as 2D functions and the standard 2D wavelet decomposition is applied to all frames.

The non-standard decomposition is obtained by first applying the 1D wavelet transform on each pixel between each of two successive video frames in the sequence. One of the two transformed frames contains the temporal average values, while the other frame contains the temporal difference of the two original frames. The transformed frames are then treated as 2D functions and the non-standard wavelet decomposition is applied to both frames. Provided there were four frames to begin with, the process is repeated recursively on the two quadrants containing both temporal and spatial averages which are contained in the two temporal average frames. The non-standard construction of a three-dimensional basis is similar to the two-dimensional case except that the temporal basis is obtained first. That is, the three-dimensional scaling function is defined as:

$$\phi\phi\phi(x,y,t) \quad = \quad \phi(x) \otimes \phi(x) \otimes \phi^T(x),$$

and the seven wavelet functions are:

$$\begin{aligned}
\phi\phi\psi(x,y,t) &= \phi(x) \otimes \phi(x) \otimes \psi^T(x), \\
\phi\psi\phi(x,y,t) &= \phi(x) \otimes \psi(x) \otimes \phi^T(x), \\
\phi\psi\psi(x,y,t) &= \phi(x) \otimes \psi(x) \otimes \psi^T(x), \\
\psi\phi\phi(x,y,t) &= \psi(x) \otimes \phi(x) \otimes \psi^T(x), \\
\psi\phi\psi(x,y,t) &= \phi(x) \otimes \phi(x) \otimes \psi^T(x), \\
\psi\psi\phi(x,y,t) &= \phi(x) \otimes \psi(x) \otimes \phi^T(x), \\
\psi\psi\psi(x,y,t) &= \phi(x) \otimes \psi(x) \otimes \psi^T(x).
\end{aligned}$$

The constructions will generate orthogonal 3D bases given orthogonal 1D functions. Examples of the non-standard decomposition are given in §5.7.

5.6 Wavelet Filters

The multiresolution wavelet decomposition and reconstruction, depicted in Table 3, can be implemented by a two-band filter bank, as shown in Figure 15. To maintain consistency with signal processing convention, the discrete sequences $\{p_k\}$, $\{q_k\}$, $\{\widetilde{p}_k\}$, $\{\widetilde{q}_k\}$ are replaced by the digital filters $H, G, \widetilde{H}, \widetilde{G}$ represented by discrete sequences $\{h_k\}$, $\{g_k\}$, $\{\widetilde{h}_k\}$, $\{\widetilde{g}_k\}$, respectively. Figure 15 displays decomposition and reconstruction of the signal f at one resolution level. The symbols $\downarrow 2$ and $\uparrow 2$ within circles represent dyadic downsampling and upsampling, respectively.

Multiresolution analysis at multiple levels resembles a nonuniform, tree-structured filter bank. The nonuniform qualification refers to the flexible tiling of the space-frequency grid generated by wavelet analysis (see §5.1.5, Figure 14) [Vai93]. Multiresolution decomposition and reconstruction at three levels is shown in Figures 16. In general, the digital implementation of multiresolution analysis, as described in §5.4, is often

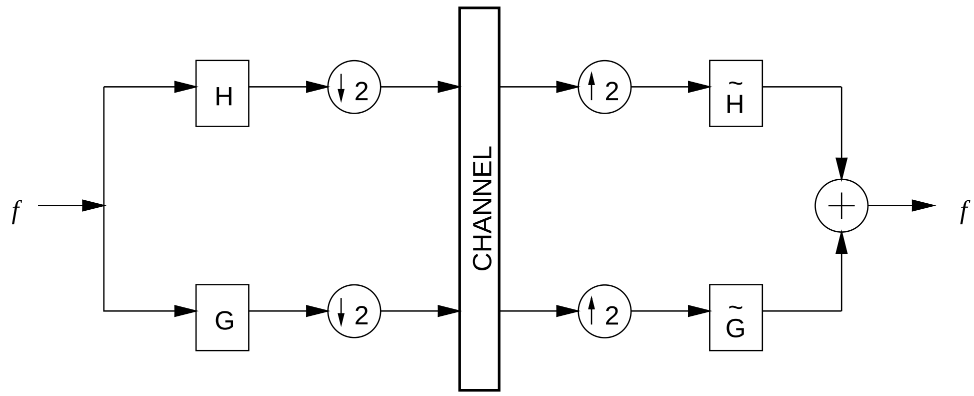


Fig. 15. One-level wavelet decomposition and reconstruction implemented by a two-band filter bank.

referred to as the Discrete Wavelet Transform (DWT).[9]

In practice, the filters H and G are chosen as lowpass and and highpass (in general, bandpass) filters, respectively. The lowpass filter corresponds to the scaling function ϕ by subsampling the signal at decreasing levels of resolution. The highpass (or bandpass) filter corresponds to the wavelet function ψ decomposing the signal by projections onto consecutive frequency bands. The dual filters $\widetilde{H},\widetilde{G}$ are derived from H,G subject to desired orthogonality constraints between filters. These constraints are delineated by the four wavelet classes discussed in §5.2 resulting in the consonant families of filters, namely *orthogonal*, *bi-orthogonal*, *semi-orthogonal*, and *non-orthogonal*.

The Discrete Wavelet Transform can be represented in matrix form [PTVF92]. At a given scale j, the finite, discrete function f, represented by the sequence $\mathbf{c}^j$, is transformed into the sequences $\mathbf{c}^{j-1}$ and $\mathbf{d}^{j-1}$ by the square matrix $\mathbf{M}^j$ consisting of null (zero) elements, and elements of the scaling and wavelet filters $\{h_k\}$, $\{g_k\}$. The transformed sequences $\mathbf{c}^{j-1}$, $\mathbf{d}^{j-1}$ are each half the length of $\mathbf{c}^j$ due to downsampling. For example, using scaling and wavelet filters $\{h_k\}$ and $\{g_k\}$, each of length 4, the decomposition of the sequence $\mathbf{c}^j$ of

[9]Strictly speaking, the term *Wavelet Transform* generally refers to the *Integral Wavelet Transform*, relative to the basic wavelet ψ, defined in Equation (5.7), and *Discrete Wavelet Transform* refers to the wavelet series expansion of f, relative to ψ. The transform is *dyadic* when a and b are chosen such that the wavelet basis is obtained by a binary dilation and dyadic translation of a single function ψ. In the signal processing domain, and especially in image and video processing applications, the term *Wavelet Transform*, or *Discrete Wavelet Transform (DWT)*, has come to mean a multiresolution analysis of the underlying signal. Although not technically accurate, this terminology is adopted here meaning that *Discrete Wavelet Transform* and the abbreviation *DWT* should be interpreted as "discrete, dyadic multiresolution analysis". The term *Inverse Discrete Wavelet Transform (IDWT)* should be interpreted as "discrete, dyadic multiresolution synthesis".

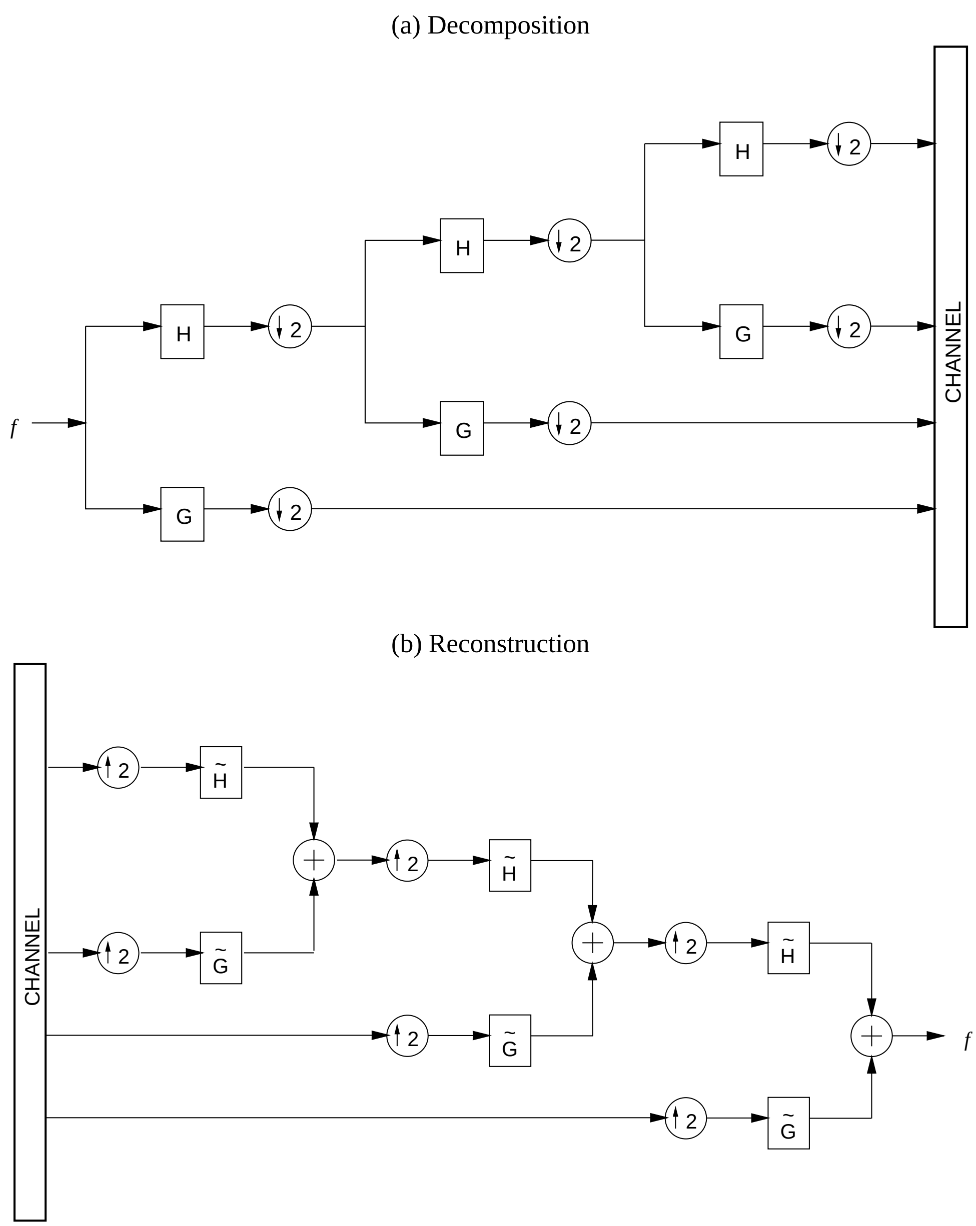


Fig. 16. Discrete Wavelet Transform implemented by a nonuniform, tree-structured, two-band filter bank.

length 8 is given by:

$$
\begin{bmatrix} c_0^{j-1} \\ c_1^{j-1} \\ c_2^{j-1} \\ c_3^{j-1} \\ d_0^{j-1} \\ d_1^{j-1} \\ d_2^{j-1} \\ d_3^{j-1} \end{bmatrix}
=
\begin{bmatrix}
h_0 & h_1 & h_2 & h_3 & & & & \\
 & & h_0 & h_1 & h_2 & h_3 & & \\
 & & & & h_0 & h_1 & h_2 & h_3 \\
h_2 & h_3 & & & & & h_0 & h_1 \\
g_0 & g_1 & g_2 & g_3 & & & & \\
 & & g_0 & g_1 & g_2 & g_3 & & \\
 & & & & g_0 & g_1 & g_2 & g_3 \\
g_2 & g_3 & & & & & g_0 & g_1
\end{bmatrix}
\begin{bmatrix} c_0^j \\ c_1^j \\ c_2^j \\ c_3^j \\ c_4^j \\ c_5^j \\ c_6^j \\ c_7^j \end{bmatrix}
$$

$$\mathbf{f}^{j-1} = \mathbf{M}^j \mathbf{f}^j \tag{5.48}$$

where $\mathbf{f}$ denotes the finite, discrete function f, and null elements of the matrix $\mathbf{M}^j$ are shown as empty spaces. The original function f can be perfectly reconstructed if the inverse matrix $(\mathbf{M}^j)^{-1}$ can be found and the dual filters $\{\widetilde{h}_k\}$, $\{\widetilde{g}_k\}$ exist. Construction of the dual filters depends on the chosen class of wavelets. Reconstruction is represented by a similar matrix operation where the reconstruction matrix resembles $\mathbf{M}^j$ except that the reconstruction filters $\{\widetilde{h}_k\}$ and $\{\widetilde{g}_k\}$ replace the decomposition filters $\{h_k\}$, $\{g_k\}$, e.g.,

$$\mathbf{f}^j = (\mathbf{M}^j)^{-1}\mathbf{f}^{j-1}.$$

Considering the filters $\{h_k\},\{g_k\}$ as convolution kernels, notice that the above matrix operation incorporates the subsampling step by performing dyadic translations of the kernels. In some signal processing implementations, convolution is carried out through monadic translation of the convolution filter, relying on the subsampling step to drop every other element. In the above matrix representation, however, the subsampling step is made implicit by dyadic translation precluding the need for explicit subsampling and supersampling. In the discussion on filters, below, dyadic kernel translation is assumed.

5.6.1 Orthogonal Filters

The orthogonality condition for the wavelet ψ, initially given in §5.2.2 by Equation (5.19), is restated here with respect to the analysis filter $\{g_k\}$: if ψ is an orthogonal wavelet, then the filter G forms an intra-scale orthonormal basis of $L^2(\mathbf{R})$,

$$\langle g_k, g_m\rangle = \delta_{k,m}, \quad k,m \in \mathbf{Z}. \tag{5.49}$$

Provided the filter $\{g_k\}$ also satisfies inter-scale orthogonality, the subspaces of $L^2(\mathbf{R})$ generated by G are mutually orthogonal as in Equation (5.20). Condition (5.49) effectively states that the translated wavelet function ψ, and hence highpass filter $\{g_k\}$, does not overlap, or if it does, the overlapped segments sum to zero in the sense of the $L^2(\mathbf{R})$ inner product.

As outlined in §5.4, $\{g_k\}$ can be obtained from the orthonormal lowpass filter $\{h_k\}$, corresponding to the scaling function ϕ, as per Equation (5.36), in which case H and G are called *quadrature mirror filters* [Mal89a]. Equation (5.36) can be rewritten more compactly in terms of the filters $\{h_k\},\{g_k\}$ by:

$$g_k \quad = \quad (-1)^k h_{1-k}, \quad k \in \mathbf{Z}, \tag{5.50}$$

so that the following intra-scale conditions hold:

$$\langle h_k, h_m\rangle = \langle g_k, g_m\rangle = \delta_{k,m}, \quad k,m \in \mathbf{Z} \quad \text{(orthonormal filters);} \tag{5.51}$$

$$\langle h_k, g_m\rangle = 0, \quad k,m \in \mathbf{Z} \qquad \text{(orthogonal subspaces } V_j \perp W_j\text{).} \tag{5.52}$$

Under this construction, the matrix $\mathbf{M}^j$ is orthogonal in the sense that the reconstruction matrix $(\mathbf{M}^j)^{-1}$ is the transpose of $\mathbf{M}^j$, i.e., $(\mathbf{M}^j)^{-1} = (\mathbf{M}^j)^T$, and the filters $\{h_k\},\{g_k\}$ are self-dual, i.e., $\{\widetilde{h}_k\} = \{h_k\}$ and $\{\widetilde{g}_k\} = \{g_k\}$.

Referring to the above matrix decomposition example with filters $\{h_k\}$ and $\{g_k\}$ of length 4, the reconstruction $\mathbf{f}^j = (\mathbf{M}^j)^{-1}\mathbf{f}^{j-1}$ is given by:

$$\begin{bmatrix} c_0^j \\ c_1^j \\ c_2^j \\ c_3^j \\ c_4^j \\ c_5^j \\ c_6^j \\ c_7^j \end{bmatrix} = \begin{bmatrix} \widetilde{h}_0 & \widetilde{g}_0 & & & & & \widetilde{h}_2 & \widetilde{g}_2 \\ \widetilde{h}_1 & \widetilde{g}_1 & & & & & \widetilde{h}_3 & \widetilde{g}_3 \\ \widetilde{h}_2 & \widetilde{g}_2 & \widetilde{h}_0 & \widetilde{g}_0 & & & & \\ \widetilde{h}_3 & \widetilde{g}_3 & \widetilde{h}_1 & \widetilde{g}_1 & & & & \\ & & \widetilde{h}_2 & \widetilde{g}_2 & \widetilde{h}_0 & \widetilde{g}_0 & & \\ & & \widetilde{h}_3 & \widetilde{g}_3 & \widetilde{h}_1 & \widetilde{g}_1 & & \\ & & & & \widetilde{h}_2 & \widetilde{g}_2 & \widetilde{h}_0 & \widetilde{g}_0 \\ & & & & \widetilde{h}_3 & \widetilde{g}_3 & \widetilde{h}_1 & \widetilde{g}_1 \end{bmatrix} \begin{bmatrix} c_0^{j-1} \\ d_0^{j-1} \\ c_1^{j-1} \\ d_1^{j-1} \\ c_2^{j-1} \\ d_2^{j-1} \\ c_3^{j-1} \\ d_3^{j-1} \end{bmatrix} \tag{5.53}$$

Substituting $\{\widetilde{h}_k\}$ by $\{h_k\}$ and $\{\widetilde{g}_k\}$ by $\{g_k\}$, where $\{g_k\}$ is obtained as in (5.50), and permuting rows of $\mathbf{f}^{j-1}$ and columns of $(\mathbf{M}^j)^{-1}$, Equation (5.53) is rewritten as:

$$\begin{bmatrix} c_0^j \\ c_1^j \\ c_2^j \\ c_3^j \\ c_4^j \\ c_5^j \\ c_6^j \\ c_7^j \end{bmatrix} = \begin{bmatrix} h_0 & h_3 & & & & & h_2 & h_1 \\ h_1 & -h_2 & & & & & h_3 & -h_0 \\ h_2 & h_1 & h_0 & h_3 & & & & \\ h_3 & -h_0 & h_1 & -h_2 & & & & \\ & & h_2 & h_1 & h_0 & h_3 & & \\ & & h_3 & -h_0 & h_1 & -h_2 & & \\ & & & & h_2 & h_1 & h_0 & h_3 \\ & & & & h_3 & -h_0 & h_1 & -h_2 \end{bmatrix} \begin{bmatrix} c_0^{j-1} \\ d_0^{j-1} \\ c_1^{j-1} \\ d_1^{j-1} \\ c_2^{j-1} \\ d_2^{j-1} \\ c_3^{j-1} \\ d_3^{j-1} \end{bmatrix} \tag{5.54}$$

In this example, $(\mathbf{M}^j)^T$ is the inverse of $\mathbf{M}^j$ if and only if

$$h_0^2 + h_1^2 + h_2^2 + h_3^2 = 1, \text{ and,} \tag{5.55}$$

$$h_0 h_2 + h_1 h_3 = 0. \tag{5.56}$$

Equations (5.55) and (5.56) in combination express the intra-scale orthonormality condition (5.51). If condition (5.56) is not evident from the invertible matrix requirement, consider the intra-scale orthogonality of the subspace $\{V_j\}$, covered by the scaling function, which can be exemplified by two vectors formed by the spatial translation of the lowpass filter, h_k, h_{k+1}:

$$\begin{matrix} [\ldots & h_0 & h_1 & h_2 & h_3 & 0 & 0 & \ldots] \\ [\ldots & 0 & 0 & h_0 & h_1 & h_2 & h_3 & \ldots] \end{matrix}$$

where the inner product $\langle h_k, h_{k+1}\rangle = h_0h_2 + h_1h_3$. These are precisely the terms required to sum to 0 in Equation (5.56). In general, for any different translations $k, m,\ k \neq m$, the inner product must sum to zero. In other words, $\langle h_k, h_m\rangle = 0,\ k \neq m$ so that the scaling function ϕ generates an orthogonal basis. Equations (5.55) and (5.56), along with two additional relations, were recognized and solved by Daubechies, while coefficients for filters of length 2 were first given by Haar. Coefficients of both filters are given in Table 4.

TABLE 4
Orthonormal filters.

(a) Haar.

k	$\sqrt{2}(h_k)$	$\sqrt{2}(g_k)$
0	1	1
1	1	-1

(b) Daubechies-4.

k	$4\sqrt{2}(h_k)$	$4\sqrt{2}(g_k)$
0	$1+\sqrt{3}$	$1-\sqrt{3}$
1	$3+\sqrt{3}$	$-3+\sqrt{3}$
2	$3-\sqrt{3}$	$3+\sqrt{3}$
3	$1-\sqrt{3}$	$-1-\sqrt{3}$

Orthogonal wavelets guarantee perfect reconstruction and generally facilitate implementation. In practice, however, orthogonal wavelets are not always easily constructed and may lack desirable properties such as symmetry or continuity. Filter symmetry is incompatible with exact reconstruction, if the same FIR filters are used for decomposition and reconstruction. Except for the Haar basis, *all* compactly supported, real orthonormal wavelet bases are asymmetric [Dau92, p.252,p.253,p.259]. The Haar wavelet is the only real-valued wavelet that is compactly supported, symmetric and orthogonal [JS94a]. Non-orthogonal, or overlapping filters, relax the orthogonality condition and subsequently are considered more flexible.

5.6.2 Semi-Orthogonal Filters

Recall that a function $\psi \in L^2(\mathbf{R})$ is semi-orthogonal if the generated basis $\{\psi_{j,k}\}$ is orthogonal, as expressed by (5.22). This condition suggests that the corresponding filters need not be fully orthonormal, only orthogonal, generating mutually orthogonal subspaces. In effect, the intra-scale orthonormality condition (5.51) is relaxed so that

$$\langle h_k, h_m\rangle = \langle g_k, g_m\rangle = 0, \quad k \neq m, \quad k, m \in \mathbf{Z} \quad \text{(orthogonal filters);}$$

while condition (5.52) remains:

$$\langle h_k, g_m\rangle = 0, \quad k, m \in \mathbf{Z} \quad \text{(orthogonal subspaces } V_j \perp W_j\text{).}$$

Semi-orthogonal filters can produce orthogonal filters through an orthogonalization procedure, as mentioned in §5.2.3.

5.6.3 Non-Orthogonal Filters

Non-orthogonal filters are filters that are not semi-orthogonal. That is, non-orthogonal filters do not generate mutually orthogonal subspaces. Effectively, they are overlapping filters. In general, non-orthogonal filters require their duals to guarantee perfect reconstruction.

5.6.4 Bi-orthogonal Filters

Following §5.4.3, given a pair of lowpass and highpass filters $\{h_k\},\{g_k\}$, neither necessarily being orthogonal, dual filters $\{\widetilde{h}_k\},\{\widetilde{g}_k\}$ are required to guarantee perfect reconstruction. In particular, by (5.35), the following relations must hold:

$$\langle h_k, \widetilde{g}_m \rangle = 0, \quad k,m \in \mathbf{Z}; \tag{5.57}$$

$$\langle g_k, \widetilde{h}_m \rangle = 0, \quad k,m \in \mathbf{Z}. \tag{5.58}$$

Note that orthogonal filters satisfy these requirements through the stringent condition of orthonormality placed on $\{h_k\}$ and subsequently on $\{g_k\}$. Since $H = \widetilde{H}$, and H is orthonormal, i.e., $\langle h_k, h_m \rangle = \delta_{k,m}$, then $\langle \widetilde{h}_k, \widetilde{h}_m \rangle = \delta_{k,m}$ also holds. Moreover, since $\{g_k\}$ is the quadrature mirror of $\{h_k\}$, $\{g_k\}$ and $\{\widetilde{g}_k\}$ are also orthonormal. The construction of biorthogonal filters, on the other hand, is based on the relaxation of the orthonormality condition, so that in general, $H \neq \widetilde{H}$. The requirement of bi-orthogonal dual bases remains. That is, the intra-scale orthonormality condition, contained in (5.13), is rewritten in terms of the two sets of filters $H, \widetilde{H}, G, \widetilde{G}$ as:

$$\langle h_k, \widetilde{h}_m \rangle = \delta_{k,m}, \quad k,m \in \mathbf{Z}; \tag{5.59}$$

$$\langle g_k, \widetilde{g}_m \rangle = \delta_{k,m}, \quad k,m \in \mathbf{Z}. \tag{5.60}$$

In general, the relaxation of the orthonormality condition and the use of dual filters provides greater flexibility in the construction of filters. Specifically, symmetric filters can be constructed. The construction of bi-orthogonal wavelets is typically performed by specifying the decomposing (or reconstructing) pair of functions (ϕ, ψ), then deriving their duals such that the above bi-orthogonal conditions are satisfied. One method, as suggested in §5.4.3, is to derive $\{\widetilde{h}_k\}$ from $\{g_k\}$ and $\{\widetilde{g}_k\}$ from $\{h_k\}$ by the quadrature mirror construction (see [Dau92, §8.3]):

$$\begin{aligned} \widetilde{g}_k &= (-1)^k h_{1-k}, \quad k \in \mathbf{Z}, \\ \widetilde{h}_k &= (-1)^k g_{1-k}, \quad k \in \mathbf{Z}. \end{aligned}$$

Various authors have constructed symmetric, bi-orthogonal filters. Most constructions rely on filter bank theory [GB92] or on multiresolution derivations usually using the spline family of functions which provides continuity as well as symmetry. Ueda and Lodha provide an excellent introduction into B-spline wavelets including derivations of linear, quadratic, and cubic B-spline wavelet filters [UL95]. Well known bi-orthogonal filters have been designed by Cohen, Daubechies, and Feaveau [Bar94]. Chui has developed a family of spline wavelets based on cardinal B-spline functions [Chu92, §4]. Barlaud derived near-orthonormal dual spline wavelets constructed from the popular Laplacian pyramid filter introduced by Burt and Adelson [BA83b], which itself is a near-orthonormal wavelet filter [ABMD92]. The Laplacian filters are in turn very similar to the orthonormal *coiflet* basis developed by Coifman. Mallat et al. have developed quadratic spline wavelets which are particularly suitable for singularity detection [MZ92a]. Coefficients of the Mallat, Chui (multiplicity-2), Barlaud, and Burt and Adelson filters are given in §A.

Unfortunately, although one set of filters may possess many desirable properties, the dual filters are, in general, not compactly supported [JS94a]. This may cause significant implementational problems. For example, Chui's (multiplicity-2) decomposition filters are of length 41, while Mallat's filters require special normalization operations at various levels of reconstruction. Furthermore, if the filters are not separable then implementation of multi-dimensional wavelet transforms becomes even more problematic.

5.7 Discrete Wavelet Transform

The one-dimensional Discrete Wavelet Transform (DWT) is characterized by the decomposition and reconstruction Equations (5.46) and (5.47) described in §5.5. The implementation of the 1D DWT follows the general digital filter representation portrayed by Figures 15 and 16, and an example of the 1D decomposition through convolution was given by the matrix representation (5.48) in §5.6.

Given an n-length discrete function at the j^{th} level of resolution,

(5.61) $$f^j(x) = f^j_\phi(1), f^j_\phi(2), \ldots, f^j_\phi(n),$$

the decomposition relations of the function are:

(5.62) $$f^{j-1}_\phi(x) = \sum_k h_k f^j_\phi(2x+k),$$

(5.63) $$f^{j-1}_\psi(x) = \sum_k g_k f^j_\phi(2x+k),$$

where $\{h_k\}, \{g_k\}$ are the one-dimensional low- and high-pass filters. This gives the discrete wavelet transform:

(5.64) $$\{Wf(x)\}(j-1) = f^{j-1}_\phi(1), f^{j-1}_\psi(2), \ldots, f^{j-1}_\phi(n-1), f^{j-1}_\psi(n).$$

Permuting the terms so that the first $n/2$ elements are the low-pass (scale) coefficients, i.e.,

$$\{Wf(x)\}(j-1) = \\ f_\phi^{j-1}(1), f_\phi^{j-1}(3), \ldots, f_\phi^{j-1}(n-1), f_\psi^{j-1}(2), f_\psi^{j-1}(4), \ldots, f_\psi^{j-1}(n), \tag{5.65}$$

and relabeling the indices,

$$\{Wf(x)\}(j-1) = f_\phi^{j-1}(1), \ldots, f_\phi^{j-1}(n/2-1), f_\psi^{j-1}(n/2), \ldots, f_\psi^{j-1}(n) \tag{5.66}$$

the smooth (or averaged) elements (the first $n/2$ elements) are recursively decomposed. The fully transformed function contains the global average as the first element, the next 2^j elements contain the detail (or difference) information at each resolution level j. Except for the average value, the transformed elements comprise the so-called wavelet coefficients of the function.

To reconstruct the function, the terms at each resolution level are repermuted so that the average and wavelet coefficients are interleaved, as per Equation (5.64). Introducing the $\bowtie$ operator denoting element interleave, the $j-1$ level coefficients can be arranged into an intermediate representation for reconstruction at level j:

$$f_{\phi\bowtie\psi}^{j-1}(2x+p) \quad = \quad (1-p)\, f^{j-1}(x) + (p)\, f^{j-1}(x),$$

for $p \in \{0,1\}$. Reconstruction at level j, with $p \in \{0,1\}$ is then written as:

$$f_\phi^j(2x+p) = (1-p)\sum_k \widetilde{h}_k f_{\phi\bowtie\psi}^{j-1}(x-k) + (p)\sum_k \widetilde{g}_k f_{\phi\bowtie\psi}^{j-1}(x-k)$$

which gives the original function $f^j(x)$ in (5.61). Note that the variable p is used as a selection variable, that is, in the dyadic wavelet reconstruction, the element at position $2x+p$ is the result of filtering the lower level elements with either $\{\widetilde{h}_k\}$ or $\{\widetilde{g}_k\}$. This is a convenient substitute for writing two equations:

$$f_\phi^j(2x) = \sum_k \widetilde{h}_k f_{\phi\bowtie\psi}^{j-1}(x-k);$$
$$f_\phi^j(2x+1) = \sum_k \widetilde{g}_k f_{\phi\bowtie\psi}^{j-1}(x-k).$$

The permutation function $f_{\phi\bowtie\psi}^j$ serves as an alternate method of reconstruction used instead of traditional supersampling. In contrast, without permutating the average and wavelet coefficients, there would be two sequences f_ϕ^{j-1}, f_ψ^{j-1}, each of length $n/2$, where n is the length of the sequence at level j. In this case, reconstruction is given as:

$$f_\phi^j(2x-l) = \sum_k \widetilde{h}_{l-2k} f_\phi^{j-1}(x-k) + \sum_k \widetilde{g}_{l-2k} f_\psi^{j-1}(x-k), \tag{5.67}$$

which follows from the decomposition relation given in (5.45). The reconstruction in (5.67) is equivalent to the one given by (5.67), however its implementation is obviously different. In all following (multidimensional) wavelet transform discussions the permutation approach is adopted. This is consistent with the 1D reconstruction example of (5.54) corresponding to the decomposition of (5.48) in §5.6. A numerical example of orthogonal decomposition and reconstruction using Haar filters is given in Table 5, where the symbol $\bowtie$ denotes element permutation.

TABLE 5
Numerical 1D DWT example.

Decomposition							
$\mathbf{f}^2 = \mathbf{c}^2$:	1		4		0		-2
$\mathbf{d}^1$:		$\frac{-3}{\sqrt{2}}$				$\frac{2}{\sqrt{2}}$	
$\mathbf{c}^1$:		$\frac{5}{\sqrt{2}}$				$\frac{-2}{\sqrt{2}}$	
$\mathbf{d}^0$:				$\frac{7}{2}$			
$\mathbf{c}^0$:				$\frac{3}{2}$			
$\mathbf{W}f$:	$\frac{3}{2}$		$\frac{7}{2}$		$\frac{-3}{\sqrt{2}}$		$\frac{2}{\sqrt{2}}$
Reconstruction							
$\mathbf{c}^0 \bowtie \mathbf{d}^0$:		$\frac{3}{2}$				$\frac{7}{2}$	
$\mathbf{c}^1$:				$\frac{5}{\sqrt{2}}, \frac{-2}{\sqrt{2}}$			
$\mathbf{c}^1 \bowtie \mathbf{d}^1$:	$\frac{5}{\sqrt{2}}$		$\frac{-3}{\sqrt{2}}$		$\frac{-2}{\sqrt{2}}$		$\frac{2}{\sqrt{2}}$
$\mathbf{c}^2$:		1,4				0,-2	
$\mathbf{c}^2 = \mathbf{f}^2$:	1		4		0		-2

Multidimensional extensions of the DWT rely on the use of multidimensional bases, described in §5.5, which are constructed by obtaining the tensor product of unidimensional bases. The matrix tensor product operation is reviewed in §B. The computational realization of the 2D DWT described here is an instance of the well-known pyramidal multiresolution representational framework first proposed by Tanimoto and Pavlidis (see [TK80, §2, pp.31-56] and [JR94, p.3]). The pyramidal model stipulates a hierarchical processing paradigm known as the *coarse-to-fine* (resolution) strategy. The pyramidal wavelet transform in particular is related to the Laplacian pyramid introduced by Burt and Adelson for image coding [BA83b]. In two dimensions, a $\log_2 N$ level pyramid is constructed from an $N \times N$ image, where the bottom level of the pyramid (level $j = \log_2 N - 1$) contains the finest resolution, and the top level ($j = 0$) contains the coarsest. Note that the original image is considered the finest level of resolution (level $j = \log_2 N$), however it is not contained within the pyramid itself.

Given 1D scaling and wavelet filters H, G associated with ϕ, ψ, respectively, the 2D filters corresponding to the 2D wavelet bases $\phi\phi$, $\phi\psi$, $\psi\phi$, $\psi\psi$, as described in §5.5, are generated by the non-standard 2D wavelet basis construction using tensor products:

$$\begin{aligned} HH &= H \otimes H^T, \\ HG &= H \otimes G^T, \\ GH &= G \otimes H^T, \\ GG &= G \otimes G^T. \end{aligned}$$

As an example consider the Haar filters given in Table 4. Their two-dimensional extensions are derived below:

$$\begin{aligned} HH = \phi \otimes \phi^T &= \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} & \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}; \\ HG = \phi \otimes \psi^T &= \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} & \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix}; \\ GH = \psi \otimes \phi^T &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} & -\frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}; \\ GG = \psi \otimes \psi^T &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} & -\frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}. \end{aligned}$$

Multiplying each 2D filter by the dyadic normalization factor $2^{j/2}$ as suggested by Equations (5.8) and (5.10), the 2D filters become:

$$HH = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix};\ HG = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix};$$
$$GH = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix};\ GG = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}.$$

At the next level of resolution, the filters are derived by:

$$\begin{aligned} HHHH &= (\phi \otimes \phi^T) \otimes (\phi \otimes \phi^T); \\ HGHH &= (\phi \otimes \psi^T) \otimes (\phi \otimes \phi^T); \\ GHHH &= (\psi \otimes \phi^T) \otimes (\phi \otimes \phi^T); \\ GGHH &= (\psi \otimes \psi^T) \otimes (\phi \otimes \phi^T). \end{aligned}$$

Taking dyadic filter translation into consideration, the 2D filters are clearly mutually orthogonal in x- and y-directions, and in this example orthonormal. In general, multidimensional orthogonal filters are also *separable*, i.e., satisfying

$$h(k,m) = h(k)h(m),$$

due to their tensor product construction. Note that the above example illustrates one of the drawbacks of the DWT, namely, depending on the choice of bases, the DWT is neither necessarily translationally nor rotationally invariant.

The multidimensional tensor product filters are useful for visualizing spatiotemporal properties of the multidimensional wavelet transform, however direct implementation with multidimensional filters is inefficient. Instead, relying on the separability of the filters, the wavelet transform can be implemented by processing each dimension separately. The decomposition relations describing the non-standard decomposition of the spatial average image at level j are:

$$\begin{array}{ll} f_{\phi_r}^{j-1}(x,y) = \sum_k h_k f_{\phi\phi}^{j}(x,2y+k) & f_{\psi_r}^{j-1}(x,y) = \sum_k g_k f_{\phi\phi}^{j}(x,2y+k) \\ f_{\phi\phi}^{j-1}(x,y) = \sum_k h_k f_{\phi_r}^{j-1}(2x+k,y) & f_{\phi\psi}^{j-1}(x,y) = \sum_k h_k f_{\psi_r}^{j-1}(2x+k,y) \\ f_{\psi\phi}^{j-1}(x,y) = \sum_k g_k f_{\phi_r}^{j-1}(2x+k,y) & f_{\psi\psi}^{j-1}(x,y) = \sum_k g_k f_{\psi_r}^{j-1}(2x+k,y) \end{array}$$

where $\{h_k\},\{g_k\}$ are the one-dimensional low- and high-pass filters. The non-standard DWT first involves subsampling the rows of the lower resolution level spatial average image (denoted by $f_{\phi\phi}^{j}$) to generate the temporary upper resolution level images $f_{\phi_r}^{j-1}$ and $f_{\psi_r}^{j-1}$. Due to dyadic downsampling of rows, these images are half the width of the image at the lower resolution level. The columns of these subimages are then subsampled to generate the four subimages denoted by the subscripts $\phi\phi$, $\phi\psi$, $\psi\phi$, and $\psi\phi$. The above equations are written verbosely in order to facilitate implementation. Rewriting the equations concisely,

$$f_{\phi\phi}^{j-1}(x,y) = \sum_{k,m}(h_k \otimes h_m)\, f_{\phi\phi}^{j}(2x+k, 2y+m) \tag{5.68}$$

$$f_{\psi\phi}^{j-1}(x,y) = \sum_{k,m}(g_k \otimes h_m)\, f_{\phi\phi}^{j}(2x+k, 2y+m) \tag{5.69}$$

$$f_{\phi\psi}^{j-1}(x,y) = \sum_{k,m}(h_k \otimes g_m)\, f_{\phi\phi}^{j}(2x+k, 2y+m) \tag{5.70}$$

$$f_{\psi\psi}^{j-1}(x,y) = \sum_{k,m}(g_k \otimes g_m)\, f_{\phi\phi}^{j}(2x+k, 2y+m) \tag{5.71}$$

it is clear that the decomposition algorithm follows the two-scale relations (5.43) and (5.44). The smooth (or averaged) subimage $f_{\phi\phi}^{j}$ is recursively subsampled at each stage of the decomposition. The transformed image contains the global average at the top of the pyramid, the lower layers contain the detail (or difference) information at each pyramid level. These lower layers comprise the so-called wavelet coefficients of the transformed image. The decomposition is shown schematically in Figure 17 where the image matrix f is subsampled with low- and high-pass filters h, g. The subscripts r, c represent the subsampling operation

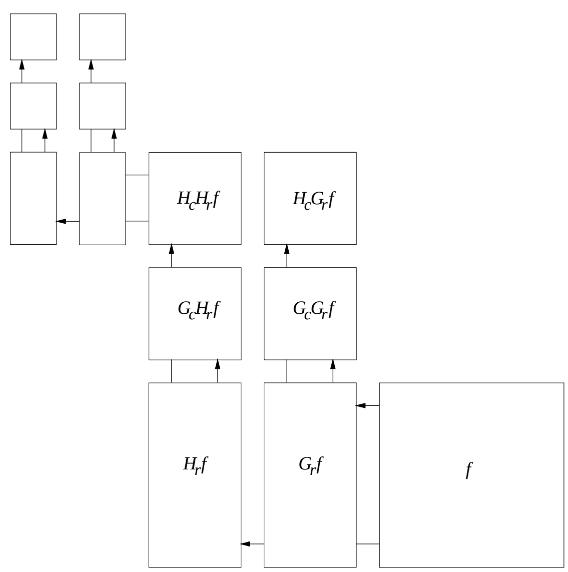


Fig. 17. Non-standard 2D pyramidal decomposition.

performed on rows and columns, i.e., $f_{\phi\phi}^{j-1} = H_cH_r f^j$.

In practice, depending on the length of filters $\{h_k\}, \{g_k\}$, boundary conditions require special consideration. There are generally two strategies used to handle this problem: extending the image by padding with zero values, or periodic extension of the image (i.e., tiling copies of the image). In the present implementation the latter strategy is implemented by applying modulo r, c to the indices at each resolution level. This generates a *wraparound* at the image borders when the filters extend beyond the image boundary. All references to image locations (x, y) are extended beyond image boundaries by the indices $((r+x) \bmod r, (c+y) \bmod c)$ where r, c are the dimensions of $f^j_{\phi\phi}$. This strategy allows the use of negative indices (required during reconstruction) and the processing of non-square images. An example of the 2D DWT applied to an image is shown in Figure 18. The transformed image has been processed for display purposes.

(a) Original *cnn* image. Reprinted with permission from Turner Broadcasting System, Inc. (see §F).

(b) 2-level DWT (processed by histogram equalization with subsampled image inset).

Fig. 18. Non-standard 2D DWT.

In traditional pyramidal approaches, where the pyramid contains only smoothed multiscale versions of the original image (e.g., texture-mapping applications), subimages at each level provide the pixel intensity values for reconstruction usually involving interpolation (cf. §5.10). In the wavelet transform, the image is synthesized by a recursive process of adding detail information to the average (smoothed) subimages in order to reconstruct the next level's average subimage. Rows and columns are interleaved prior to filtering instead of the traditional null row and column padding (see [PTVF92] for an example of the interleave operation, and [Cas96] for padding examples). Generally, row and column padding (supersampling) is used under monadic convolution. Dyadic convolution precludes the need for padding, but instead requires that rows and columns be interleaved prior to reconstruction filtering. Introducing the $\bowtie_r$ and $\bowtie_c$ operators denoting row and column interleave, respectively, the reconstruction is obtained with the use of the following intermediate relations:

(5.72)
$$f^j_{\phi\phi \bowtie_r \psi\phi}(2x+p, y) \quad = \quad (1-p)\, f^{j-1}_{\phi\phi}(x, y) + (p)\, f^{j-1}_{\psi\phi}(x, y),$$

$$f^{j}_{\phi\psi\bowtie_r\psi\psi}(2x+p,y) \quad = \quad (1-p)\,f^{j-1}_{\phi\psi}(x,y)+(p)\,f^{j-1}_{\psi\psi}(x,y), \tag{5.73}$$

$$f^{j}_{\phi_r\bowtie_c\psi_r}(x,2y+q) \quad = \quad (1-q)\,f^{j}_{\phi_r}(x,y)+(q)\,f^{j}_{\psi_r}(x,y), \tag{5.74}$$

where $p,q\in\{0,1\}$, $x,y,k\in\mathbf{Z}$ and

$$\begin{aligned} f^{j}_{\phi_r}(2x+p,y) &= (1-p)\sum_k \widetilde{h}_k\, f^{j}_{\phi\phi\bowtie_r\psi\phi}(2x-k,y)+(p)\sum_k \widetilde{g}_k\, f^{j}_{\phi\phi\bowtie_r\psi\phi}(2x-k,y), \\ f^{j}_{\psi_r}(2x+p,y) &= (1-p)\sum_k \widetilde{h}_k\, f^{j}_{\phi\psi\bowtie_r\psi\psi}(2x-k,y)+(p)\sum_k \widetilde{g}_k\, f^{j}_{\phi\psi\bowtie_r\psi\psi}(2x-k,y), \end{aligned}$$

so that

$$f^{j}_{\phi\phi}(x,2y+q) \quad = \quad (1-q)\sum_k \widetilde{h}_k\, f^{j}_{\phi_r\bowtie_c\psi_r}(x,2y-k)+(q)\sum_k \widetilde{g}_k\, f^{j}_{\phi_r\bowtie_c\psi_r}(x,2y-k).$$

The above reconstruction relations in two dimensions can be rewritten succinctly following the decomposition relation given in (5.45):

$$\begin{aligned} f^{j}_{\phi\phi}(2x+p,2y+q) \quad = \quad & (1-q)\left[(1-p)\sum_{k,m}(\widetilde{h}_k\otimes\widetilde{h}_m)\,f^{j-1}_{\phi\phi}(x-k,y-m)+\right. \\ & \left. p\sum_{k,m}(\widetilde{h}_k\otimes\widetilde{g}_m)\,f^{j-1}_{\psi\phi}(x-k,y-m)\right]+ \\ & (q)\left[(1-p)\sum_{k,m}(\widetilde{g}_k\otimes\widetilde{h}_m)\,f^{j-1}_{\phi\psi}(x-k,y-m)+\right. \\ & \left. p\sum_{k,m}(\widetilde{g}_k\otimes\widetilde{g}_m)\,f^{j-1}_{\psi\psi}(x-k,y-m)\right], \end{aligned} \tag{5.75}$$

where $x,y,k,m\in\mathbf{Z}$, wraparound indices in the reverse direction, written as $(x-k)$, are assumed, and $p,q\in\{0,1\}$ are used as selection variables analogously as in the one-dimensional reconstruction.

To show that Equation (5.75) is derived from the above relations, expand $f^{j}_{\phi\phi}(x,2y+q)$:

$$\begin{aligned} f^{j}_{\phi\phi}(x,2y+q) &= (1-q)\sum_k \widetilde{h}_k\, f^{j}_{\phi_r\bowtie_c\psi_r}(x,2y-k)+(q)\sum_k \widetilde{g}_k\, f^{j}_{\phi_r\bowtie_c\psi_r}(x,2y-k) \\ &= (1-q)\sum_k \widetilde{h}_k\left[(1-q)\,f^{j}_{\phi_r}(x,y-k)+(q)\,f^{j}_{\psi_r}(x,y-k)\right]+ \\ & \quad (q)\sum_k \widetilde{g}_k\left[(1-q)\,f^{j}_{\phi_r}(x,y-k)+(q)\,f^{j}_{\psi_r}(x,y-k)\right]. \end{aligned} \tag{5.76}$$

Since the (q), $(1-q)$ terms are symbolic for binary selection, multiple like terms can be combined into one, i.e., $(1-q)^k=(1-q)(1-q)\cdots(1-q)=(1-q)$, and $(q)^k=(q)(q)\cdots(q)=(q)$ for all $q\in\{0,1\},k\in\mathbf{Z}$. Conversely, unlike terms cancel, simplifying Equation (5.76) to:

$$f^{j}_{\phi\phi}(x,2y+q) \quad = \quad (1-q)\sum_k \widetilde{h}_k\, f^{j}_{\phi_r}(x,y-k)+(q)\sum_k \widetilde{g}_k\, f^{j}_{\psi_r}(x,y-k).$$

Substituting appropriately for $f^j_{\phi_r}$, $f^j_{\psi_r}$ by changing the resolution of x to $2x+p$ and taking care to disambiguate summation indices,

$$
\begin{aligned}
f^j_{\phi\phi}(2x+p,2y+q) \quad = \quad & (1-q)\sum_m \widetilde{h}_m \left[(1-p)\sum_k \widetilde{h}_k f^j_{\phi\phi\bowtie_r\psi\phi}(2x-k,y-m)+\right.\\
& \left.(p)\sum_k \widetilde{g}_k f^j_{\phi\phi\bowtie_r\psi\phi}(2x-k,y-m)\right]+\\
& (q)\sum_m \widetilde{g}_m \left[(1-p)\sum_k \widetilde{h}_k f^j_{\phi\psi\bowtie_r\psi\psi}(2x-k,y-m)+\right.\\
& \left.(p)\sum_k \widetilde{g}_k f^j_{\phi\psi\bowtie_r\psi\psi}(2x-k,y-m)\right].
\end{aligned} \tag{5.77}
$$

Noting that the filter summation terms apply to rows and columns separately, i.e., $\sum_m \widetilde{h}_m \sum_k \widetilde{g}_k$ refers to the tensor product of $\widetilde{H}$ and $\widetilde{G}$ since $\sum_k \widetilde{g}_k$ applies to the columns of f^j, the double summation terms can be collected and represented by the single summation term $\sum_{k,m}(\widetilde{h}_k \otimes \widetilde{g}_m)$. Equation (5.77) is then rewritten as:

$$
\begin{aligned}
f^j_{\phi\phi}(2x+p,2y+q) \quad = \quad & (1-q)\left[(1-p)\sum_{k,m}(\widetilde{h}_k \otimes \widetilde{h_m})\, f^j_{\phi\phi\bowtie_r\psi\phi}(2x-k,y-m)+\right.\\
& \left.(p)\sum_{k,m}(\widetilde{h}_k \otimes \widetilde{g}_m)\, f^j_{\phi\phi\bowtie_r\psi\phi}(2x-k,y-m)\right]+\\
& (q)\left[(1-p)\sum_{k,m}(\widetilde{g}_k \otimes \widetilde{h}_m)\, f^j_{\phi\psi\bowtie_r\psi\psi}(2x-k,y-l)+\right.\\
& \left.(p)\sum_{k,m}(\widetilde{g}_k \otimes \widetilde{g}_m)\, f^j_{\phi\psi\bowtie_r\psi\psi}(2x-k,y-m)\right].
\end{aligned} \tag{5.78}
$$

Substituting Equations (5.72) and (5.73) into (5.78) gives:

$$
\begin{aligned}
f^j_{\phi\phi}(2x+p,2y+q) \quad = \quad & (1-q)\left[(1-p)\sum_{k,m}(\widetilde{h}_k \otimes \widetilde{h}_m)\left\{(1-p)\, f^{j-1}_{\phi\phi}(x-k,y-m)+\right.\right.\\
& \left.(p)\, f^{j-1}_{\psi\phi}(x-k,y-m)\right\}+\\
& (p)\sum_{k,m}(\widetilde{h}_k \otimes \widetilde{g}_m)\left\{(1-p)\, f^{j-1}_{\phi\phi}(x-k,y-m)+\right.\\
& \left.\left.(p)\, f^{j-1}_{\psi\phi}(x-k,y-m)\right\}\right]+\\
& (q)\left[(1-p)\sum_{k,m}(\widetilde{g}_k \otimes \widetilde{h}_m)\left\{(1-p)\, f^{j-1}_{\phi\psi}(x-k,y-m)+\right.\right.\\
& \left.(p)\, f^{j-1}_{\psi\psi}(x-k,y-m)\right\}+\\
& (p)\sum_{k,m}(\widetilde{g}_k \otimes \widetilde{g}_m)\left\{(1-p)\, f^{j-1}_{\phi\psi}(x-k,y-m)+\right.\\
& \left.\left.(p)\, f^{j-1}_{\psi\psi}(x-k,y-m)\right\}\right].
\end{aligned} \tag{5.79}
$$

Using similar arguments for like $(1-p)$, (p) terms in Equation (5.79), the reconstruction algorithm simplifies to:

$$\begin{aligned} f^{j}_{\phi\phi}(2x+p,2y+q) \quad = \quad & (1-q)\left[(1-p)\sum_{k,m}(\widetilde{h}_k\otimes\widetilde{h}_m)\, f^{j-1}_{\phi\phi}(x-k,y-m)+\right.\\ & \left.(p)\sum_{k,m}(\widetilde{h}_k\otimes\widetilde{g}_m)\, f^{j-1}_{\psi\phi}(x-k,y-m)\right]+\\ & (q)\left[(1-p)\sum_{k,m}(\widetilde{g}_k\otimes\widetilde{h}_m)\, f^{j-1}_{\phi\psi}(x-k,y-m)+\right.\\ & \left.(p)\sum_{k,m}(\widetilde{g}_k\otimes\widetilde{g}_m)\, f^{j-1}_{\psi\psi}(x-k,y-m)\right], \end{aligned} \tag{5.80}$$

which is equivalent to (5.75). Equations (5.75) and (5.80) roughly state that $f^{j}_{\phi\phi}$ is reconstructed from the expansion of the f^{j-1} functions at resolution level $j-1$ by the dual (bi-orthogonal) filters $\{\widetilde{h}_k\},\{\widetilde{g}_k\}$ representing scale and wavelet bases generated by $\widetilde{\phi}$, $\widetilde{\psi}$. Since $f^{j}_{\phi\phi}$ was originally projected by the 2D bases functions $\phi\phi$, $\phi\psi$, $\psi\phi$, and $\psi\psi$, represented by the tensor products of $\{h_k\}$ and $\{g_k\}$, producing the four lower resolution level functions $f^{j-1}_{\phi\phi}$, $f^{j-1}_{\phi\psi}$, $f^{j-1}_{\psi\phi}$, and $f^{j-1}_{\psi\psi}$, $f^{j}_{\phi\phi}$ is faithfully reconstructed provided $(\widetilde{\phi},\widetilde{\psi})$, (ϕ,ψ) satisfy (bi-)orthogonality conditions as specified in §5.2 and §5.4.3. Equation (5.75) also makes intuitive sense when Equations (5.68)–(5.71) are considered in matrix form:

$$\begin{aligned} \mathbf{f}^{j-1}_{\phi\phi} = (\mathbf{H}\otimes\mathbf{H})\mathbf{f}^{j}_{\phi\phi}, \quad \mathbf{f}^{j-1}_{\psi\phi} = (\mathbf{G}\otimes\mathbf{H})\mathbf{f}^{j}_{\phi\phi},\\ \mathbf{f}^{j-1}_{\phi\psi} = (\mathbf{H}\otimes\mathbf{G})\mathbf{f}^{j}_{\phi\phi}, \quad \mathbf{f}^{j-1}_{\psi\psi} = (\mathbf{G}\otimes\mathbf{G})\mathbf{f}^{j}_{\phi\phi}. \end{aligned}$$

To reconstruct $\mathbf{f}^{j}$, both sides of each equation should be left-multiplied by some tensor product combination of $\widetilde{\mathbf{H}}$ and $\widetilde{\mathbf{G}}$. If the decomposition matrices are orthogonal, then each reconstruction matrix is the transpose of the decomposition matrix. That is,

$$\begin{aligned} \mathbf{f}^{j-1}_{\phi\phi} = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{H}})^T\mathbf{f}^{j}_{\phi\phi}, \quad \mathbf{f}^{j-1}_{\phi\phi} = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{H}})^T\mathbf{f}^{j}_{\psi\phi},\\ \mathbf{f}^{j-1}_{\phi\phi} = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{G}})^T\mathbf{f}^{j}_{\phi\psi}, \quad \mathbf{f}^{j-1}_{\phi\phi} = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{G}})^T\mathbf{f}^{j}_{\psi\psi}. \end{aligned}$$

In the orthogonal case, $\mathbf{H}=\widetilde{\mathbf{H}}$ and $\mathbf{G}=\widetilde{\mathbf{G}}$, and

$$\begin{aligned} (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{H}})^T = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{H}}), \quad (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{H}})^T = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{G}}),\\ (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{G}})^T = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{H}}), \quad (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{G}})^T = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{G}}), \end{aligned}$$

giving

$$\begin{aligned} \mathbf{f}^{j}_{\phi\phi} = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{H}})\mathbf{f}^{j-1}_{\phi\phi}, \quad \mathbf{f}^{j}_{\phi\phi} = (\widetilde{\mathbf{H}}\otimes\widetilde{\mathbf{G}})\mathbf{f}^{j-1}_{\psi\phi},\\ \mathbf{f}^{j}_{\phi\phi} = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{H}})\mathbf{f}^{j-1}_{\phi\psi}, \quad \mathbf{f}^{j}_{\phi\phi} = (\widetilde{\mathbf{G}}\otimes\widetilde{\mathbf{G}})\mathbf{f}^{j-1}_{\psi\psi}, \end{aligned}$$

which coincides with the order of the tensor product terms in (5.75). Note that each tensor product transpose is not the tensor product's inverse. That is, the individual matrix product components above do not each guarantee perfect reconstruction. However, the appropriate row- and column-permutation of the combination of the tensor product components will create the appropriate perfect (orthogonal) reconstruction matrix. To illustrate, consider the matrix $(\mathbf{M}^j)^{-1}$ defined by:

$$(\mathbf{M}^j)^{-1} = ((\widetilde{\mathbf{H}} \otimes \widetilde{\mathbf{H}}) \bowtie_r (\widetilde{\mathbf{G}} \otimes \widetilde{\mathbf{H}})) \bowtie_c ((\widetilde{\mathbf{H}} \otimes \widetilde{\mathbf{G}}) \bowtie_r (\widetilde{\mathbf{G}} \otimes \widetilde{\mathbf{G}})).$$

Provided the appropriate (bi-)orthogonality conditions hold, matrix $(\mathbf{M}^j)^{-1}$ is symmetric-orthogonal, being the transpose and inverse of the decomposition matrix $\mathbf{M}^j$ (cf.§5.6). In this case, the reason for the tensor product terms in the reconstruction Equation (5.75) being the transpose of the tensor products in the decomposition Equation (5.68) can be seen as a constraint on the orthogonality of $\mathbf{M}^j$. This is clearly seen in the case of the Haar filters where $\mathbf{M}^j$ is given below:

$$\begin{aligned}
\mathbf{M}^j &= \left[\begin{array}{c|c} HH & HG \\ \hline GH & GG \end{array}\right] = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \\
&= ((\widetilde{\mathbf{H}} \otimes \widetilde{\mathbf{H}}) \bowtie_r (\widetilde{\mathbf{G}} \otimes \widetilde{\mathbf{H}})) \bowtie_c ((\widetilde{\mathbf{H}} \otimes \widetilde{\mathbf{G}}) \bowtie_r (\widetilde{\mathbf{G}} \otimes \widetilde{\mathbf{G}})) = (\mathbf{M}^j)^{-1}.
\end{aligned}$$

The matrix $\mathbf{M}^j$ is its own inverse save for the resolution scale factor 2^j. Reconstruction of the image f, built by applying reconstruction filters $\widetilde{h}, \widetilde{g}$, is shown schematically in Figure 19. Row- and column-interleave

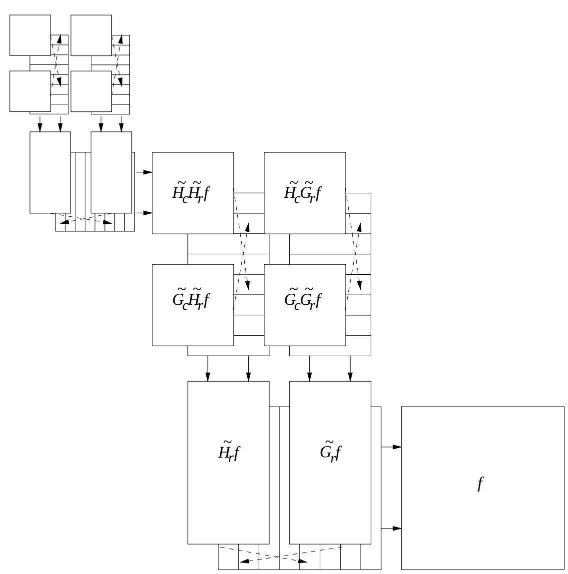


Fig. 19. Non-standard 2D pyramidal reconstruction.

operations are denoted by dashed arrows.

The 3D filters corresponding to the 3D wavelet bases described in §5.5, are generated by the non-standard 3D wavelet basis construction using tensor products:

$$\begin{aligned}
HHH &= H \otimes H \otimes H^T, \\
HHG &= H \otimes H \otimes G^T, \\
HGH &= H \otimes G \otimes H^T, \\
HGG &= H \otimes G \otimes G^T, \\
GHH &= G \otimes H \otimes H^T, \\
GHG &= G \otimes H \otimes G^T, \\
GGH &= G \otimes G \otimes H^T, \\
GGG &= G \otimes G \otimes G^T.
\end{aligned}$$

For example, the three-dimensional Haar filter GHG is a 2×4 filter derived below:

$$\begin{aligned}
GHG = \psi \otimes \phi \otimes \psi^T &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \otimes \left(\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \right) \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} & \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} & -\frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & -\frac{1}{2\sqrt{2}} & -\frac{1}{2\sqrt{2}} \\ -\frac{1}{2\sqrt{2}} & -\frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} \end{bmatrix}.
\end{aligned}$$

Multiplying the filter by the dyadic normalization factor $2^{j/2}$ in each dimension, that is by $2^{2j/2}$ or 2 at one resolution level, the filter becomes:

$$GHG = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}.$$

Considering the application of the filter on two consecutive video frames, the filter can be represented by the two 2×2 spatial templates:

$$GHG = \begin{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} & -\frac{1}{\sqrt{2}} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \end{bmatrix}$$

$$\rightsquigarrow \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_t}, \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}_{G_{t+1}},$$

where the normalization factor is implicit and the subscripts G_t, G_{t+1} denote the application of the appropriate temporal element of G. To help visualize the temporal filter application, Figure 20 shows the correspondence between filter elements. This representation is introduced only for convenience since it facilitates the representation of the three-dimensional gradient components. The remaining seven filters are derived in a similar manner.

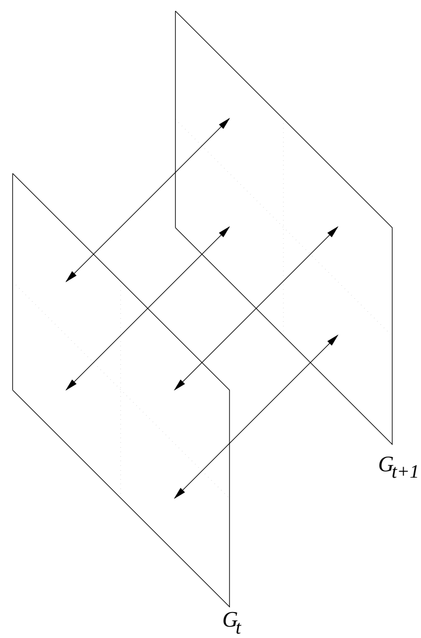


Fig. 20. Visualization of temporal filter element application.

The decomposition relations describing the non-standard decomposition of a 3D digital signal represented by a sequence of 2D images are:

$$\begin{aligned}
f_{\phi}^{j-1}(x,y,t) &= \sum_k h_k f_{\phi\phi\phi}^{j}(x,y,2t+k) & f_{\psi}^{j-1}(x,y,t) &= \sum_k g_k f_{\phi\phi\phi}^{j}(x,y,2t+k) \\
f_{\phi_r\phi}^{j-1}(x,y,t) &= \sum_k h_k f_{\phi}^{j-1}(x,2y+k,t) & f_{\phi_r\psi}^{j-1}(x,y,t) &= \sum_k h_k f_{\psi}^{j-1}(x,2y+k,t) \\
f_{\psi_r\phi}^{j-1}(x,y,t) &= \sum_k g_k f_{\phi}^{j-1}(x,2y+k,t) & f_{\psi_r\psi}^{j-1}(x,y,t) &= \sum_k g_k f_{\psi}^{j-1}(x,2y+k,t) \\
f_{\phi\phi\phi}^{j-1}(x,y,t) &= \sum_k h_k f_{\phi_r\phi}^{j-1}(2x+k,y,t) & f_{\phi\phi\psi}^{j-1}(x,y,t) &= \sum_k h_k f_{\phi_r\psi}^{j-1}(2x+k,y,t) \\
f_{\psi\phi\phi}^{j-1}(x,y,t) &= \sum_k g_k f_{\phi_r\phi}^{j-1}(2x+k,y,t) & f_{\psi\phi\psi}^{j-1}(x,y,t) &= \sum_k g_k f_{\phi_r\psi}^{j-1}(2x+k,y,t)
\end{aligned}$$

$$f^{j-1}_{\phi\psi\phi}(x,y,t) = \sum_k h_k f^{j-1}_{\psi_r\phi}(2x+k,y,t) \quad f^{j-1}_{\phi\psi\psi}(x,y,t) = \sum_k h_k f^{j-1}_{\psi_r\psi}(2x+k,y,t)$$
$$f^{j-1}_{\psi\psi\phi}(x,y,t) = \sum_k g_k f^{j-1}_{\psi_r\phi}(2x+k,y,t) \quad f^{j-1}_{\psi\psi\psi}(x,y,t) = \sum_k g_k f^{j-1}_{\psi_r\psi}(2x+k,y,t)$$

where $\{h_k\},\{g_k\}$ are the one-dimensional low- and high-pass filters. The non-standard DWT first temporally subsamples the sequence images of the lower (finer) resolution level's spatial average sequence (denoted by $f^{j}_{\phi\phi\phi}$) to generate the temporary upper resolution level images f^{j-1}_{ϕ} and f^{j-1}_{ψ}. These images will be of the same dimension as the images at the lower resolution level, but half of them will contain averaged temporal values, the other half will contain temporal difference values. Each temporal average and difference image is then decomposed at one level by the 2D DWT. The lower resolution level sequence is transformed into $n/2$ temporal average frames $f^{j-1}_{\cdot\cdot\phi}$, and $n/2$ temporal difference frames $f^{j-1}_{\cdot\cdot\psi}$. These frames are then permuted as in the 1D decomposition using the organization given in (5.65) and (5.66):

(5.81) $$\{W f(x)\}(j-1) = f^{j-1}_{\cdot\cdot\phi}(1),\ldots,f^{j-1}_{\cdot\cdot\psi}(n/2),\ldots,f^{j-1}_{\cdot\cdot\psi}(n)$$

The spatiotemporal smooth (or averaged) subimages $f^{j}_{\phi\phi\phi}$ are recursively processed at each stage of the decomposition. The transformed image sequence contains the global spatiotemporal average at the top of the (volume) pyramid. The decomposition is shown schematically in Figure 21, where the resultant wavelet transform volume is represented by a segmented cube resembling a nonuniform octtree data structure. The subscript t represents the temporal subsampling operation performed on sequence images, i.e., $f^{j-1}_{\phi} = H_t * f^{j}$. A pictorial example is shown in Figure 22.

The video sequence is synthesized by a recursive process of adding detail information to the average (smoothed) subimages in order to reconstruct the next level's temporal average subimages. At each level of reconstruction, frames must be interleaved in the temporal dimension prior to the spatial 2D wavelet reconstruction:

$$f^{j-1}_{\cdot\cdot\phi\bowtie\cdot\cdot\psi}(x,y,2t+p) \quad = \quad (1-p)\,f^{j-1}(x,y,t) + (p)\,f^{j-1}(x,y,t).$$

After permutation, each frame is spatially reconstructed by the 2D DWT, following (5.75), giving the two temporally subsampled frames $f^{j-1}_{\phi}(x,y,t)$ and $f^{j-1}_{\psi}(x,y,t)$. Each such pair of frames is then reconstructed as in the 1D DWT case:

$$f^{j}_{\phi\phi\phi}(x,y,2t+p) = (1-p)\sum_k \widetilde{h}_k f^{j-1}_{\phi\bowtie\psi}(x,y,2t-k) + (p)\sum_k \widetilde{g}_k f^{j-1}_{\phi\bowtie\psi}(x,y,2t-k).$$

The above description of the DWT assumes that decomposition in each dimension is possible to the same extent, i.e., the above multidimensional transform is *isotropic* with respect to spatiotemporal dimensions. The maximum level of isotropic signal decomposition coincides with the dimension of *lowest* magnitude. To illustrate, consider a video sequence of 4 frames, each frame of size 640×480. The sequence can only be decomposed isotropically to 2 levels since there is an insufficient number of sequence frames to decompose the signal any further in the temporal dimension. Conversely, given 1024 frames, a temporal decomposition

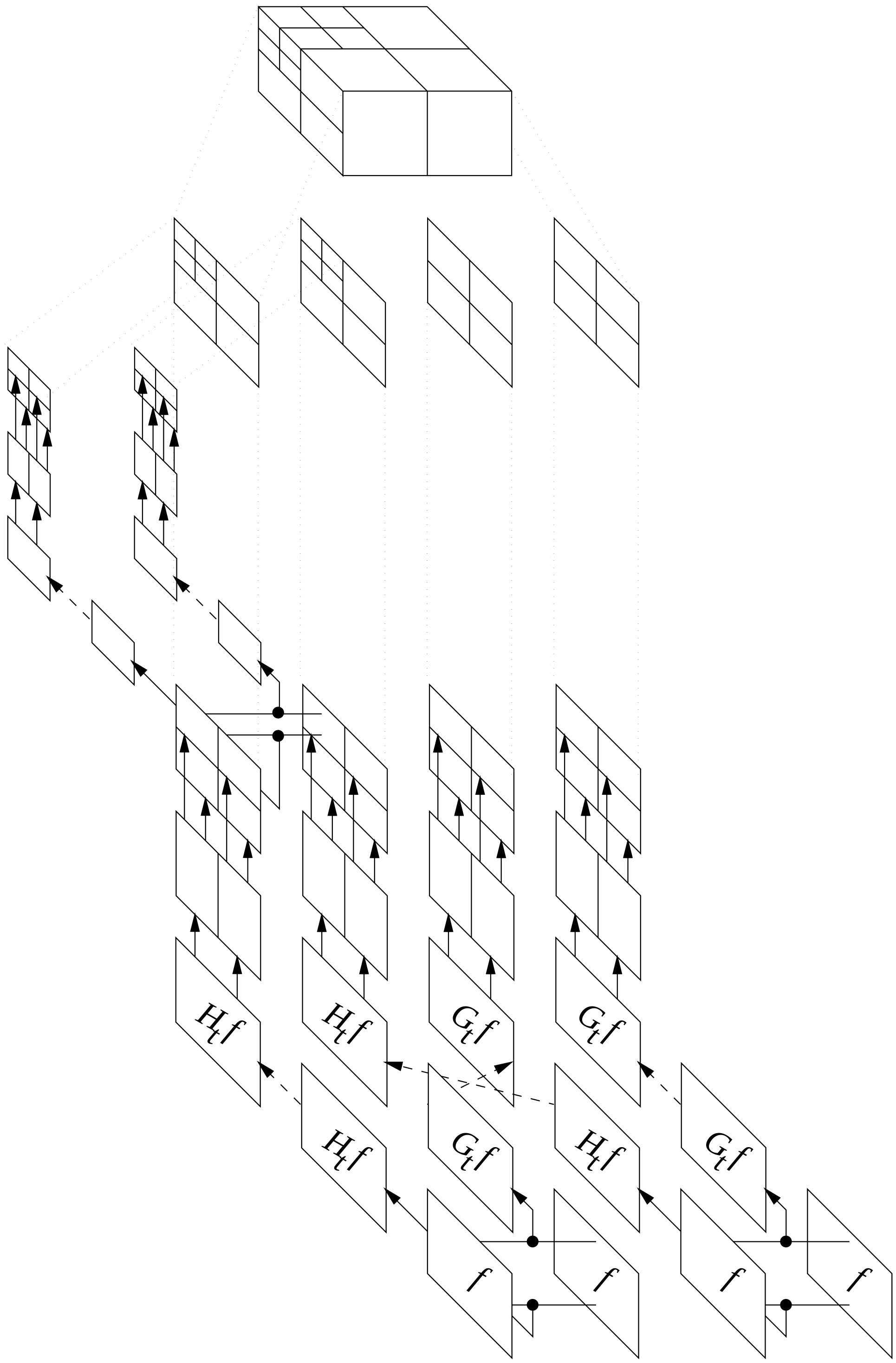
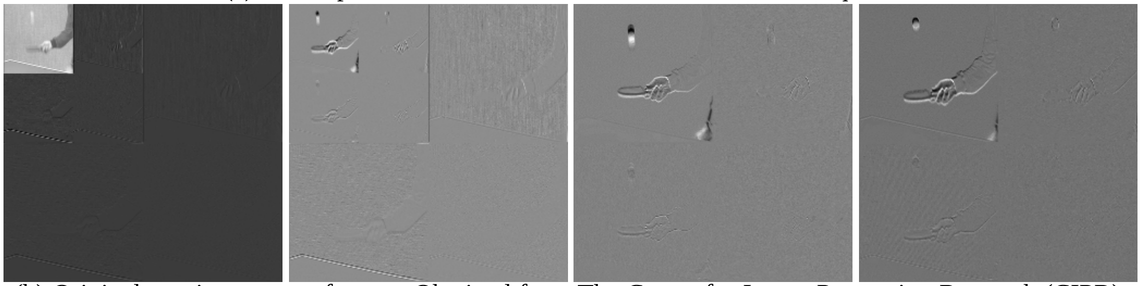


Fig. 21. Schematic non-standard 3D pyramidal wavelet decomposition.

(a) Decomposition of the first four frames of the *tennis* sequence.



(b) Original *tennis* sequence frames. Obtained from The Center for Image Processing Research (CIPR), an Internet public domain archive (ftp://ipl.rpi.edu/pub/image/sequence/tennis/).

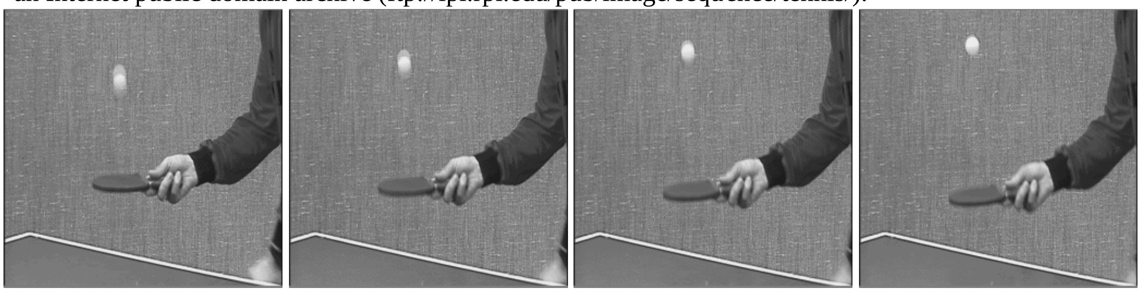


Fig. 22. Non-standard 3D pyramidal discrete wavelet decomposition.

of 10 levels is not possible since there is insufficient information in the x-dimension beyond the 8^{th} level of decomposition (the x-dimension is reduced to 1 pixel). Alternatively, it is possible to decompose the 4-frame sequence *anisotropically* to 2 levels in the temporal dimension, then decompose each frame spatially to 8 levels but in this case the decomposition is incomplete in terms of partial derivative information (see §5.9). The complete spatiotemporal decomposition governed by the dimension of least magnitude gives at least a full first-order representation of the signal. That is, depending on the choice of wavelet, all three partial derivatives $\partial/\partial x$, $\partial/\partial y$, $\partial/\partial t$ are available. This enables localization of multiscale, three-dimensional edges in video (see §5.8). This is a significantly powerful method of video analysis since it extends inspection of the temporal domain over a longer temporal duration than just two frames. Two-frame motion detection has been extensively studied in the context of motion-compensated video encoding, but temporal analysis over many frames has not been widely utilized. Furthermore, the complete 3D DWT offers second-order information, i.e., the partial derivatives $\partial/\partial x \partial y$, $\partial/\partial x \partial t$, $\partial/\partial y \partial t$, and $\partial/\partial x \partial y \partial t$ are all present in the transformed signal, although it is not clear at this point how this information can be used to enhance video analysis applications relying on first-order derivative information. For example, three-dimensional edges can be located by examining either the set of first-order partials, or the set of second-order partial derivatives (detection of zero-crossings), in which case the set not used appears redundant.

5.8 Multidimensional Multiscale Edge Detection

Edges, or sharp variation points in general, can be located in signals through the use of Mallat's multiscale edge detection algorithm. If ψ is chosen such that it approximates the first derivative of a smoothing function θ, as described in §5.3, then the 2D filters HG, GH constitute the gradient components $\partial\theta(x,y)/\partial x$, $\partial\theta(x,y)/\partial y$, respectively. This is due to the tensor product construction of the filters. The elements of HG are two rotated (transposed) copies of G multiplied by scalar elements of H. Similarly, the elements of GH are copies of G scaled by elements of H. Considering the 1D filter G as the 1D vector $[\partial\theta/\partial x]$, the transpose of G is the derivative of θ in the y-direction, i.e., $[\partial\theta/\partial y] = [\partial\theta/\partial x]^T$. The scalar multiplier cancels if the original filters H, G are orthonormal, and the filters HG, GH are left with components of the gradient:

$$HG = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial x} \end{bmatrix}; \; GH = \begin{bmatrix} \frac{\partial f}{\partial y} \\ \\ \frac{\partial f}{\partial y} \end{bmatrix}.$$

Considering the Haar wavelet, which is a first derivative operator, the second resolution level filters are:

$$HGHH = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}; \; GHHH = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

Save for scale factors, these resemble (directionally, at least) the well known Sobel or Prewitt gradient operators used in edge detection (for comparison, see the discussions on gradient operators in [GW87, pp.336-338], [Sch89, pp.146-150], and [Jai89, pp.348-349]). The locations of the gradient components in the 2D-DWT of an image frame are schematically shown Figure 23(a). Due to the finite sampling of a 2D image $f(x,y)$, the angle defined by (5.26) can be quantized to octants specifying 8 possible neighbors, relative to the point at scale j and location (x_0, y_0), namely $\{Mf(x_0+\Delta x, y_0+\Delta y)\}(j)$, where $(\Delta x, \Delta y)$ are arranged as:

(-1,-1)	(0,-1)	(1,-1)
(-1, 0)	(0, 0)	(1, 0)
(-1, 1)	(0, 1)	(1, 1)

defining 4 planar directions corresponding to the compass directions N-S, NE-SW, E-W, and SE-NW. In the two-dimensional case, Equation (5.26) directly specifies one of the 4 directions used to identify pixel neighbor for determination of the modulus maxima.

Extending Mallat's algorithm to three dimensions, gradient components $\partial\theta(x,y,t)/\partial x$, $\partial\theta(x,y,t)/\partial y$, and $\partial\theta(x,y,t)/\partial t$, are represented by the 3D filters HHG, HGH and GHH, respectively:

$$HHG = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_t}, \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_{t+1}},$$

$$HGH = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_t}, \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_{t+1}},$$

$$GHH = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}_{G_t}, \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}_{G_{t+1}}.$$

The conceptual volumetric representation of the 3D-DWT is shown in Figure 23(b). The locations of the gradient components in the two resolution level 3D-DWT of four frames are schematically shown in Figure 24. Quantizing the angle defined by Equations (5.28)–(5.30), 26 possible neighbors are specified relative to the point at scale j and location (x_0, y_0, t_0), namely $\{Mf(x_0+\Delta x, y_0+\Delta y, t_0+\Delta t)\}(j)$, where $(\Delta x, \Delta y, \Delta t)$ are arranged as:

(-1,-1,-1)	(0,-1,-1)	(1,-1,-1)
(-1, 0,-1)	(0, 0,-1)	(1, 0,-1)
(-1, 1,-1)	(0, 1,-1)	(1, 1,-1)

(-1,-1, 0)	(0,-1, 0)	(1,-1, 0)
(-1, 0, 0)	(0, 0, 0)	(1, 0, 0)
(-1, 1, 0)	(0, 1, 0)	(1, 1, 0)

(-1,-1, 1)	(0,-1, 1)	(1,-1, 1)
(-1, 0, 1)	(0, 0, 1)	(1, 0, 1)
(-1, 1, 1)	(0, 1, 1)	(1, 1, 1)

along 13 cubic (voxel) directions, depicted in Figure 25.

In the three-dimensional case, Equations (5.28)–(5.30) do not readily specify the 13 (voxel) neighbor directions. Instead, the values of the first-order partial derivatives must be examined to first determine the relevant angular plane. This is accomplished by inspecting the three first-order partials for zero (or near-zero) values. Since there are only three values, there are $2^3 = 8$ possibilities for zero-value combinations. Labeling non-zero values as 1 for ease of notation, Table 6 lists the possible directions given the 8 possible gradient value combinations. Referring to each case associated by its binary value, the 0^{th} case identifies a uniform region. This is a common property shared by gradient operators [Jai89, p.349]. Case 1 trivially specifies direction 1. In the 2^{nd}, 4^{th}, and 6^{th} cases, Equation (5.26) can be used directly to determine the relevant direction in the xy-plane. In case 3, only $\partial/\partial x$ is zero which suggests the gradient is in the yt-plane. Equation (5.30) can be used to determine whether the gradient falls along direction 4 or 8. Case 5 is similar to case 3 in that only $\partial/\partial y$ is zero so that Equation (5.29) can be used to determine gradient direction 2 or 6, both in the xt-plane. Case 7 is the most complicated since the gradient direction has components in both xt- and yt-planes. One of the four directions is found by projecting the gradient onto each plane in turn.

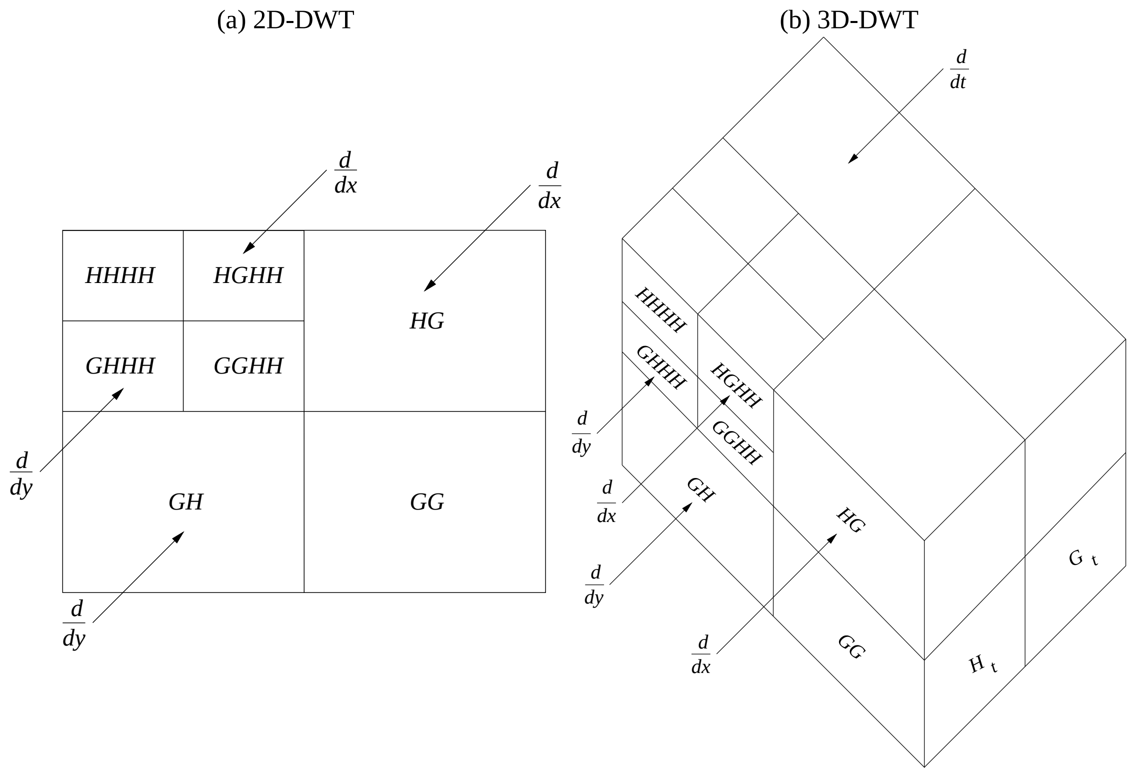


Fig. 23. 2D- and 3D-DWT multiresolution quadrants and octants, with gradient components.

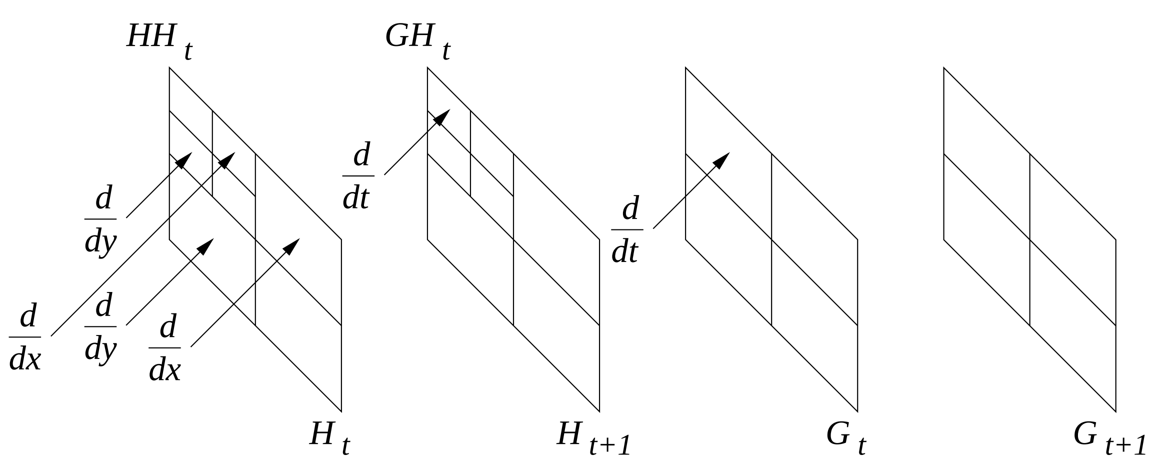


Fig. 24. Schematic 3D-DWT with gradient components.

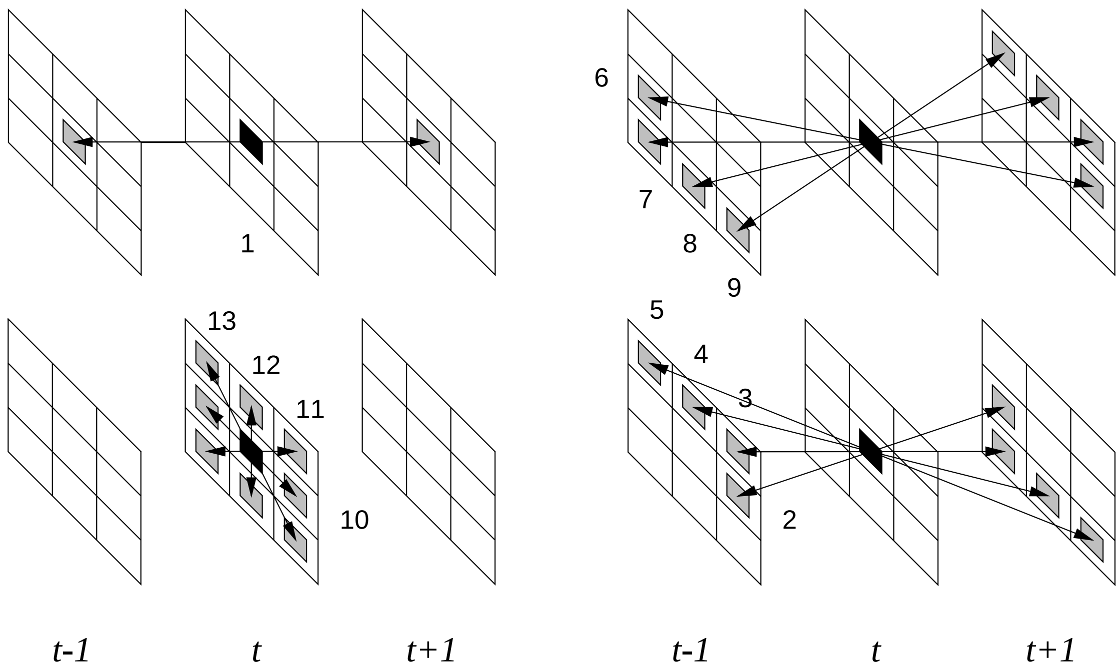


Fig. 25. Modula maxima planar (pixel) and cubic (voxel) neighbors.

TABLE 6
Three-dimensional direction identification based on first-order partial derivatives.

$\partial/\partial x$	$\partial/\partial y$	$\partial/\partial t$	Direction
0	0	0	$\{\}$
0	0	1	$\{1\}$
0	1	0	$\{12\}$
0	1	1	$\{4,8\}$
1	0	0	$\{10\}$
1	0	1	$\{2,6\}$
1	1	0	$\{10,11,12,13\}$
1	1	1	$\{3,5,7,9\}$

Given the modulus values as calculated by Equations (5.25) and (5.27) in 2D and 3D, respectively, along with the directional neighbor locations defined above, modulus maxima are located according to Equations (5.23) and (5.24) in both two- and three-dimensions. If the modulus satisfies both of these equations, then a record at the specific location is stored by the value $\{mf\}(j)$ set to the value of the modulus if the modulus is a local maxima, or 0. That is, in two dimensions, defining $\max\{Mf(x_0,y_0)\}(j)$ as:

$$\{Mf(x_0+\Delta_l x, y_0+\Delta_l y)\} \leq \{Mf(x_0,y_0)\} \geq \{Mf(x_0+\Delta_r x, y_0+\Delta_r y)\}, \text{ and}$$

$$\begin{cases} \{Mf(x_0,y_0)\} > \{Mf(x_0+\Delta_l x, y_0+\Delta_l y)\}, & \text{or} \\ \{Mf(x_0,y_0)\} > \{Mf(x_0+\Delta_r x, y_0+\Delta_r y)\}. \end{cases}$$

with the resolution level index made implicit, then $\{mf(x,y)\}(j)$ is defined as:

$$\{mf(x,y)\}(j) = \begin{cases} \{Mf(x,y)\}(j) & \text{if } \max\{Mf(x_0,y_0)\}(j) \\ 0 & \text{otherwise,} \end{cases}$$

where the left and right neighbors $\{Mf(x+\Delta_l x, y+\Delta_l y)\}(j)$, $\{Mf(x+\Delta_r x, y+\Delta_r y)\}(j)$, are identified by one of the four directions as given above. In three dimensions, $\max\{Mf(x_0,y_0,t_0)\}(j)$ is defined similarly with the left and right neighbors $\{Mf(x+\Delta_l x, y+\Delta_l y, t+\Delta_l t)\}(j)$, $\{Mf(x+\Delta_r x, y+\Delta_r y, t+\Delta_r t)\}(j)$, identified by one of the above specified thirteen directions.

During modulus maxima detection, a 2D data structure is created to hold values $\{Mf(x,y)\}(j)$, $\{Af(x,y)\}(j)\}$, and $\{mf(x,y)\}(j)$, as shown schematically in Figure 26(a). An example of modula maxima detection in two dimensions is shown in Figures 26(b), where the image has been globally normalized and gamma-corrected ($\gamma = 3$) to facilitate display. In the 2D case, the unmodified 2D wavelet transform is needed for perfect

(a) 2D-DWT modula maxima information storage.

(b) Modula maxima in 2-level decomposition of *cnn* image.

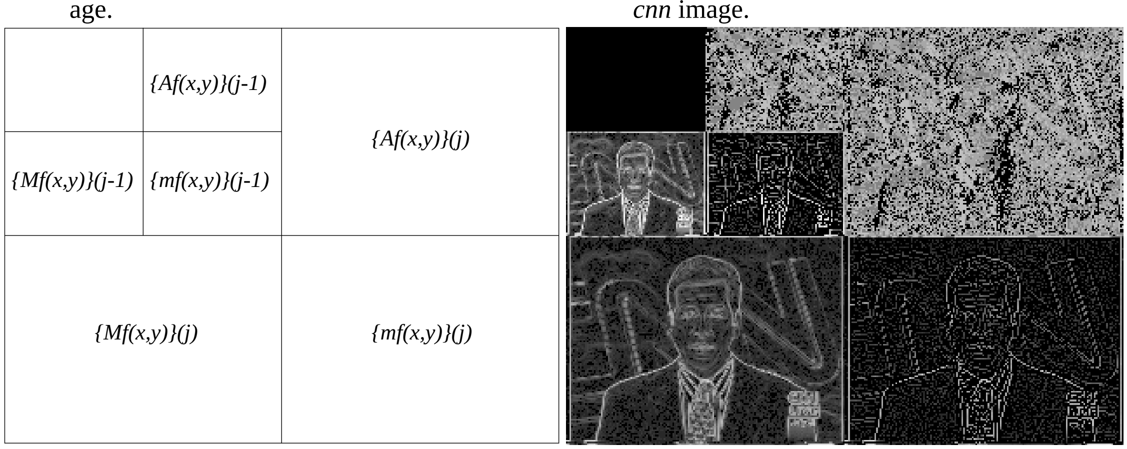
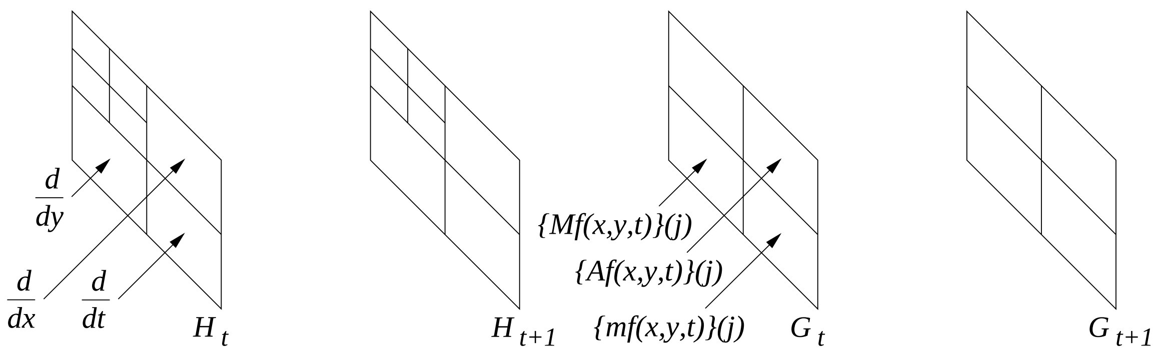


Fig. 26. 2D modula maxima detection.

reconstruction of the image. The image cannot be readily reconstructed from the maxima modulus information, although Mallat has developed an iterative algorithm that almost achieves perfect reconstruction in most cases [MZ92a]. (Meyer gives a counterexample to Mallat's conjecture in [Mey93, §8].) Typically a second image matrix array is allocated for this purpose.

In three dimensions, the 3D data structure is rearranged to hold values $\{Mf(x,y)\}(j)$, $\{Af(x,y)\}(j)\}$, and $\{mf(x,y)\}(j)$, as well as $\partial/\partial x$, $\partial/\partial y$, $\partial/\partial t$. This organization is shown schematically in Figure 27(a). An example of temporal modulus maxima detection (in a three-dimensional video sequence) is shown in Figure 27(b), where individual frames have been globally normalized and gamma-corrected ($\gamma = 3$) to facilitate display. Analogous to the 2D case, unmodified 3D wavelet transform information is needed for perfect re-

(a) Schematic 3D-DWT modula maxima information storage.



(b) Modula maxima in 2-level decomposition of the first four frames of the *tennis* sequence.

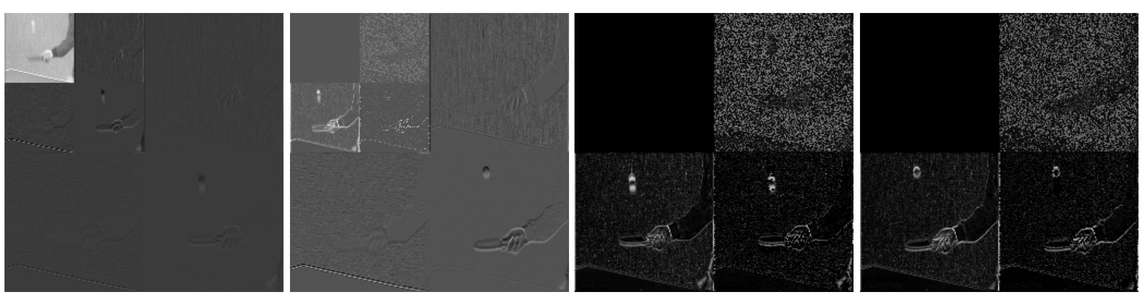


Fig. 27. 3D modula maxima detection.

construction. Best results are obtained if wavelet transform coefficients can be stored in memory with double precision. It is desirable to store both modulus maxima and wavelet coefficient information in memory. In the case of a 2D static image, a copy of the image, for the purposes of modulus maxima information storage, can generally be made due to sufficient memory availability. Unlike the two-dimensional image, duplicating an entire sequence in memory may be problematic due to memory capacity limitations. Fortunately, second-order information is not needed for modula maxima detection.[10] This allows the data quadrants to be rearranged such that the values needed for modula maxima detection can be accumulated in place of the second-order information. Note that this organization prevents straightforward reconstruction of the video sequence, but provides storage of both modulus maxima and first-order derivative information.

5.9 Anisotropic Multidimensional Discrete Wavelet Transform

The Discrete Wavelet Transform as described in §5.7 performed equal extent decomposition in each dimension. That is, the preceding discussion centered on an *isotropic* wavelet decomposition of a multidimensional signal. In general, signals may be transformed maximally in each dimension if each dimension is considered individually. In this sense, the transformation is *anisotropic* since the decomposition varies along each dimension. In this section, anisotropic decomposition of video sequences is discussed where decomposition in the spatial dimension is dissociated from the temporal dimension.

In dealing with video, the sequence may be considered as a one-dimensional temporal signal composed of two-dimensional elements. Accordingly, the sequence may first be fully decomposed in the temporal dimension on a per-pixel basis following Equations (5.62) and (5.63), i.e.,

$$\begin{aligned} f_\phi^{j-1}(x,y,t) &= \sum_k h_k f_\phi^j(x,y,2t+k), \\ f_\psi^{j-1}(x,y,t) &= \sum_k g_k f_\phi^j(x,y,2t+k), \end{aligned}$$

giving the DWT in the temporal direction:

$$\begin{aligned} &\{Wf(x,y,t)\}_t(j-1) = \\ &\quad f_{\phi_t}^{j-1}(x,y,1), f_{\psi_t}^{j-1}(x,y,2), \ldots, f_{\phi_t}^{j-1}(x,y,n-1), f_{\psi_t}^{j-1}(x,y,n). \end{aligned} \tag{5.82}$$

As in the one-dimensional case, the sequence frames are permuted so that the first $n/2$ frames, containing low-pass coefficients, are decomposed recursively. The fully transformed video sequence contains the global

[10] As an alternative to first-order derivative edge detection, directional information of edges can be obtained by searching zero-crossings of the second-order derivative along r for each direction θ, since

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial r} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial r} = f_x \cos\theta + f_y \sin\theta,$$

where r is the gradient direction vector, and

$$\frac{\partial^2 f}{\partial r^2} = \frac{\partial f_x}{\partial r}\cos\theta + \frac{\partial f_y}{\partial r}\sin\theta = \frac{\partial^2 f}{\partial x^2}\cos^2\theta + 2\frac{\partial^2 f}{\partial x \partial y}\sin\theta\cos\theta + \frac{\partial^2 f}{\partial y^2}\sin^2\theta.$$

See [Jai89, p.348,p.353].

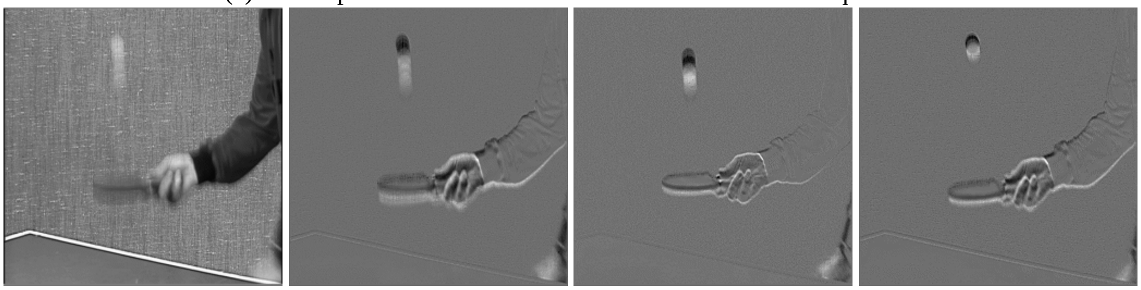
average in the first frame, with the next 2^j frames containing detail information at each resolution level j. Note that the first frame will contain an image displaying average motion information over the entire sequence. The frames containing difference information will contain motion differences over a specific interval of frames thereby indicating presence of motion over specific frequencies relative to the sampling rate of the video sequence. For example, consider a video sequence recorded at a sampling rate of 18ms, i.e., 55.5Hz, or 55.5 frames per second (fps).[11] The (dyadic) DWT of such a video sequence will contain temporal information of 55.5Hz at 1 level of decomposition (resolution level $j = 1$), 18.5Hz at level $j = 2$, 7.9Hz at level $j = 3$, etc. In general, at the j^{th} decomposition level, the dyadic DWT will contain temporal information at frequency $1/(2^j - 1)s_r$ where s_r is the sampling rate. In other words, $(2^j - 1)s_r$ gives the temporal difference information at level j with units matching the sampling rate (e.g., milliseconds). Considering the sequence captured at the 18ms sampling rate, then at level $j = 1$, the DWT will contain information over 18ms, at level $j = 2$ over 54ms, at level $j = 3$ over 126ms, etc. This is due to the fact that each temporal (dyadic) decomposition obtains difference information over $2^j - 1$ frame intervals, where each interval corresponds to the sampling rate. Such temporal information is extremely valuable in applications such as motion detection.

Having applied the 1D temporal DWT over a sequence of frames, the next natural task is motion detection. This involves detection of motion within sequence frames containing wavelet coefficients at specific (dyadic) frequencies of interest. Using the 55.5Hz video sequence as an example, if fast motion over 18Hz is sought, then sequence frames containing temporal differences at resolution level $j = 2$ must be inspected. Slower motion artifacts under 18.5Hz will be found at higher resolution levels. The problem of locating these artifacts is essentially a two-dimensional problem. In this respect, the 2D DWT (including 2D edge detection) is well suited for analysis. Temporal difference frames should be treated as two-dimensional image frames and the 2D DWT is applied exactly as in §5.7. Two-dimensional edge detection (as described in §5.8) is again applicable on a per-frame basis. The result of such an analysis locates features in video at specific frequencies. Four frames of the *tennis* sequence are shown under 2 levels of temporal decomposition in Figure 28(a). Note that the frames have been interleaved as described in §5.7 so that the first frame contains the global temporal average. Figure 28(b) shows two-dimensional edge detection carried out on each frame of the transformed sequence (compare with Figure 27 in §5.8).

Reconstruction of the sequence is carried out in the reverse order where each frame is processed by the 2D IDWT. The entire sequence is then treated as a one-dimensional signal and the 1D IDWT is applied on a per-pixel basis taking care to properly interleave whole image frames as required (see Equation (5.82)). Using

[11]The standard (NTSC) video rate is 30 fps, 30Hz, or in other words, video sampled at a rate of 33.3ms. This should not be confused with the NTSC *field* rate of 60Hz where a field corresponds to only half a given video *frame*, e.g., even or odd lines of a frame.

(a) Decomposition of the first four frames of the *tennis* sequence.



(b) 2D edge (modulus maxima) detection.

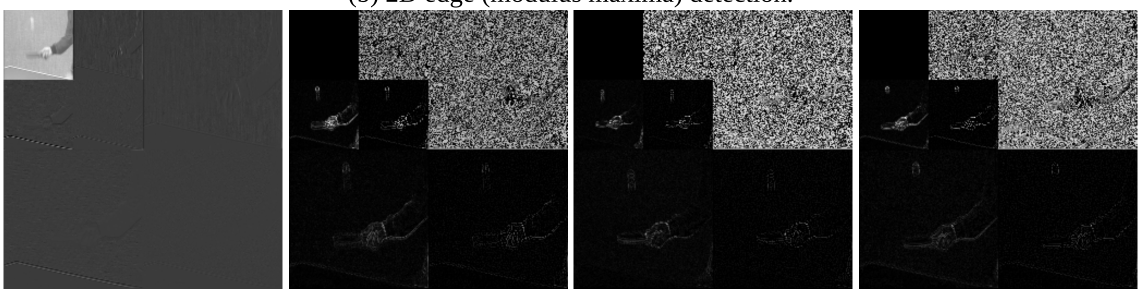


Fig. 28. Anisotropic non-standard 3D pyramidal discrete wavelet decomposition and 2D edge detection.

the interleave operator $\bowtie$, image frames are arranged for reconstruction at level j by:

$$f^{j-1}_{\phi\bowtie\psi}(x,y,2t+p) \quad = \quad (1-p)\,f^{j-1}(x,y,t)+(p)\,f^{j-1}(x,y,t),$$

for $p \in \{0,1\}$. Reconstruction is then written as:

$$f^{j}_{\phi}(x,y,2t+p) = (1-p)\sum_k \widetilde{h}_k\, f^{j-1}_{\phi\bowtie\psi}(x,y,t-k) + (p)\sum_k \widetilde{g}_k\, f^{j-1}_{\phi\bowtie\psi}(x,y,t-k),$$

giving the original function $f^{j}(x,y,t)$.

5.10 Wavelet Interpolation

The Discrete Wavelet Transform can be used to texture map images by selectively scaling wavelet coefficients. Provided appropriate wavelet filters can be found, reconstruction exactly matches linear *MIP-mapping*. MIP-mapping is a well known texture mapping algorithm used extensively in computer graphics [Wil83, WW92, §4.7].[12] MIP-mapping involves preprocessing an image at several resolution levels (decomposition) in order to texture map (reconstruct) an image at variable resolution. In this sense, MIP-mapping falls under the classical pyramid framework for early vision [JR94]. In the present context the relevant feature of MIP-mapping is the multiresolution reconstruction of the image. Specifically, gaze-contingent visual representation of digital imagery, discussed in §IX, relies on wavelet coefficient scaling developed here. In this

[12]The acronym MIP, introduced by Williams, is from the Latin phrase *multum in parvo* meaning “many things in a small place”.

section the MIP-mapping approach is briefly described, then a set of filters, termed *MIP-wavelets*, is derived to match the box filter frequently used in MIP-mapping. Using these filters, it is shown that linear interpolation of scaled wavelet coefficients is equivalent to linear interpolation under MIP-mapping. Although the present discussion is limited to two-dimensional images, the wavelet coefficient scaling method is applicable to multidimensional signals for multiresolution representation.

5.10.1 MIP-Mapping

Given an $N \times N$ image, assuming N is a power of 2 with $n = \log_2 N$, the original image $f^n(x,y)$ is subsampled and smoothed into $n+1$ subimages (or *maps*),

(5.83) $$f^j\left(\left\lfloor \frac{x}{M} \right\rfloor, \left\lfloor \frac{y}{M} \right\rfloor\right) = \frac{1}{M^2} \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} f^n(x+k, y+m), \quad 0 \leq j \leq n,$$

where M is a smoothing filter of size 2^{n-j}, and j is the resolution level. Equation (5.83) generates projections of the original image onto $n+1$ scaled subspaces equivalent to the subspaces generated by the scaling function of the DWT. The subspaces in this instance are scaled analogously to the DWT with resolution level $j = 0$ corresponding to the coarsest resolution level.[13] Equation (5.83) is a slightly different representation from the classical recursive pyramidal approach since each subimage is subsampled directly from the original image f^n, not from the image at the next finer resolution level f^{j+1}. In general, the recursive form of Equation (5.83) is given by:

(5.84) $$f^{j-1}(x,y) = \sum_{k=0}^{M} \sum_{m=0}^{M} h(k,m) f^j(2x+k, 2y+m), \quad 0 < j < n,$$

where $M+1$ is the (constant) width of the convolution kernel. If $h(k,m) = 1/\sqrt{2}$, $\forall k,m$ with $M = 1$, then $\{h(k,m)\}$ is the Haar smoothing filter. Equation (5.84) corresponds to the two-scale relation of the scaling function ϕ, given by (5.43), and is equivalent to the two-dimensional lowpass decomposition relation (5.68) where the scaling filter is the tensor product of the one-dimensional lowpass filter $\{h_k\}$, i.e., $h(k,m) = h_k \otimes h_m$. In general, the smoothing filter should satisfy the following constraints [JR94]:

1. normalization
$$\sum_k \sum_m h(k,m) = 1;$$
2. symmetry
$$h(k,m) = h(M+1-k,m) = h(k,M+1-m) = h(M+1-k,M+1-m) \quad \forall k,m;$$

[13] To scale subspaces in the opposite direction (e.g., the Daubechies convention), each subimage is generated by

$$f^j\left(\left\lfloor \frac{x}{M} \right\rfloor, \left\lfloor \frac{y}{M} \right\rfloor\right) = \frac{1}{M^2} \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} f^0(x+k, y+m), \quad 0 \leq j \leq n,$$

where the filter M is of size 2^j and resolution level $j = 0$ corresponds to the highest resolution.

3. unimodality
$$0 \le h(j,k) \le h(m,n) \text{ for } j \le m < \frac{M}{2} \text{ and } k \le n < \frac{M}{2};$$
4. equal contribution (all pixels of f contribute the same total weight to each pixel of f^j);
5. separability
$$h(k,m) = h(k)h(m).$$

The averaging box filter is often chosen as the smoothing filter with $h(k,m) = 1/M^2, \forall k,m$, where $M = 2^{n-j}$ defines the filter size as well as the filter coefficients. For example, the following subimages are obtained from a 4×4 image:

$$f^1(\frac{x}{2},\frac{y}{2}) = \sum_{k=0}^{1}\sum_{m=0}^{1}\frac{f^2(x+k,y+m)}{4},\ f^0(\frac{x}{4},\frac{y}{4}) = \sum_{k=0}^{3}\sum_{m=0}^{3}\frac{f^2(x+k,y+m)}{16},$$

where $f^2(x,y)$ is the original image. Using the normalized box filter, example subsampled images are shown in Figure 29. The MIP-mapping pyramid is formed by the union of the original image and the set of subsampled images.

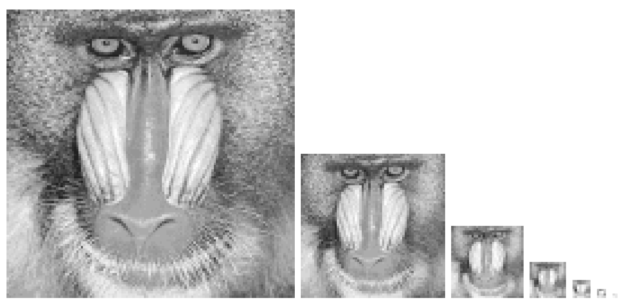


Fig. 29. MIP-map subimages, processed by normalized box filter. Obtained from The Center for Image Processing Research (CIPR), an Internet public domain archive (ftp://ipl.rpi.edu/pub/image/still/usc/bgr/-baboon).

Reconstruction of the image at a given pixel location (x,y) depends on the desired resolution of the pixel. The desired resolution level is bandlimited to the number of decomposed resolution levels (typically the decomposition is dyadic in nature) bounded by the two closest resolution subimages f^{j-1} and f^j. The final pixel value at location (x,y) is calculated as a linear combination of pixel intensities in the pyramid:

$$f(x,y) = (1-p)\,f^{j-1}(\left\lfloor\frac{x}{2^{n-(j-1)}}\right\rfloor,\left\lfloor\frac{y}{2^{n-(j-1)}}\right\rfloor) + (p)\,f^j(\left\lfloor\frac{x}{2^{n-j}}\right\rfloor,\left\lfloor\frac{y}{2^{n-j}}\right\rfloor). \tag{5.85}$$

Equation (5.85) represents linear *inter-map* interpolation, shown schematically for an 8×8 image in Figure 30. In cases where the image is not being mapped onto a flat surface, *intra-map* interpolation may also be used to prevent aliasing artifacts (see [WW92, §4.7.1]). The combination of inter- and intra-map interpolation

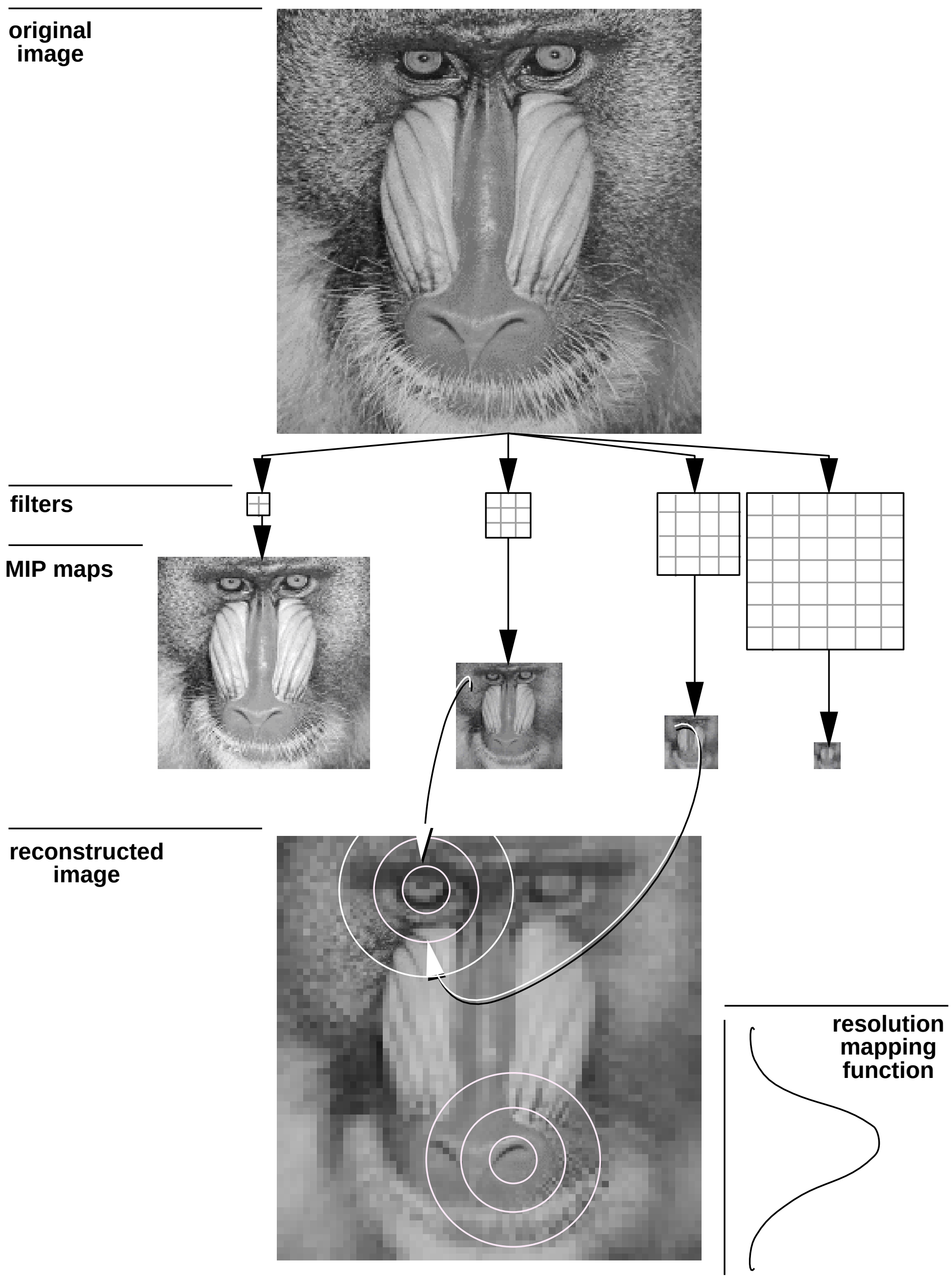


Fig. 30. Depiction of MIP-mapping algorithm.

is called bi-linear interpolation. The wavelet technique equivalent to Equation (5.85) discussed below does not consider bi-linear interpolation.

5.10.2 MIP-Wavelets

To generate wavelet multiscale representations of a given image matching the MIP-map decomposition using the normalized box filter, the lowpass wavelet filter $\{h_k\}$ is set to $\{1/2, 1/2\}$. The detail filter is then the quadrature mirror of $\{h_k\}$, i.e., $\{g_k\} = \{1/2, -1/2\}$. To guarantee perfect reconstruction, MIP-wavelet dual filters are required so that conditions (5.58) and (5.59) are simultaneously satisfied. For filters of length 2, the following equations must hold:

$$h_0\widetilde{h}_0 + h_1\widetilde{h}_1 = 1 \tag{5.86}$$

$$g_0\widetilde{h}_0 + g_1\widetilde{h}_1 = 0. \tag{5.87}$$

Given $(\{h_k\}, \{g_k\})$, $\{\widetilde{h}_k\}$ is derived with:

$$\frac{1}{2}\widetilde{h}_0 + \frac{1}{2}\widetilde{h}_1 = 1, \quad \text{from (5.86), and,} \tag{5.88}$$

$$\frac{1}{2}\widetilde{h}_0 - \frac{1}{2}\widetilde{h}_1 = 0, \quad \text{from (5.87).} \tag{5.89}$$

From (5.89), $\widetilde{h}_0 = \widetilde{h}_1$, and substituting into (5.88),

$$\widetilde{h}_0 = \widetilde{h}_1 = 1.$$

The dual detail filter coefficients are derived from conditions (5.57) and (5.60), which generate equations:

$$g_0\widetilde{g}_0 + g_1\widetilde{g}_1 = 1 \tag{5.90}$$

$$h_0\widetilde{g}_0 + h_1\widetilde{g}_1 = 0. \tag{5.91}$$

Using derived filters $(\{h_k\}, \{g_k\})$, $\{\widetilde{g}_k\}$ is found by:

$$\frac{1}{2}\widetilde{g}_0 + \frac{1}{2}\widetilde{g}_1 = 0, \quad \text{from (5.91), and,} \tag{5.92}$$

$$\frac{1}{2}\widetilde{g}_0 - \frac{1}{2}\widetilde{g}_1 = 1, \quad \text{from (5.90).} \tag{5.93}$$

From (5.92), $\widetilde{g}_0 = -\widetilde{g}_1$, and substituting into (5.93),

$$\widetilde{g}_0 = -\widetilde{g}_1 = 1.$$

MIP-wavelets, with coefficients given in Table 7, are unnormalized versions of the Haar filters. That is, MIP-wavelets are semi-orthogonal Haar wavelets (or pre-wavelets). In fact, normalized Haar filters will generate the same texture mapping at dyadic resolution boundaries, but will lose luminance information between

TABLE 7
MIP-wavelet filters.

k	$2(h_k)$	$2(g_k)$	$2(\tilde{h}_k)$	$2(\tilde{g}_k)$
0	1	1	2	2
1	1	-1	2	-2

boundaries where linear interpolation is required. The benefit of the semi-orthogonal MIP-wavelets is that correct luminance values will be generated at any desired resolution level. Note that the filter coefficients of the lowpass filter $\{h_k\}$ match the averaging box filter above exactly. This can easily be verified by obtaining the tensor product of the scaling filter at any resolution level. For example, at one level of resolution $(j = 1)$ the effective sampling filter is a 2×2 filter with cells equal to $1/4$. At level $j = 0$, the filter is a 4×4 filter with cells equal to $1/16$. Note that under the DWT, the zeroth $(j = 2)$ resolution level (i.e., the original image) is not present in the pyramidal transformation. Because MIP-wavelets generate identical subsampled lowpass images to subimages generated by averaged box filters, in terms of resolution, both decompositions are identical. That is, in the case of monochromatic images, the same luminance resolution information is present in the scaled subimages of both approaches. In other words, the MIP-wavelets derived above serve as the basis functions for the multiresolution averaged box filter.

5.10.3 Variable Resolution Reconstruction with MIP-wavelets

Reconstruction with MIP-wavelets is lossless due to the orthogonality of the filters. To obtain interpolation results identical to MIP-mapping, an intuitive approach would be to maintain reconstructed scaled subimages produced by successive steps of the IDWT, then to perform the interpolation step as given by Equation (5.85). Although this approach would yield identical results due to the equivalence of subsampling filters and the DWT's perfect reconstruction (due to the orthogonality of the MIP-wavelets), it is memory-intensive. What is perhaps not obvious is that identical interpolation results can be obtained by scaling wavelet coefficients prior to reconstruction. Scaling of the wavelet coefficients prior to reconstruction results in the attenuation of the signal with respect to the average (low-pass) signal. Full decimation of the coefficients (scaling by 0) results in a lossy, subsampled reproduction of the original. Conversely, scaling wavelet coefficients by 1 preserves all detail information producing lossless reconstruction. Selectively scaling the coefficients by a value in the range $[0, 1]$ at appropriate levels of the wavelet pyramid produces a variable resolution image upon reconstruction. This approach is equivalent to MIP-mapping reconstruction with linear interpolation of pixel values.

In MIP-mapping, the value of the interpolant p is determined by some mapping function which specifies the desired resolution level l. The two closest pyramid resolution levels are then determined by rounding down and up to find subimage levels $j-1$ and j. The interpolant value is obtained by the relation:

$$p = l - \lfloor l \rfloor .$$

Note that the slope of the mapping function should match the resolution hierarchy of the pyramid, i.e., if resolution decreases eccentrically from some reference point, the parameter l should also decrease eccentrically. If it does not, its value may be reversed by subtracting from the number of resolution levels, i.e., $n-l$. To scale wavelet coefficients, p is set to either 0, 1, or the interpolant value at particular subbands according to the following relations dependent on l:

$$p = \begin{cases} 1, & j \leq \lfloor l \rfloor; \\ l - \lfloor l \rfloor, & j = \lceil l \rceil; \\ 0, & j > \lceil l \rceil. \end{cases} \tag{5.94}$$

For example, if at some particular pixel location (x,y), $l = 1.5$, then wavelet coefficients would be preserved (scaled by 1) at levels $j \leq 1$, scaled by .5 at level $j = 2$, and decimated (scaled by 0) at levels $j > 2$ at the appropriate pixel location in the subimages.

Theorem 1 *Wavelet coefficient scaling is equivalent to linear pixel interpolation under MIP mapping.*
Proof: Consider the interpolation step in the latter,

$$f = (1-p)\, f^{j-1} + (p)\, f^{j},$$

which is equivalent to Equation (5.85) with implicit pixel coordinates. When there is no need for interpolation, i.e., the desired resolution level at a pixel falls on a mapped resolution boundary, $l - \lfloor l \rfloor = 0$, and $j = \lceil l \rceil = l$, then

$$f = \begin{cases} f^{j-1} = f^{l-1} & \text{if } p = 0; \\ f^{j} = f^{l} & \text{if } p = 1; \end{cases}$$

or simply $f = f^{l}$ meaning that the resolution at the given pixel will match the resolution level of the subimage at the l^{th} pyramid level. Simplifying the IDWT reconstruction Equation (5.75) and expanding,

$$f^{j}_{\phi\phi} = (p f^{j-1}_{\psi} + \ldots + (p f^{2}_{\psi} + (p f^{1}_{\psi} + (p f^{0}_{\psi} + f^{0}_{\phi\phi}))) \ldots), \tag{5.95}$$

where f^{j}_{ψ} collectively represents subimages containing wavelet coefficient, $f^{j}_{\psi\phi}$, $f^{j}_{\phi\psi}$, $f^{j}_{\psi\psi}$, with implied pixel coordinates. Note that in Equation (5.95) the symbol p is now the interpolant and not the binary selection variable as used in Equation (5.75). In the case when $l - \lfloor l \rfloor = 0$,

$$f = (0 \ldots + (f^{l}_{\psi} \ldots + (f^{2}_{\psi} + (f^{1}_{\psi} + (f^{0}_{\psi} + f^{0}_{\phi\phi}))) \ldots) \ldots).$$

Since $f^{j+1}_{\phi\phi} = f^{j}_{\psi} + f^{j}_{\phi\phi}$ at each level of reconstruction, the resultant image will contain resolution matching the contents of subimage f^{l+1}. The subimage f^{l+1} contains the average (lowpass) component at level $l+1$, or equivalently, it contains the entire reproduced image at level l. In this case, the reconstructed image will contain resolution no finer than that found in subimage f^l, i.e., $f = f^l$, as with MIP-mapping.

When $l - \lfloor l \rfloor \neq 0$, i.e., interpolation is required due to the mapping falling between resolution boundaries, MIP mapping reproduces resolution at each pixel location according to Equation (5.85). In the IDWT, with $\lfloor l \rfloor < l < \lceil l \rceil$,

$$f = (0 \ldots + (0 + (p f_{\psi}^{\lceil l \rceil} + (f_{\psi}^{\lfloor l \rfloor} \ldots + (f_{\psi}^{2} + (f_{\psi}^{1} + (f_{\psi}^{0} + f_{\phi\phi}^{0}))) \ldots))) \ldots).$$

That is, resolution at the given pixel location is no finer than what is reproduced at subimage $f_{\phi\phi}^{\lceil l \rceil + 1}$, since

$$\begin{aligned} f_{\phi\phi}^{\lceil l \rceil + 1} &= p f_{\psi}^{\lceil l \rceil} + f_{\phi\phi}^{\lfloor l \rfloor + 1} \\ &= p f_{\psi}^{j} + f_{\phi\phi}^{j} \\ &= f_{\phi\phi}^{j+1} \\ &= f^{j}, \end{aligned} \tag{5.96}$$

where $\lfloor l \rfloor + 1 = \lceil l \rceil$, and $j = \lceil l \rceil$ from Equation (5.94). Equation (5.96) shows that the reconstructed resolution will be no finer than the resolution contained in f^l matching the finest resolution level produced by MIP-mapping as specified by Equation (5.85).

What remains to be shown is that the resolution gain obtained by scaling wavelet coefficients (at level j) is equivalent to the gain obtained by the MIP-mapping interpolation. That is, the scaling operation $p f_{\psi}^{j} + f_{\phi\phi}^{j}$ is equivalent (in terms of resolution gain) to the interpolation $(1-p) f^{j-1} + (p) f^{j}$. To prove this, recall the reconstruction relation,

$$\begin{aligned} f_{\phi\phi}^{j} &= f_{\psi}^{j-1} + f_{\phi\phi}^{j-1} \text{ or,} \\ f_{\phi\phi}^{j+1} &= f_{\psi}^{j} + f_{\phi\phi}^{j} \text{ giving,} \\ f_{\psi}^{j} &= f_{\phi\phi}^{j+1} - f_{\phi\phi}^{j}. \end{aligned} \tag{5.97}$$

Substituting Equation (5.97) into the wavelet scaling equation,

$$\begin{aligned} p f_{\psi}^{j} + f_{\phi\phi}^{j} &= f_{\phi\phi}^{j} + p f_{\psi}^{j} \\ &= f_{\phi\phi}^{j} + p (f_{\phi\phi}^{j+1} - f_{\phi\phi}^{j}) \\ &= f_{\phi\phi}^{j} - p f_{\phi\phi}^{j} + p f_{\phi\phi}^{j+1} \\ &= (1-p) f_{\phi\phi}^{j} + p f_{\phi\phi}^{j+1} \\ &= (1-p) f_{\phi\phi}^{j-1} + p f_{\phi\phi}^{j} \\ &= (1-p) f^{j-1} + p f^{j} \end{aligned}$$

completes the proof showing that scaling wavelet coefficients by p at pyramid level j is equivalent to linearly interpolating pixel values of scaled subimages f^{j-1}, f^{j} through MIP-mapping. □