

Wavelets, Subband Coding, and Best Bases

KANNAN RAMCHANDRAN, MARTIN VETTERLI, FELLOW, IEEE,
AND CORMAC HERLEY, MEMBER, IEEE

Invited Paper

The emergence of wavelets has led to a convergence of linear expansion methods used in signal processing and applied mathematics. In particular, subband coding methods and their associated filters are closely related to wavelet constructions. We first review such constructions with a signal processing perspective. We then discuss the idea behind signal adapted bases and associated algorithms before showing how wavelets and subband coding methods are used in signal compression applications.

I. INTRODUCTION

A. Linear Expansions in Signal Processing

Signal expansions, that is methods of writing a signal as a linear combination of elementary "atoms" or building blocks, are central to signal processing theory and applications. That is, we write a signal f belonging to a space \mathcal{S} as

$$f = \sum_i \alpha_i \psi_i \quad (1)$$

where the set $\{\psi_i\}_{i \in \mathcal{Z}}$ is complete in \mathcal{S} . We will concentrate on the case where the set is actually a basis, that is, the vectors ψ_i are linearly independent.¹

For example, discrete-time signal processing is based on representing band-limited signals (signals having a Fourier transform supported between $-\pi/T$ and π/T) by a linear combination of *sinc* functions. In terms of (1), $\psi_i(t) =$

Manuscript received June 1, 1995; revised December 15, 1995. K. Ramchandran's work was supported in part by the National Science Foundation grant MIP-9409587 (RIA-94). M. Vetterli's work was supported in part by the US National Science Foundation under grant MIP-93-21302 and the Swiss National Foundation under grant 2100-043136.95.

K. Ramchandran is with the Department of Electrical Engineering, University of Illinois, Urbana, IL 61801 USA.

M. Vetterli is with the Department of Electrical Engineering, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland, and Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720 USA.

C. Herley is with the Hewlett-Packard Laboratories, Palo Alto, CA 94304-1126 USA.

Publisher Item Identifier S 0018-9219(96)02572-8.

¹The case where the set is overcomplete leads to frames [1]. Frames are often obtained as sampled versions of continuous transforms like the continuous wavelet transform (CWT) or the continuous short-time Fourier transform (STFT), and they are discussed elsewhere in this issue [2].

$\sin(\pi(t/T-i))/\pi(t/T-i)$ and $\alpha_i = f(iT)$. The frequency $2\pi/T$ (that is, twice the maximum frequency present in $f(t)$) is called the Nyquist sampling frequency.

Another standard example is the Fourier series expansion of periodic signals with period T in terms of sine and cosines of frequencies $2\pi i/T, i \in \mathcal{Z}$.

In what follows, we will be mostly concerned with signals that belong to $L_2(\mathcal{R})$ (the space of square integrable functions) or $l_2(\mathcal{Z})$ (the space of square summable sequences). In that case, orthonormal or biorthogonal bases play a key role. In the former case, there exists a complete set $\{\psi_i\}_{i \in \mathcal{Z}}$ such that

$$\langle \psi_i, \psi_j \rangle = \delta_{i,j} \quad (2)$$

where $\langle \cdot, \cdot \rangle$ is the usual inner product in $L_2(\mathcal{R})$ or $l_2(\mathcal{Z})$ and $\delta_{i,j} = 1$ if $i = j, 0$ else. Equation (1) becomes

$$f = \sum_i \langle \psi_i, f \rangle \psi_i \quad (3)$$

where the convergence is in the square norm sense. In the latter, biorthogonal case, there exists a dual set $\{\tilde{\psi}_i\}_{i \in \mathcal{Z}}$ such that

$$\langle \tilde{\psi}_i, \psi_j \rangle = \delta_{i,j} \quad (4)$$

and (1) becomes

$$f = \sum_i \langle \tilde{\psi}_i, f \rangle \psi_i = \sum_i \langle \psi_i, f \rangle \tilde{\psi}_i. \quad (5)$$

Usually, both for analytical and computational reasons, only structured bases are of interest. Historically, the Fourier, local Fourier and the Haar basis are typical examples [3]. That is, all the basis functions ψ_i are obtained through a small set of transformations from a prototype function ψ . These transformations typically include the shift, the modulation, and the scaling operation. In this case, the atoms used to write the function f are closely related to each other.

B. Time-Frequency Localization

A key concept in signal analysis is the notion of "localization" in time and frequency. Given a suitable measure of spread in time and frequency (e.g., second moment in time and frequency of the function normalized to unit energy [4]), one can define a region in the time/frequency plane where the function is concentrated. Each atom in an expansion has a particular localization, and in a structured expansion, these localizations are related to each other through simple transformations. The result is a tiling of the time-frequency (TF) plane which depends on the set of transformations. Typically, the short-time Fourier transform leads to a rectangular tiling, while the wavelet transform leads to a dyadic tiling.² If a signal is expanded using a particular set of atoms, one gets a good picture of the TF localization of the signal by considering the tiles corresponding to the largest coefficients (in absolute value). Obviously, certain tilings will be better suited for capturing the TF content of certain signals, which leads to the concept of "best bases" discussed in Section IV.

C. Multiresolution Signal Representations

An intuitive concept is that of multiresolution, or the idea that one can consider a signal at different levels of resolution. This notion is particularly evident in image processing and computer vision, where coarse versions of images are often used as a first approximation in computational algorithms. In signal processing parlance, a lowpass and subsampled version of signal is often a good coarse approximation for many real life signals.

It turns out that this intuitive paradigm is also the mathematical framework for wavelet constructions, as formalized by Mallat and Meyer [5], [6]. The wavelet decomposition is a successive approximation method which adds more and more projections onto "detail" spaces, or spaces spanned by wavelets and their shifts at different scales.

In addition, this multiresolution approximation is well suited to many applications. For the sake of example, we will briefly discuss an obvious case where successive approximation is useful, namely in browsing through image databases as done for instance on the worldwide web. Rather than downloading each full image, which would be time consuming, one only downloads a coarse version, which can be done speedily. Then, one can fetch the rest, or additional detail, if the image seems of interest. This indicates that the coarse version is probably more useful than the detail. In communication applications, this leads to transmission methods where the coarse version of a signal is better protected against transmission errors than the detail information [7].

D. Best Bases

Since one of the main applications of wavelets and subband coding is compression, it is important to discuss the difference between such methods and the more standard transform coding methods like the Karhunen-Loève trans-

²For a picture of these tilings, see the article by Cohen and Kovačević in this issue.

form (KLT) and its close cousin and approximation, the discrete cosine transform (DCT). It should be noted that the optimality of the KLT for compression can only be shown in some limited cases (stationarity, fine quantization, l_2 error measure [8]). For example, in low bit rate cases, and using perceptual criteria, it is unclear what method would be best. More generally, in the nonstationary case, there is no known superiority of the DCT over wavelet/subband methods.

An interesting case is when an adapted basis is chosen, that is, the transform depends on the signal. This is the case for best bases, which try to find a tiling of the TF plane that gathers most of the energy in the fewest possible number of coefficients. An important feature is that no model of the signal is assumed *a priori*.

E. Overview of the Paper

We start by reviewing the construction of structured bases for $l_2(\mathcal{Z})$ using multirate filter banks in Section II. The classic constructions of filter banks are presented. While this is standard signal processing material, it comes with a twist, since these constructions will be used to derive wavelet bases for $L_2(\mathcal{R})$ through Daubechies' iteration algorithm in Section III. This section presents the basic construction of wavelet bases from filter banks, as well as variations on that theme. Section IV addresses the construction of best bases, especially with an eye on compression. The idea of picking a basis that is suited to a particular signal is appealing, but requires efficient algorithms that are also presented. The impact of wavelet and subband methods on signal compression is covered in Section V.

II. STRUCTURED BASES FOR $l_2(\mathcal{Z})$ AND MULTIRATE FILTER BANKS

A. Filter Banks

Assume a sequence of real or complex samples³ $x[n]_{n \in \mathcal{Z}}$ in $l_2(\mathcal{Z})$, the space of finite energy sequences, $\sum_{n \in \mathcal{Z}} |x[n]|^2 < \infty$. An orthonormal basis is a set of sequences $\{\psi_i[n]\}_{i \in \mathcal{Z}}$ such that $\langle \psi_i[n], \psi_j[n] \rangle = \sum_{n \in \mathcal{Z}} \psi_i^*[n] \cdot \psi_j[n] = \delta_{i,j}$ where $*$ stands for complex conjugation and the set is complete in $l_2(\mathcal{Z})$.

Structured bases are obtained when a finite set $g_i[n], i = 0 \dots N-1$ is used to generate the infinite basis, for example through shifts by N

$$\psi_{i+kN}[n] = g_i[n - kN], \quad k \in \mathcal{Z}, \quad i = 0, 1, \dots, N-1. \quad (6)$$

Expansion in such a basis can be computed using multirate filter banks. Start by convolving the signal with filters having impulse responses $h_i[n] = g_i[-n]$ ⁴ followed by subsampling by N . This first operation, commonly called an analysis filter bank [10], [3], computes the expansion coefficients. A synthesis filter bank, consisting of upsampling by N followed by interpolation filters with impulse

³We adhere to the signal processing convention that uses square brackets for discrete-time sequences [9].

⁴The change of sign in the index is needed because convolution involves time-reversal.

responses $g_i[n], i = 0 \dots N - 1$ calculates the linear combination of the basis vectors, that is, the reconstructed signal. The overall scheme is shown in Fig. 1 for a two-channel case. Note that the output of the i th interpolation filter is the orthogonal projection of the input signal onto the space spanned by $\{g_i[n - kN]\}_{k \in \mathcal{Z}}$. Therefore, the analysis/synthesis system computes projections onto N orthonormal subspaces $\mathcal{S}_0, \mathcal{S}_1, \dots, \mathcal{S}_{N-1}$, followed by a summation. Because the filter impulse responses $g_i[n]$ and their translates by integer multiples of N form an orthonormal basis, we have

$$\mathcal{S}_0 \oplus \mathcal{S}_1 \oplus \dots \oplus \mathcal{S}_{N-1} = l_2(\mathcal{Z}). \quad (7)$$

where \oplus denotes the direct sum.

B. Design of Filter Banks

How do we design such an orthonormal filter bank? We will concentrate on the two channel case with real impulse responses, because it is the simplest and most important case in practice. The problem consists in finding a sequence $g_0[n]$ which is orthonormal to its shifts by two. The sequence $g_1[n]$ can then be easily found. Taking a z -transform⁵ of the sequence $g_0[n]$ or $G_0(z) = \sum_{n \in \mathcal{Z}} g_0[n]z^{-n}$, we can write the orthonormality condition in z -domain as

$$\langle g_0[n], g_0[n - 2k] \rangle = \delta_k \Leftrightarrow \frac{1}{2}(P(z) + P(-z)) = 1 \quad (8)$$

where $P(z) = G_0(z)G_0(z^{-1})$ is the deterministic autocorrelation of the sequence $g_0[n]$. This relation is shown by noting that the even terms of the autocorrelation are zero except the central one, and expressing this in z -domain using the subsampling formula [3]. On the unit circle (8) becomes

$$|G_0(e^{j\omega})|^2 + |G_0(e^{j(\omega+\pi)})|^2 = 2 \quad (9)$$

that is, the filter is power complementary [10]. Such filters are also called quadrature mirror filters (QMF's).⁶

Given a sequence $g_0[n]$ satisfying (8), one can show that a sequence $g_1[n]$ with z -transform

$$G_1(z) = z^{-1}A(z)G_0(-z^{-1}), \quad \text{where} \\ A(z)A^*(z^{-1}) = 1 \quad \text{and} \quad A(z) = A(-z) \quad (10)$$

is orthonormal with respect to even shifts, as well as orthogonal to $g_0[n - 2k], k \in \mathcal{Z}$. Together, $\{g_0[n - 2k], g_1[n - 2l]\}_{k, l \in \mathcal{Z}}$ form an orthonormal basis for $l_2(\mathcal{Z})$.

A case of great importance is when the sequence $g_0[n]$ is of finite length L , that is, the filter is FIR. In that case, the filter $g_1[n]$ is trivially related to $g_0[n]$ since $A(z)$ in (10) is simply $z^{2l}, l \in \mathcal{Z}$, or an even shift. Then, by taking

⁵We assume that the sequences we consider have sufficient decay so that convergence on the unit circle $z = e^{j\omega}$ is insured.

⁶Actually, the original quadrature mirror filters [11] were linear phase and nonorthogonal but satisfied a similar quadrature formula.

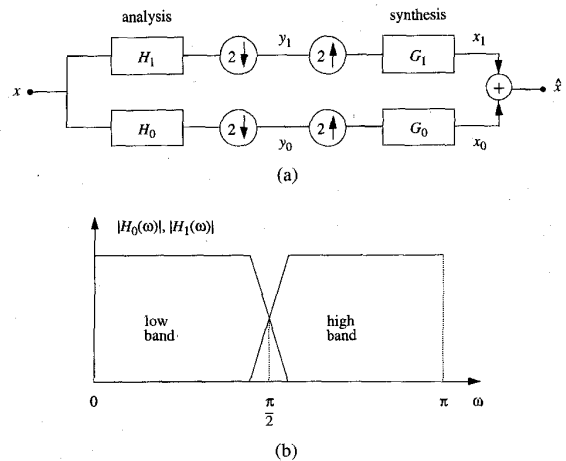


Fig. 1. Two channel filter bank. (a) The two channels, with filtering using $h_i[n]$, subsampling, upsampling and interpolation filtering with $g_i[n]$. (b) Spectrum splitting between low and highpass filters.

the inverse z -transform of (10), $g_1[n] = (-1)^{n-1}g_0[-n - 2l + 1]$.⁷

For the filter impulse response to be orthogonal to its even translates, L must be even, or $L = 2M$. In order to satisfy (8), $P(z)$ must be a symmetric polynomial of the form

$$P(z) = 1 + \sum_{n=0}^{M-1} p_{2n+1}(z^{2n+1} + z^{-(2n+1)}) \quad (11)$$

and $P(z)$ must satisfy $P(e^{j\omega}) \geq 0$ since it is an autocorrelation function. Thus one way to design an orthonormal FIR filter bank is to find an autocorrelation sequence satisfying (8), and then to take its spectral factor. This last step involves a choice in the phase of the factor (choice of zeros inside or outside the unit circle) and a possible delay factor.

The above method has been used in the first designs of orthonormal filter banks by Smith and Barnwell [12], [13] and Mintzer [14] using traditional signal processing criteria for the filter design (e.g., good out-of-band signal rejection). Daubechies [15] used a different criterion leading to the design of maximally flat filters that can be used to generate wavelet bases (see Section III). In this design, the length $L = 2N$ filter is forced to have N zeros at $z = -1$, that is, $P(z)$ has the form

$$P(z) = \left(\frac{1+z}{2}\right)^N \left(\frac{1+z^{-1}}{2}\right)^N R(z) \quad (12)$$

where $R(z) = r_0 + \sum_{n=1}^{N-1} r_n(z^n + z^{-n})$ is chosen so that the resulting $P(z)$ satisfies (8). The case of the minimal degree $R(z)$ (having powers going from $-N+1$ to $N-1$) leads to the shortest filters with N zeros at $\omega = \pi$.

Given that the lowpass filter has N zeros at $\omega = \pi$, the highpass filter $G_1(z)$ will then have N zeros at

⁷This can lead to noncausal filters. However, if $g_0[n]$ is FIR, a finite even shift of $g_1[n]$ will make it causal. The same holds true for the analysis filters (which are time-reversed versions of the synthesis filters) and thus in the FIR case, causality is not a problem if one can tolerate a finite delay between input and output of a filter bank.

$\omega = 0$ [see (10)]. Thus from the moment property of the Fourier transform, it has N zero moments, and the polynomial sequences $\{n^k\}, n \in \mathcal{Z}, k = 0, 1, \dots, N-1$ are zeroed out by this filter. This zero moment property has a certain importance in compression applications, since smooth signals can be modeled approximately as a linear combination of low order polynomial sequences, and will thus lead to zero or at least small highpass components. Note that the lowpass filter, by complementarity [see (9)], is maximally flat at $\omega = 0$.

Fig. 2 shows the comparison between a signal processing based design [13] and a wavelet oriented design [15] for an FIR orthogonal filter. Note the good out of band rejection in case (a) and the flatness and monotonicity in case (b).

One more design technique is based on the fact that one can associate a unitary operator with an orthogonal filter bank [10], [3]. Write the filters $G_0(z)$ and $G_1(z)$ in terms of their even and odd components [called polyphase components $G_{ij}(z)$]

$$G_i(z) = G_{i0}(z^2) + z^{-1}G_{i1}(z^2), \quad i = 0, 1. \quad (13)$$

It can be shown that the associated matrix of polyphase components

$$G(z) = \begin{pmatrix} G_{00}(z) & G_{01}(z) \\ G_{10}(z) & G_{11}(z) \end{pmatrix} \quad (14)$$

is unitary on the unit circle, and more generally, paraunitary or

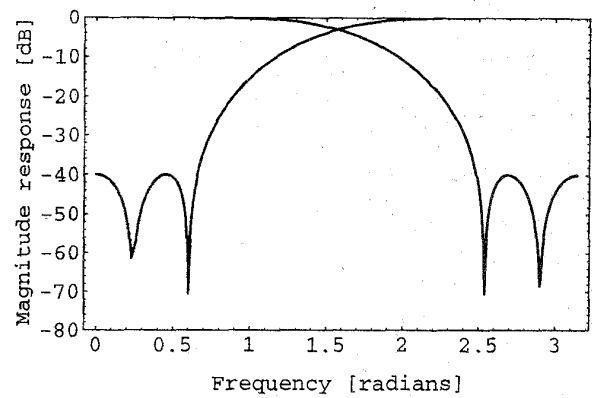
$$G(z)\tilde{G}(z) = \tilde{G}(z)G(z) = I \quad (15)$$

where the $\tilde{\cdot}$ stands for transposition and replacing z by z^{-1} (as well as conjugation of the coefficients if they are complex). The interest in such a polyphase representation is that paraunitary matrices possess complete factorizations in terms of elementary matrices of degree zero (rotations) and one (diagonal matrix with delays) [16], [17]. Thus one can optimize such a factorized structure to find a filter bank meeting specific constraints. Note that the constraint of a large number of zeros at $\omega = \pi$ (as required for wavelet designs) is difficult to enforce in this form.

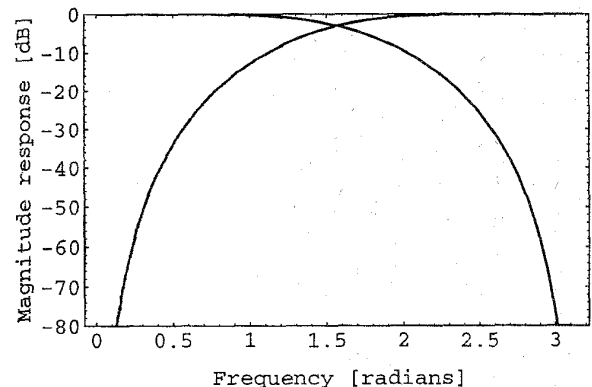
C. Variations on the Filter Bank Theme

So far, we have discussed the two-channel orthogonal filter bank case. An obvious extension is obtained by cascading elementary two channel filter banks in a binary tree structure. As to be expected, any such tree is again an orthogonal filter bank. Such general tree structured filter banks are often called wavelet packets [18] because the resulting basis functions often resemble a localized train of waves or wavelets.

The two most important representatives are the full tree and the wavelet tree. In the full tree case, each band is further split into two bands, and so on, leading to a balanced tree. In the wavelet tree case, only the lowpass channel is further split, leading to an unbalanced tree. Assume for this discussion that each two channel bank performs an ideal split into a half-band lowpass and highpass channel, respectively. Then, the full tree leads to an evenly spaced



(a)



(b)

Fig. 2. Comparison of a traditional signal processing design and a wavelet based design of an orthogonal lowpass filter. (a) Length-eight Smith and Barnwell orthogonal filter and (b) length-eight Daubechies orthogonal filter.

frequency analysis (each channel keeps $1/2^k$ th of the spectrum, where k is the tree depth) and we obtain an approximation to a short-time Fourier analysis. In the TF plane, a rectangular tiling is obtained. In the wavelet tree case, only the lowpass part of the spectrum is further analyzed, leading to an octave-band or constant relative bandwidth analysis typical for wavelet schemes. Therefore, such a filter bank is often called a discrete-time wavelet transform. The tiling of the TF plane is a dyadic tiling, with fine frequency resolution at low frequencies, and good time resolution at high frequencies.

If one is interested in the actual basis functions of such tree structures, one can use some simple multirate identities and derive the equivalent filter impulse responses. In particular, note that filtering with $F(z)$ followed by upsampling by two can be written as upsampling by two followed by convolution with $F(z^2)$. Using this relation, one can verify that a cascade of k stages of upsampling by two and filtering by $G_0(z)$ is equivalent to upsampling by 2^k followed by a convolution with

$$G_0^{\{k\}}(z) = G_0(z)G_0(z^2)G_0(z^4) \cdots G_0(z^{2^{k-2}})G_0(z^{2^{k-1}}). \quad (16)$$

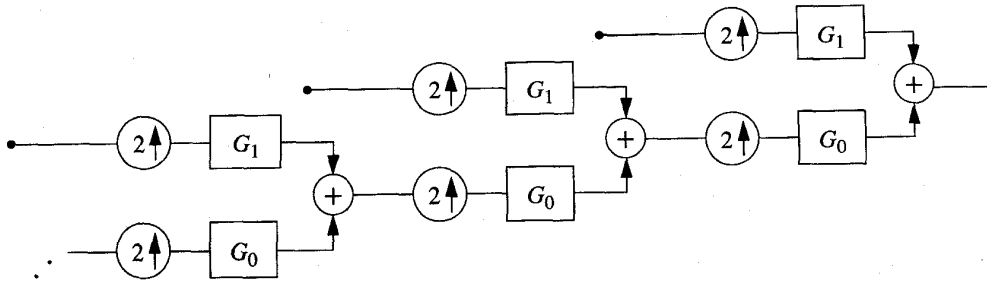


Fig. 3. Iterated two channel synthesis filter bank. The two equivalent impulse responses of the lowest channel (iterated lowpass filters) and the first bandpass channel (one highpass followed by iterated lowpass filters) will lead to the scaling function and the wavelet, respectively, as shown in Section III.

This filter corresponds to the lowpass impulse response in a wavelet tree. The next impulse response, corresponding to the first bandpass channel, is found similarly to have a z -transform given by

$$G_1^{(k)}(z) = G_0(z)G_0(z^2)G_0(z^4) \cdots G_0(z^{2^{k-2}})G_1(z^{2^{k-1}}). \quad (17)$$

These two impulse responses will play a critical role in the construction of scaling functions and wavelets in the next section. The equivalent synthesis filter bank is shown in Fig. 3.

Instead of tree structured filter banks, one can design multichannel filter banks directly. While the principles are similar to the two channel case, the analysis and design are somewhat more involved [10]. It is to be noted that block transforms like the KLT or the DCT are particular cases of N channel filter banks. Simply, the filters have a length restricted to the block size N , which corresponds to the subsampling factor.

Another possibility is to relax orthogonality, and consider biorthogonal FIR filter banks [19], [20]. In that case, the analysis and synthesis filters satisfy a relation similar to (8), namely

$$H_0(z)G_0(z) + H_0(-z)G_0(-z) = 2 \quad (18)$$

where now $H_0(z)$ is not necessarily a time reversed version of $G_0(z)$. An attraction of biorthogonal solutions is that linear phase (or symmetry/antisymmetry of the impulse response) FIR filters are now possible, whereas it is excluded in the orthogonal case (except for the trivial Haar filters, where $G_0(z) = 1/\sqrt{2}(1+z^{-1})$). Thus biorthogonal filters have become quite popular in image coding [21], [22]. The lack of orthogonality complicates quantization and bit allocation between the bands (there is no conservation of energy anymore) [23]. However, smooth filters can be obtained: in particular, one can find synthesis filters which lead to smooth wavelets. This in turn leads to smooth approximations which are often more pleasant in image compression.

If finite impulse response is not required, then infinite impulse response (IIR) filters can be designed. In the orthogonal IIR case, because of time-reversal, causality

will be violated in either analysis or synthesis (or both). However, solutions like orthogonal linear phase filter banks are now possible [24], [25]. The lack of causality is not problematic in image processing, where the signal is finite in size, and convolution can be applied in either direction [26].

III. BASES FOR $L_2(\mathcal{R})$ FROM ITERATED DISCRETE-TIME BASES

A. Fourier-Like Bases

We are now concerned with functions or signals in $L_2(\mathcal{R})$, that is having finite squared norm $\int |x(t)|^2 dt < \infty$. Again, structured bases are of greatest interest. Historically, Fourier series are an important class of structured bases, and they are useful for periodic signals or signals of compact support. An obvious extension, consisting in slicing an infinitely supported nonperiodic signal into nonoverlapping adjacent pieces and writing a Fourier series for each piece, can be done.⁸ The TF analysis achieved using such a piecewise Fourier series (assuming pieces of equal size T) corresponds to a rectangular tiling or a short-time Fourier transform with a square window. However, the frequency resolution is not very good due to the order $(1/\omega)$ decay of the Fourier transform of the window. Unfortunately, there are no better Fourier-like local bases, as demonstrated by the Balian–Low theorem [15]. An important feature is that, given the square window, all the basis functions are obtained through shifts (by multiples of T) and modulation (by complex sinusoids of frequency $2\pi k/T$). An excellent review of such Fourier-like bases, also called Weyl–Heisenberg wavelets, can be found in [27].

B. Wavelet Bases

A very different family of orthonormal bases is obtained when choosing a family based on a single prototype but using now shifting and scaling (instead of modulation)

$$\psi_{mn}(t) = 2^{-m/2} \psi\left(\frac{t-2^n}{2^m}\right) \quad (19)$$

⁸There are convergence problems at the boundaries due to the implicit periodization of each piece.

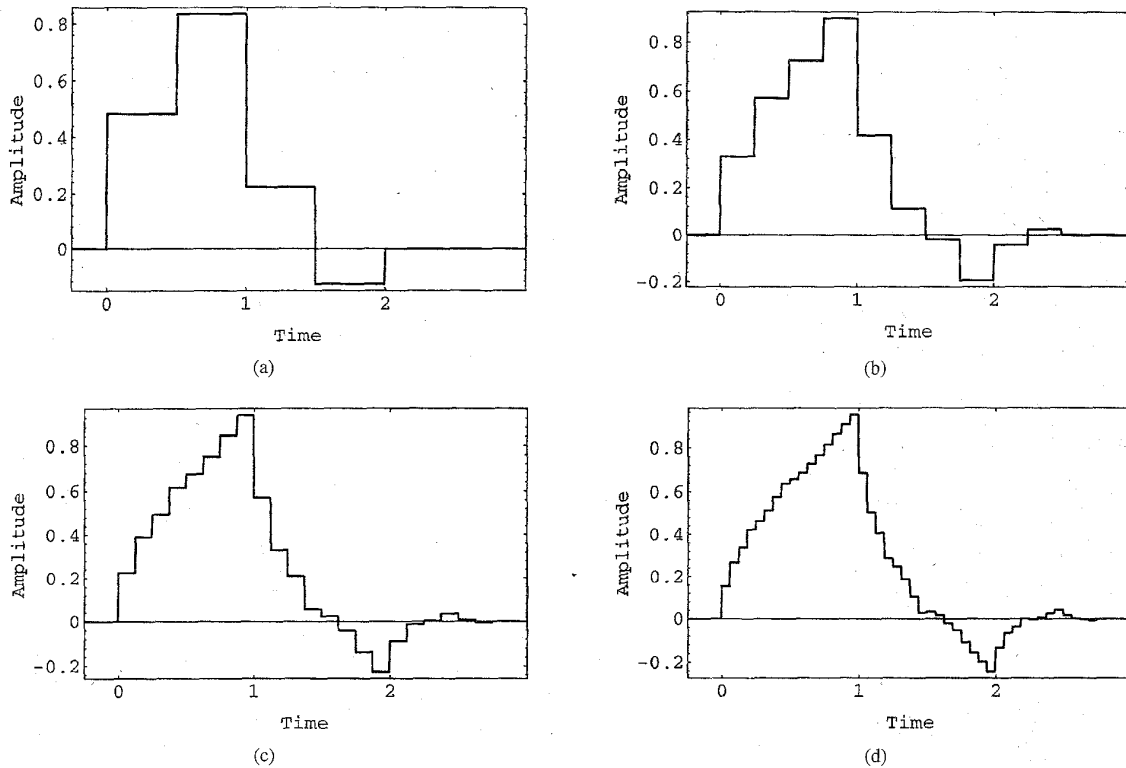


Fig. 4. First four iterations of the length-four Daubechies orthogonal filter. (a) $k = 1$, (b) $k = 2$, (c) $k = 3$, and (d) $k = 4$.

where the prototype function $\psi(t)$, called a wavelet, satisfies a number of conditions so that the family is an orthonormal basis for $L_2(\mathcal{R})$. The frequency domain analysis has constant relative frequency resolution and inverse time resolution, which leads to a dyadic TF tiling. Historically, the two most important wavelet examples are

- The Haar wavelet, with $\psi(t) = 1, 0 \leq t < 1/2, \psi(t) = -1, 0 \leq t < 1/2, 0$ else.
- The sinc wavelet, with Fourier transform $\Psi(\omega) = 1, \pi < |\omega| \leq 2\pi, 0$ else.

The first has very good time resolution and the second very good frequency resolution. However, their resolution in the other domain is poor. The Haar construction was considered somewhat of a curiosity, until the general wavelet constructions of the 1980's were found. These constructions can be roughly divided into two classes. First, there are the direct constructions by Battle [28], [29], Lemarié [30], Meyer [6], and Stromberg [31] which consist in "smoothing" the sinc wavelet in the Fourier domain so that the time domain function has faster decay. Then, there are indirect constructions pioneered by Daubechies [15] and based on discrete-time filters banks. These can be seen as smoothed versions of the Haar wavelet, leading to better frequency resolution. That is, all new wavelet constructions can be seen as exploring the range between the two extreme cases of the Haar and sinc wavelet.

Given our interest in signal processing, the Daubechies construction is more relevant, since it leads to imple-

mentable exact algorithms with finite complexity.⁹ Let us review this construction very briefly. We restrict ourselves to orthogonal filter banks with finite length filters $g_0[n]$ and $g_1[n]$ of length L ($g_i[n] = 0, n \notin [0 \dots L-1]$). Consider the k -times iteration of such a filter bank as in a discrete-time wavelet transform leading to equivalent filters $g_0^{\{k\}}[n]$ and $g_1^{\{k\}}[n]$ [see (16) and (17)]. We can associate two piecewise constant approximations to these impulse responses

$$\phi^{\{k\}}(t) = g_0^{\{k\}}[n], \quad \frac{n}{2^k} \leq t < \frac{(n+1)}{2^k}, \quad k = 1, 2, \dots \quad (20)$$

$$\psi^{\{k\}}(t) = g_1^{\{k\}}[n], \quad \frac{n}{2^k} \leq t < \frac{(n+1)}{2^k}, \quad k = 1, 2, \dots \quad (21)$$

Fig. 4 shows the iteration process for a length-four Daubechies filter. As the iteration progresses, the piecewise constant approximation resembles more and more a continuous function, and indeed, the limit is continuous [15]. Fig. 5(a) shows a family of scaling functions obtained from length-four orthogonal filters.

From properties of the underlying filter banks, it can be verified that the functions $\phi^{\{k\}}(t)$ and $\psi^{\{k\}}(t)$ are orthogonal to their integer shifts, as well as to each other (also with integer shifts). Since the $g_i[n]$'s are of finite

⁹The direct constructions lead either to algorithms with unbounded complexity, or finite complexity approximations.

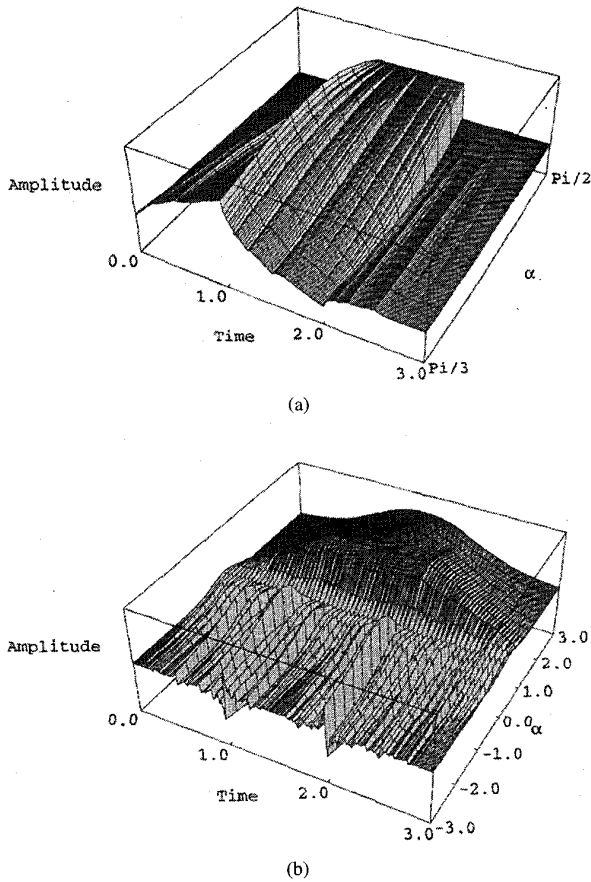


Fig. 5. Examples of scaling function families based on four tap filters. (a) Orthogonal family based on lowpass filters with one zero at $\omega = \pi$. At one end, one obtains the Daubechies scaling function, while at the other, the Haar solution is obtained (the box function). (b) Biorthogonal family based on symmetric length four filters with impulse response $[1, \alpha, \alpha, 1]$ and α going from -3 to 3 . At one end ($\alpha = 3$) is a quadratic spline, while at the other ($\alpha = -3$) is highly irregular. For $\alpha = 0$ there is actually no convergence of the iteration scheme.

length, the length of $\phi^{[k]}(t)$ and $\psi^{[k]}(t)$ remains bounded, namely they are supported on the interval $[0, L - 1]$.

Of great interest is the limit of $\phi^{[k]}(t)$ and $\psi^{[k]}(t)$ when $k \rightarrow \infty$. We will not discuss convergence issues here [15], [1] but simply mention that under relatively mild conditions, we can consider infinite products in the Fourier domain which correspond to the Fourier transforms of $\phi(t)$ and $\psi(t)$.

Because the $g_i[n]$'s are of finite length, $G_i(e^{j\omega})$ are trigonometric polynomials. A necessary condition for convergence of (20) is that $G_0(e^{j\omega})|_{\omega=\pi} = 0$ which implies, because of (9), that $|G_0(0)| = \sqrt{2}$. It will be convenient to work with a renormalized version of $G_0(e^{j\omega})$

$$M_0(\omega) = \frac{1}{\sqrt{2}} G_0(e^{j\omega}) = \left(\frac{1 + e^{j\omega}}{2} \right)^N M'_0(\omega) \quad (22)$$

where N is at least one because of the necessary condition of at least one zero at $\omega = \pi$ in $G_0(e^{j\omega})$. Note

that both $M_0(0)$ and $M'_0(0)$ equal one because of the renormalization.

Using the function $M_0(\omega)$, it can be verified that as $k \rightarrow \infty$, the Fourier transform of (20) becomes

$$\Phi(\omega) = \prod_{i=1}^{\infty} M_0\left(\frac{\omega}{2^i}\right) \quad (23)$$

which is called the *scaling function* in a wavelet scheme. The span of $\{\phi(t-n)_{n \in \mathcal{Z}}\}$ defines a space called V_0 . Equation (21) leads to, using the notation $M_1(\omega) = G_1(\omega)/\sqrt{2}$

$$\Psi(\omega) = M_1(\omega/2) \prod_{i=2}^{\infty} M_0\left(\frac{\omega}{2^i}\right) \quad (24)$$

which is the *wavelet*. The space spanned by $\{\psi(t-n)_{n \in \mathcal{Z}}\}$ is called W_0 .

The particular form of these products leads to interesting properties of the associated functions. For example, the scaling function satisfies a two scale equation property, since

$$\Phi(\omega) = M_0(\omega/2) \prod_{i=2}^{\infty} M_0\left(\frac{\omega}{2^i}\right) = M_0(\omega/2) \Phi(\omega/2) \quad (25)$$

which in time domain equals

$$\phi(t) = \sqrt{2} \sum_{k=0}^{L-1} g_0[k] \phi(2t - k). \quad (26)$$

In terms of spaces, if we call V_{-1} the span of $\{\phi(2t - n)_{n \in \mathcal{Z}}\}$ then $V_0 \subset V_{-1}$ because of the two scale equation (25). Further, it can be verified that $V_{-1} = V_0 \oplus W_0$. Calling W_i the space spanned by $\{1/2^{i/2} \psi(2^i t - n)_{n \in \mathcal{Z}}\}$, it turns out that the direct sum of $W_i, n \in \mathcal{Z}$ is $L_2(\mathcal{R})$.¹⁰ That is $\psi_{mn}(t)$ as in (19) generates an orthonormal basis for $L_2(\mathcal{R})$.

A study of the infinite product in (23) shows that zeros of $M_0(\omega)$ at $\omega = \pi$ play a key role. For example, for V_0 to contain polynomials up to degree M requires $M_0(\omega)$ to have $M + 1$ zeros at $\omega = \pi$, which is known as the Strang-Fix condition. Conversely, more zeros at $\omega = \pi$ lead to smoother scaling functions. A key result by Daubechies [15] was to show a sufficient condition on the discrete-time sequence $g_0[n]$ such that $\phi(t)$ is regular, that is, it is continuous, differentiable, etc. This regularity is obtained by placing a maximum number of zeros at $\omega = \pi$ while maintaining the orthogonality condition $\langle g_0[n], g_0[n - 2k] \rangle = \delta_k$. This leads to a family of filters first described in [15] and thus called Daubechies' filters. We show the first few scaling functions in Fig. 6.

The result is striking, because a very simple condition imposed on the discrete-time sequence $g_0[n]$ leads to strong properties of the associated scaling function $\phi(t)$.

Is regularity important in practice, especially considering the fact that real implementations of wavelet expansions

¹⁰The exact conditions under which this holds true is the basis of multiresolution analysis [5], [6].

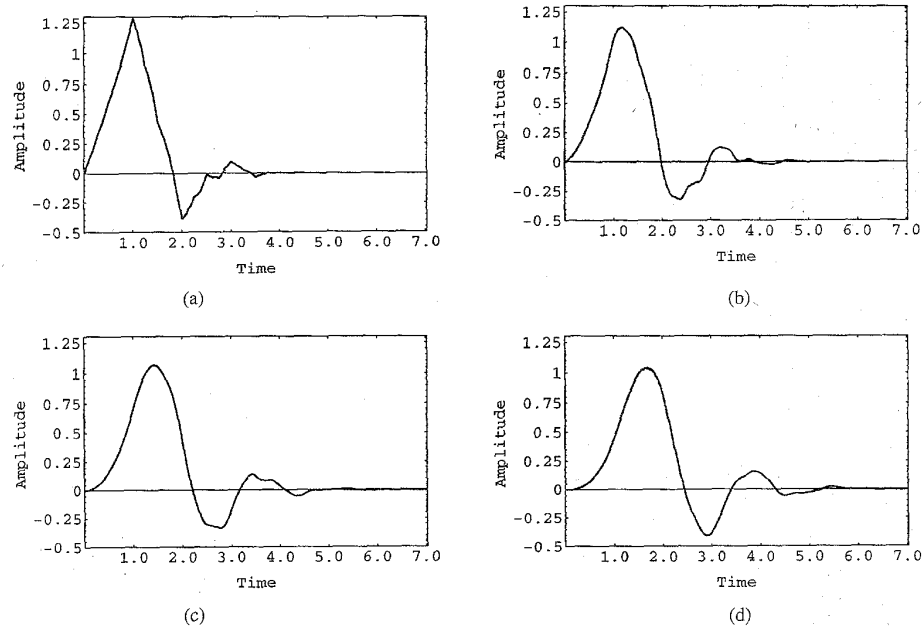


Fig. 6. Four scaling functions obtained from Daubechies filters. (a) Length-five scaling function obtained from a six tap filter. (b) Length-seven scaling function obtained from a eight tap filter. (c) Length-nine scaling function obtained from a 10-tap filter (only part from zero to seven shown). (d) Length-11 scaling function obtained from a 12-tap filter (only part from zero to seven shown).

are often done in discrete-time using a finite iteration of a filter bank? It is true that if no approximation is used, such a scheme is simply a discrete-time orthonormal basis, and regularity is not necessary. However, in most cases, one is looking for a good approximation even if not all coefficients of the expansion are retained, or only coarse approximations of the coefficients are used. This is the case in compression applications. Then, regularity becomes an important feature. It is easiest to consider the case when higher frequency (or fine scale) coefficients are dropped. This corresponds to a truncated wavelet series, and thus the signal is approximated by linear combinations of the low frequency or coarse scale basis functions. If these are smooth, the approximation is smooth, while if the filter is not regular, the approximation is likely to be very jittery. This intuitive argument can be verified in practice, for example in real compression systems, where a certain degree of regularity of an iterated filter bank is necessary [32].

C. Variations on Wavelet Constructions from Filter Bank

As to be expected, the Daubechies' construction can be used with other filter banks as well, like biorthogonal, multichannel, and multidimensional ones.

Take a biorthogonal filter bank with filters $\{h_0, h_1, g_0, g_1\}$, where h_0 and g_0 satisfy (18). We now consider the iteration of both the analysis and the synthesis filter banks (which are now not related by time reversal anymore). Both filters need at least one zero at $\omega = \pi$ or $z = -1$ for convergence of the iteration scheme, and thus $H_0(1)G_0(1) = 2$. Assuming that $H_0(1) = G_0(1) = \sqrt{2}$ (otherwise, rescale appropriately), we can define $M_0(\omega)$ as

in (22) and introduce

$$\tilde{M}_0(\omega) = \frac{1}{\sqrt{2}} H_0(e^{j\omega}) = \left(\frac{1 + e^{j\omega}}{2} \right)^N \tilde{M}_0'(\omega). \quad (27)$$

Then $\Phi(\omega)$ and $\Psi(\omega)$ are as in (23) and (24) but in addition, we define similarly $\tilde{\Phi}(\omega)$ and $\tilde{\Psi}(\omega)$ based on (27) and $\tilde{M}_1(\omega) = H_1(\omega)/\sqrt{2}$. It can be verified that, if the infinite products converge in $L_2(\mathcal{R})$, then $\Psi(\omega)$ and $\tilde{\Psi}(\omega)$ generate a biorthogonal basis through shifts and scales, and

$$\left\langle \frac{1}{2^m/2} \psi(2^m t - n), \frac{1}{2^k/2} \tilde{\psi}(2^k t - l) \right\rangle = \delta_{mk} \delta_{nl}. \quad (28)$$

An interesting question arises, namely that of the regularity of the analysis versus that of the synthesis. They can be very different and still form a valid biorthogonal expansion. However, when a truncated series expansion is considered as in a compression application, then a regular synthesis will lead to smoother approximations, which is preferable in practice. In particular, the perceptual quality is much higher when a very regular synthesis filter is used.

Fig. 5(b) shows a biorthogonal scaling function family which corresponds to length-four biorthogonal filter banks. The difference in regularity is obvious.

The next class of possible wavelet constructions involves multichannel filter banks. The easiest case is based on tree structured filter banks, which generates wavelet packets. One can think of this construction as corresponding to the iteration of lowpass filtering and subsampling, followed by a finite tree structured filter bank. The iteration part generates a scaling function, which only depends on the lowpass filter, while the final tree structured bank generates linear combinations of this scaling function. The regularity is thus given

by the lowpass filter. Consider the example of a depth-two full tree filter bank. This will generate one scaling function and three wavelets. The scaling function and the first wavelet are as in the usual two channel case, and there are two additional wavelets or wavelet packets. The key difference is that now, the scale change is by four instead of two.

More general wavelet packets are obtained from multi-channel filter banks. In that case also, the regularity of the lowpass filter is the key. Very similarly to the two channel case, the N -channel case requires the lowpass filter to have a sufficient number of zeros at the N -th roots of unity (except one), that is at $\omega = 2\pi k/N, k = 1, 2, \dots, N-1$. Then the iterated scheme can be applied to this filter bank [33]. An advantage of this construction is that one can obtain simultaneously finite support, orthogonality and linear phase, since such filter banks do exist in the case of more than two channels.

The multidimensional case is more open at this point. While Daubechies' construction can be applied in this setting as well (including nonseparable filters and non-separable lattices), a family of regular orthonormal filters similar to the 1-D Daubechies's filters has yet to be found. The generalization of the Haar case to multidimensional lattices is studied in [34] and nonseparable wavelet bases are investigated in [35], [36]. An orthonormal compactly supported wavelet basis for the quincunx lattice (nonseparable subsampling by two in R^2) constructed in [36] was shown to be continuous [37].

Finally, time-varying filter banks can be used to generate multiwavelets, a new class of wavelets where a finite number of wavelets and their shifts and scales are used [38]. This construction poses some interesting design questions, since the infinite product in (23) becomes an infinite matrix product.

D. Interaction of Continuous and Discrete Time Computations

The construction of continuous-time wavelets based on iterated discrete-time filters creates an interesting and new link between continuous-time and discrete-time signal processing. Another such link is the computation of the wavelet series expansion using filter banks. This is known as Mallat's algorithm [5]. Assume we have the projection of the original signal onto V_0 , or $x^{(0)}[n] = \langle \phi(t-n), x(t) \rangle, N \in \mathcal{Z}$. Next, we wish to compute the projection onto V_1 , or

$$x^{(1)}[n] = \left\langle \frac{1}{\sqrt{2}} \phi\left(\frac{t}{2} - n\right), x(t) \right\rangle. \quad (29)$$

Using the fact that $\phi(t)$ satisfies a two-scale equation (26), one can rewrite $\phi(t/2)$ in terms of $\phi(t)$ and its shifts. Using the orthogonality of $\phi(t)$ and its translates, it then follows that

$$x^{(1)}[n] = \sum_k g_0[k-2n] x^{(0)}[k] = \sum_k h_0[2n-k] x^{(0)}[k] \quad (30)$$

where the last equality follows since $h_0[n] = g_0[-n]$. Thus once an initial projection¹¹ is available, one can use a discrete-time algorithm to compute the wavelet expansion coefficients. Thus an exact computation of the wavelet expansion coefficients is possible thanks to the two scale equation property.

IV. ADAPTED BASES

A large class of compression algorithms exploits the fact that applying a linear transformation to the input signal can result in coding gain. Transform coding works on the principle that if \mathbf{x} is the input signal vector, and \mathbf{A} a "good" linear transform, then $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}$ is "easier" to compress (due to better energy compaction and decorrelation of the transformed signal [8], [39]), i.e., $\text{Cost}(\mathbf{y}) < \text{Cost}(\mathbf{x})$, a cost metric $\text{Cost}(\cdot)$. For compression, the appropriate metric is $J = D + \lambda R$ of a quantized version $\hat{\mathbf{y}}$ of the vector \mathbf{y} , trading off coding rate R for quantization distortion D at a quality factor determined by the Lagrange multiplier λ .

The traditional approach in compression is to use a fixed transformation \mathbf{A} (e.g., commercial image and video compression standards like JPEG and MPEG use the DCT) and then choose a quantization strategy matched to the properties of the input process and the fixed transform \mathbf{A} . For example, in image coding, the DCT is popular because it is a good approximation to the optimal KLT of Markov-one processes with high correlation, a fairly good model for images. The quantization strategy, i.e., for allocating bits to the elements of \mathbf{y} is typically based on a model for their probability density functions, or in the absence of an analytical model, from training over a large class of "typical" possible signals \mathbf{x}_i .

One choice for (a fixed) \mathbf{A} that has attained popularity in image coding due to its recent superior performance is the discrete wavelet transform. The wavelet transform gives a logarithmic decomposition of the frequency axis: this gives good frequency selectivity at lower frequencies and good time selectivity at higher frequencies. This trade-off is well suited to many signals which exhibit long-duration low-frequency events and short-duration high-frequency events. When coupled with a quantization strategy that exploits this characterization (e.g., zerotrees [40] and space-frequency quantization [41]), the wavelet-transform has achieved significantly improved performance over other transforms like the DCT and represents the current state-of-the-art in image compression.

While coding algorithms that use a fixed transformation \mathbf{A} can be useful if the class of signals \mathbf{x}_i is well suited in some sense (e.g., in TF characterization) to the fixed transform \mathbf{A} , this may not be adequate for dealing with arbitrary classes of signals with either unknown or time-varying characteristics. For example, for signals or signal segments having high frequency stationary components (e.g., localized textures, stripes, etc. in images or other periodic high-frequency phenomena), the wavelet transform

¹¹This initial projection still involves integrals or has to be approximated.

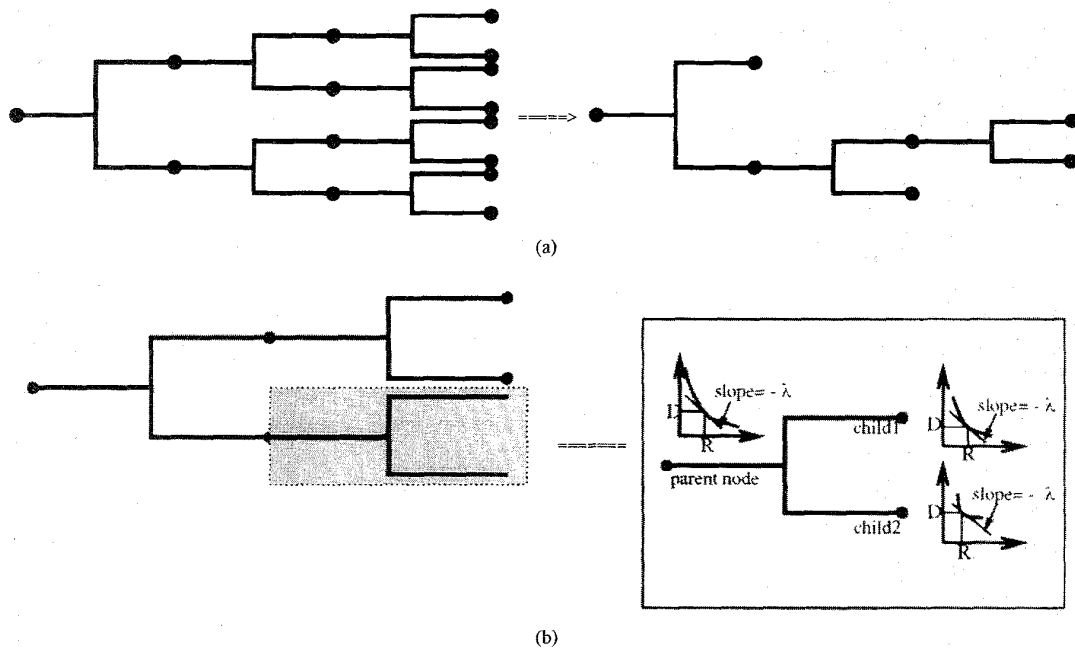


Fig. 7. The single tree algorithm finds the best tree-structured wavelet packets basis for a given signal. (a) The algorithm starts from the full STFT-like tree, and prunes back from the leaf nodes to the root node until the best pruned subtree is obtained. (b) At each node, the split-merge decision is made according to the criterion: prune if $J(\text{parentnode}) \leq [J(\text{child1}) + J(\text{child2})]$.

is a bad fit. This motivates us to consider an alternative, adaptive, and more robust approach to dealing with a large class of signals of either unknown or, more typically, time (or space-) varying characteristics. In this approach, the goal is to make the transformation signal-adaptive, i.e., to make A vary with x . That is, we seek to find

$$\min_A \text{Cost}(A \cdot x). \quad (31)$$

To minimize this over all possible A is obviously infeasible, since, even if the signal is of finite length, there are an infinite number of $N \times N$ matrices. What if however we constrain A to be a member of a very large (but finite) and useful library of transforms? And what if this large library has the added attraction of being easy to search? Then, this is clearly a more desirable transform than the original fixed transform, and it is certainly possible to find a lower cost description of the signal than when A is fixed (especially if the "old" fixed transform is included in the "new" menu of choices). Of course we must be careful to ensure that we can afford to pay for this flexibility (in terms of the overhead cost needed to identify which library entry is chosen). For a "universal" representation framework we must include the cost $L(A)$ of sending a description of the winning basis, thus the best possible cost outcome would be

$$C_{\min} = \min_A (\text{Cost}(A \cdot x) + L(A)). \quad (32)$$

For most image coding applications, the contribution of $L(A)$ to the total cost is small enough to have negligible effect on the minimization, and it can be upper bounded as an overhead of a fixed number of bits, being the logarithm to the base two of the number of elements in the library. If

the overhead $L(A)$ is not negligible, a suitable framework such as Rissanen's minimum description length (MDL) [42] (or a more practical version thereof) must be chosen in which the sum total of the bits needed to describe the model (overhead) plus the data coded with respect to the model needs to be minimized at a target coding quality. For this discussion, we will ignore the role of $L(A)$.

It deserves emphasis that we are interested in the best expansion for an arbitrary *fixed* signal and a *fixed* cost function. This is very different from the case of finding a transform A that minimizes the expected value of $\text{Cost}(A \cdot x)$ over an ensemble of signals (e.g., the Karhunen-Loève transform minimizes the mean squared error cost for Gaussian random variables using scalar quantizers). The main advantage of this approach is that no models are assumed and no training required (as in vector quantization methods), but restrictions to linear mappings of a constrained kind need to be made, and complexity is increased.

The idea of searching for good representations from a large library has something in common with the method of matching pursuit [43]. The idea of adapting the representation to the signal is related to the field of universal coding [42]. Tree structured representations have also been examined in a vector quantization context [44]. Quadtree spatial segmentations for coding have been explored in [45].

In building our library of useful signal expansions, we will take advantage of the recent insight that filter bank structures merely represent different ways of constructing bases for signal expansion. The advantages of using filter bank trees as opposed to general linear expansions are that they are efficiently implementable, and the set of all possible trees based on some filter set provides a very large

library of possible expansions (a 2-D decomposition of maximum depth of five has a library of about 5.60×10^{78} bases!). More important, due to the tree-structure of these bases, there are efficient tree-pruning algorithms to search for the best basis for a desired input and (additive) cost function. Note that we will consider orthogonal transformations, since then the MSE of the quantized coefficients equals that of the reconstructed signal, enabling examination of quantization effects directly in the transform domain [46], [8]. This is critical to the speed of the search for the least cost best basis, when the cost includes quantization distortion, as for compression applications.

As a first step toward attaining an adaptive transform, it is clear that an improvement can be found if we search over the whole set of binary trees for a particular filter set, instead of using the fixed tree of the wavelet transform. A fast algorithm to do this was employed by Coifman and Wickerhauser [47] based on measures like minimum distortion or minimum number of nonzero coefficients above a certain threshold, etc. None of these is optimal for compression, where the appropriate cost function is rate-distortion (R - D), i.e., a cost function which minimizes the coded rate for a target quality level or equivalently minimizes the quantization distortion for a target coding bit rate. Furthermore, the scheme of [47] does not address the important task of optimizing the quantization scheme, which is key to the lossy compression problem. The rate-distortion (R - D) optimized best basis search was done by Ramchandran and Vetterli in [48], where the quantization and best basis choice were jointly optimized in an operational R - D sense using the *single tree* algorithm, which is summarized now.

A. Single Tree Algorithm

A brief description of the single tree algorithm follows [48]. The aim of the algorithm is to search for the best basis (in compression efficiency sense) for the whole *unsegmented* signal from the library of wavelet packet bases. In order to achieve this, two entities are needed: a cost function for basis comparison and a fast search algorithm.

Let us first address the cost function. The compression problem is a constrained problem of minimizing the quantization distortion D for a target bit rate R and is easier solved by being converted to an unconstrained problem that involves minimizing a cost function that involves both D and R . This is the Lagrangian cost function $J = D + \lambda R$, which trades off rate for distortion at a “quality” factor given by the Lagrange multiplier $\lambda \geq 0$, which represents the absolute slope of the R - D curve. The optimal slope λ^* for a specific coding goal has to be matched to the target rate R and involves a search in the λ space, but this turns out to be a very fast search due to the convexity of the R - D function. Arbitrary (finite) quantization choices are assumed available to quantize the WP coefficients in each tree node, with both rate (R) and distortion (D) being assumed to be additive cost metrics over the WP tree: i.e., $R(\text{tree}) = \sum R(\text{leaf nodes})$ and $D(\text{tree}) = \sum D(\text{leaf nodes})$. Note that first-order entropy and MSE measures for R and D satisfy this additivity condition.

Turning now to the fast search problem, one possible approach to finding the best tree is the “greedy tree growing” algorithm, which starts at the root and divides each signal in two if it is profitable to do so (if the cost of the subsignals generated is less than the cost of the signal they come from). It terminates when no more profitable splits remain. It is easy to determine that this, however, does not find the globally optimal tree, which is found by starting at the deepest level of the tree, and pruning pairs of branches having higher total cost than that of their parent.

We now describe the details using a 1-D case for simplicity (See Fig. 7). The idea is to first grow the full (STFT-like) tree (see Fig. 7(a)) to full depth (or some maximum fixed depth in practice) for the whole signal. Note that due to the tree structure of the bases, we now have available the WP coefficients corresponding to all the bases on our search list. That is, if we grow the coefficients of a depth-five tree, we know the coefficients associated with all subtrees grown to depth-five *or less*.

The next step is to populate each WP tree node with the minimum Lagrangian cost over all quantization choices for that tree node. This minimum cost at each node is associated with the quantizer which minimizes the rate-distortion tradeoff (for a fixed “quality factor” λ)

$$J(\text{node}) = \min_{\text{quantizer}} [D(\text{node}) + \lambda R(\text{node})].$$

Note the implication of this step—we do not yet know if an arbitrary tree node will be part of our desired optimal subtree choice, but we *do* know what quantization choice to use for that node *if* it is part of the best basis subtree. This is particularly satisfying because it has enabled us to decouple the best quantizer/basis choice without sacrificing optimality.

We now have remaining only the unfinished business of finding the best basis. The special tree structure of the basis can be exploited in formulating a fast tree-based search strategy. The idea is to use a bottom-up recursive “split-merge” decision at each node, corresponding to whether it is costlier, in the Lagrangian sense, to keep the parent node or its children nodes. This fast dynamic programming (DP) based pruning method is also optimal because the signal subspace spanned by the parent node is the direct sum of the signal subspaces spanned by its children nodes thanks to the orthogonality of the filter bank. We now describe the details. Assume known the optimal subtree from a tree node n “onwards” to the full tree-depth $\log N$. Then, by Bellman’s optimality principle of DP [49], all surviving paths passing through node n must invoke this same optimal “finishing” path. There are only two contenders for the “surviving path” at every node of the tree, the parent and its children, with the winner having the lower Lagrangian cost. That is, starting from the full tree, the leaf nodes are recursively subjected to an optimal split-merge decision, following a policy of

$$\text{Prune if: } J(\text{parentnode}) \leq [J(\text{child1}) + J(\text{child2})]$$

where $J(\text{childnode})$ corresponds to the cost of the cheapest path that “goes through” the child node. Using this, we

begin at the complete tree-depth $n = \log N$ and work our way toward the root of the tree, using the above split/merge criterion at each node, making sure that we record the optimal decisions along the way, until we arrive at the tree root. At this point, the best basis is known by simply backtracking our way down the tree using our recorded decisions at each tree node. In fact, both the best basis *and* the best quantization choice are now known!

Of course, this corresponds to a particular choice of λ , which was fixed during this tree-pruning operation. Unfortunately, this λ may not be the correct one: we want the one that corresponds to the target bit budget R . However due to the convexity of the rate-distortion curve, the optimal slope λ^* matched to the desired R can be easily obtained using standard convex search techniques, e.g., the bisection method or Newton's method or other standard root-solving methods. An important point of note is that the Lagrangian method can only obtain solutions that reside on the convex-hull of the rate-distortion curve, and thus, a target rate whose optimal operating point is not on the convex hull will be approximated by the nearest convex-hull rate. In practice, for most practical coding applications, the convex hull of the R - D curve is dense enough that this approximation is almost exact.

We will now summarize the single tree algorithm:

- Grow a full balanced (STFT-like) tree to some desired fixed depth (i.e., find all the WP coefficients associated with all bases in the library);
- For a fixed λ , populate each node of the full tree with the best Lagrangian cost $D + \lambda R$ over all quantizer choices (i.e., find the best quantizer choice for each node);
- Prune the full tree recursively, starting from the leaf nodes (i.e., find the best basis subtree);
- Iterate over λ using a convex-search method to meet the target bitrate (i.e., match the best subtree/quantizer choice to the desired bit budget).

The name for this algorithm is derived from the fact that a single (frequency) tree is optimally pruned. For a signal of size N , the computational complexity of this algorithm is $O(N \log N)$. It is straightforward to extend the single tree from 1-D signals to 2-D images, and it can be shown by a simple inductive argument that the number of 2-D bases $S(d)$ searched by a single tree of depth- d is given by the recursion: $S(d) = [S(d-1)]^2 + 1$ with $S(1) = 2$. For example, a depth-five 2-D wavelet packet decomposition has a library of 5.60×10^{78} bases!

B Adaptive Wavelet Packets

The single tree algorithm finds a signal-adaptive best basis (like the Karhunen-Loève basis) in a computationally attractive framework and is a considerable improvement over the fixed basis paradigm for handling signals with diverse or unknown TF characteristics. However, the single tree effectively finds the best "static" frequency decomposition for the *unsegmented* signal taken as a whole. If the signal is nonstationary or exhibits time-varying behavior, the algorithm will choose a basis that is best in an average

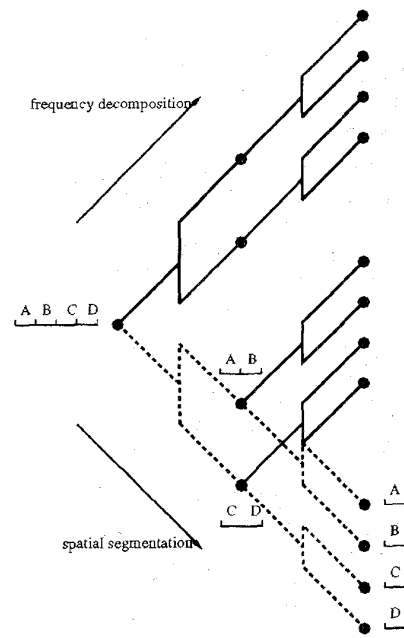


Fig. 8. A full double tree of depth-two for a 1-D signal. Dotted lines represent the spatial tree, whereas solid lines represent the frequency tree.

sort of way for the whole signal, but cannot adapt the tree locally to different signal segments. A number of signal classes in applications like speech, images, and video typically exhibit time-varying characteristics that are better handled if the frequency decompositions can be made dynamic. For image representations, adaptivity can be obtained, for example, by spatially segmenting the image and finding the best WP (frequency) tree for each spatial segment. This leads to the paradigm of *adaptive wavelet packets*, i.e., wavelet packet decompositions that adapt spatially in order to best match the signal's locally varying time- or space-frequency characteristics.

An algorithm that attacks this problem of segmenting the signal in time and in frequency was introduced in [50] as the *double tree algorithm*, which addresses the *joint* fast search for binary time-segmentations of the signal, along with the best WP frequency decomposition for each segment. The double tree thus represents a hierarchical extension of the single tree to accommodate binary time splits, so as to better address time-varying signal characteristics using time-varying filter banks.

1) *Double Tree Algorithm*: The basic idea of the double tree algorithm is simple and is based on the single tree algorithm described in the previous subsection. We explain the double tree structure again through a 1-D example in Fig. 8, while the (separable) extension of the double tree structure from 1-D signals to 2-D images is obvious. Suppose the signal consists of four quarters labeled as A, B, C, and D, respectively, in Fig. 8. We can grow a single tree on the whole signal (ABCD), or we can first segment the signal into two halves (AB and CD), and then grow individual single tree for each half, or we can further

segment the halves into quarters (A, B, C, and D) before growing single trees for the quarters, and so on. In the end, we have a redundant double tree structure for representing the original signal, allowing both tree-structured time-segmentations and frequency decompositions.

Basically, this algorithm calculates the best single-tree for the whole signal and stores the cost. It then calculates the best single-tree for the first and second halves (segmented in time) and stores the costs. It then calculates the best single-trees for the signal segmented in time into quarters and so on. These costs are written in a binary tree which is pruned using the single-tree algorithm one more time, to find the best binary time-segmentation jointly with the best WP frequency decomposition corresponding to each time segment.

The name here is derived from the two kinds of trees that are pruned, frequency trees (corresponding to the solid-line trees of Fig. 8) associated with each dyadic segment of the original signal, and temporal trees (corresponding to the dashed-line trees of Fig. 8) associated with the time-segmentations of the signal. The computational complexity of the double tree algorithm can be shown to be of $O(N(\log N)^2)$ for a size N signal.

It can be shown that the number of 2-D bases $D(d)$ searched by a double tree of depth- d is given by the recursion

$$D(d) = [D(d-1)]^4 + S(d) - S(d-1) + 1$$

with $D(1) = 2$, where $S(d)$ is the number of bases searched by a single tree of depth- d . For example, a depth-five 2-D double tree decomposition has a library of 6.5×10^{96} bases.

An example of a double tree space-frequency tiling for an image coding application for the House image is shown in Fig. 9.

Note that the side-information needed to inform the decoder of the winning basis is more than that of the single-tree but is still negligible for most practical image coding scenarios. For example, for a double tree of maximum depth- d , the number of bits in representing the tree-structured side information is $\sum_{k=1}^{d-1} 4^k$ bits, with each bit specifying the spatial/frequency split decision at each node of the best double tree. For a depth-five split for a 512×512 image, the side information for conveying the best double tree amounts to about 0.0013 b/pixel, certainly negligible for most applications.

2) *Boundary Filters*: When the filter bank tree changes over time, special treatments are necessary at the boundaries between signal segments. One way to handle boundaries is to use circular extensions. But circular extension usually causes undesirable high frequency artifacts at boundaries. Another approach, which does not introduce boundary artifacts, is to use symmetric extensions. However, symmetric extensions require that the filters be symmetric, a condition that is incompatible with having finite length orthogonal filters in the two-channel case [15] (except for two-tap filters). Since orthogonality has to be preserved across segment boundaries, symmetric extensions cannot be used in this case. So, the problem of designing an orthogonal basis across boundaries in such a time-varying system is an interesting one, and can be solved in a number of ways

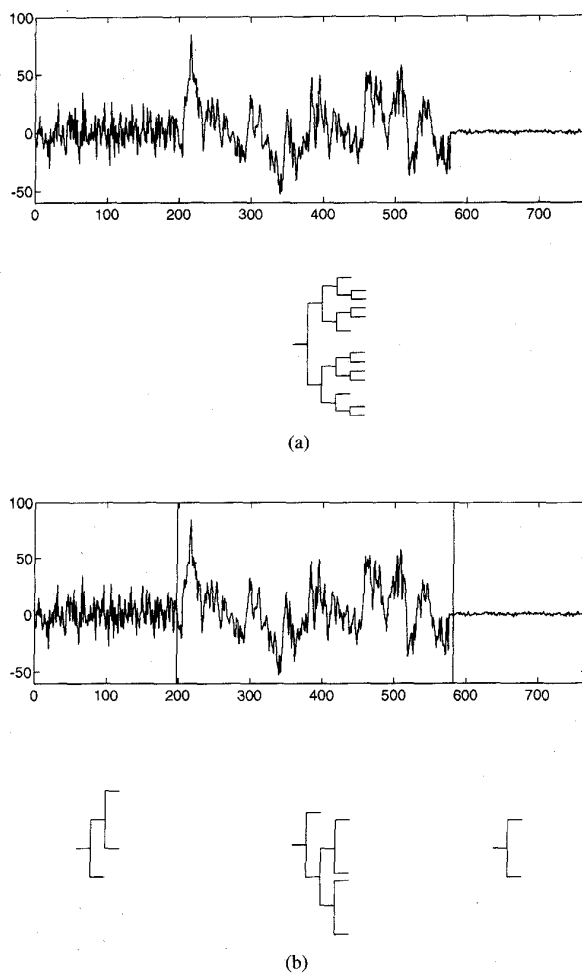


Fig. 9. Coding results using the double tree algorithm on the "house" image. (a) Original image (b) Double tree segmentation and tiling for the house at 1.0 b/p: dark lines represent spatial segments, while light lines the frequency boundaries of the wavelet packet tree. Note that the upper left corners are the lowpass bands in each spatial segment. The maximum double tree depth is five, and 341 b are sent as side information to convey the tiling information.

[51], [52]. The double-tree algorithm uses the boundary filters designed in [53].

3) *Improvements to the Double Tree*: The double tree-algorithm represents a first step toward an adaptive representation framework using time-varying bases, but it exhibits an "unbalanced" TF tree in the sense that it explores the choice of best frequency tree for every binary time-segment of the original signal (so-called nodes of the "time" tree), but it ignores the choice of time segmentation of frequency decompositions (so-called nodes of the "frequency" tree). See the double tree example of Fig. 8, where it is clear that every node of the time or spatial tree (dashed-line tree) is the root of a frequency tree (solid-line tree) but not vice versa. This limitation has been recently addressed in [54] and [55] where a balanced TF tree representation that is a generalization of the double tree has been defined together with a fast way to search for the best basis, and has been successfully applied to

image coding using a space-frequency tree data-structure. A simplified version of the TF-tree algorithm for the special case of block transforms is considered in [54], while a similar approach was used in [56] to search for the best wavelet packet basis and in [57] for the best Walsh basis.

Thus there has been an evolutionary trail in the quest for adaptive representations using linear expansions, from the fixed wavelet tree to the (full-signal) adaptive single wavelet packet tree to the (unbalanced time-segmented) double tree to the (time-and-frequency segmented) balanced TF-tree. While this is certainly an impressive evolutionary tree (no pun intended), these algorithms all have the drawback that the segmentations they consider are binary (or *M-ary* in general), which can be restrictive. To highlight this point, suppose that we are given a signal with the statistical characteristics of its first quarter A significantly different from the remaining three quarters B, C, and D, then this signal should be segmented into two subsignals, with the first one consisting of just the first quarter A, and the second one the rest three quarters BCD. However, when a binary tree-structured spatial segmentation is attempted on this signal, as in the double or TF-tree algorithm, in order to have the first quarter of the signal as a segmented subsignal, the original signal has to be segmented into two halves (AB and CD) at the first level of the tree, which is inefficient for the given statistics of the signal. Furthermore, as a direct consequence of this, the best bases obtained from these tree algorithms are very sensitive to time shifts of the original signal, an undesirable property in most situations.

Recently, a way of mitigating this shift-variance problem has been tackled through the proposal of a more flexible spatial segmentation paradigm [58], [59] that is considerably less sensitive to time shifts. In this approach, the binary time segmentation constraint is relaxed, at the cost of additional complexity, and a fast dynamic programming based algorithm is applied to search for the best "flexible" spatial segmentation tree (with the binary splits in the frequency tree remaining due to the use of two-channel filter banks, although this too can be generalized to the *M*-channel case). This flexible spatial segmentation tree represents another evolutionary step in the generalization of the existing trees, and its library of bases is a superset of all previously considered ones. More importantly, by allowing flexible segmentations, the algorithm of [59] overcomes what may be perceived to be the real shortcomings of these existing bases: the stationary nature of the wavelet and single tree bases, and the constrained nature of the time variation in the double tree and the TF tree. This flexible representation is achieved by permitting time segmentations of resolution L , i.e., time-segments can begin at any arbitrary integer multiple of the chosen L . For a 1-D signal of length N , a maximum of $M = N/L$ time segments (numbered from zero to $M - 1$) are permissible. While an exhaustive search for the best segmentation would require a complexity of $O(2^M)$, the approach of [58] uses a fast dynamic-programming based algorithm whose search complexity is only $O(M^2)$, assuming that the cost metric is additive over disjoint spatial segments, which holds for an

orthogonal basis and additive rate constraint, as achieved by using the boundary filters of [50] and a first-order entropy rate measure. We refer the reader to [59] for details of the algorithm due to lack of space in this overview article. The main idea is to find the optimal segmentation for signal $[0, kL - 1]$ recursively, based on those for all its admissible subsignals. Computational saving is achieved by using the known optimal segmentations for all previous subsignals in a recursive manner, thus avoiding an exhaustive search.

Note that all the adaptive representations addressed so far (the single tree, the double tree, the TF tree, and the DP based segmentation) are constructed (and implemented) using adaptive filter bank tree structures. In all these constructions, the trees are assumed to be based on a *fixed* filter kernel (i.e., the low-pass/high-pass filter pair in the filter bank tree). An obvious question is: what about making these filters also signal adaptive in addition to the tree topology? Signal-adaptive filter design for subband coding has been addressed in the literature [60], [61], where fast algorithms are formulated to find the optimal design of filters (FIR of a desired length) that optimize the cost function (e.g., the coding gain). A globally optimal filter bank design technique based on linear programming to maximize coding gain has been recently described in [62]. A promising avenue for future research that represents a universal approach to adaptive signal representation in this framework consists in the jointly optimal formulation of adaptive filters *and* adaptive tree structures using a framework such as Rissanen's MDL [42] to optimize the trade-off between model complexity and performance.

V. APPLICATIONS TO SIGNAL COMPRESSION

This section gives a brief overview of the state of affairs in compression using wavelet and subband techniques. There is no attempt at exhaustivity given the space limitations [63].

A. Successive Approximation Source Coding

Wavelet or subband techniques have had a significant impact on the practice of signal compression. These techniques have made it into standards that are now widely used (for example, the MUSICAM standard used in CD quality audio) and are also used in state of the art image coders.

One feature that makes such coders attractive is the possibility to decode only part of the bitstream in order to get a first approximation of the signal. This is useful in applications that involves both compression and communication, in particular in joint source-channel coding methods.

While successive approximation coding is not always the best approach [64], in many instances, the best practical coders also happen to be successive approximation coders. Actually, if transmission issues are taken into account, successive approximation coding together with appropriate communication schemes can be best both in theory and in practice, as will be discussed below.

Subband and wavelet techniques, through the concept of best bases discussed in Section IV, are also a step in

the direction of universal coding, since no model of the signal to be compressed needs to be assumed. Instead, the expansion “fits” the signal. Together with adaptive quantization and adaptive entropy coding, one achieves an adaptive transform coder which can outperform fixed coders (for example based on the DCT) in many instances. The development of such a “universal transform coder” is a topic of current research.

B. Successful Examples of Subband and Wavelet Coding

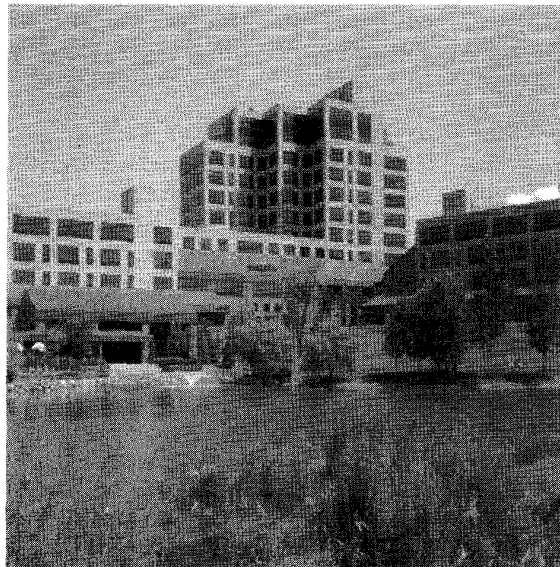
1) Wavelet/Subband-based Compression for 1-D Signals:

Among the most important 1-D signals to be compressed are speech and audio signals. While the nature of these signals is different, the receiver is in both cases the human ear. Inquiries into the psychophysics of hearing lead to models which are essentially filter banks with some nonlinearities (called masking effects). The filter bank has constant bandwidth filters at low frequencies and constant relative bandwidth at high frequencies. In our parlance, it behaves like a short-time Fourier transform up to approximately 500 Hz, and like a wavelet transform above 500 Hz.

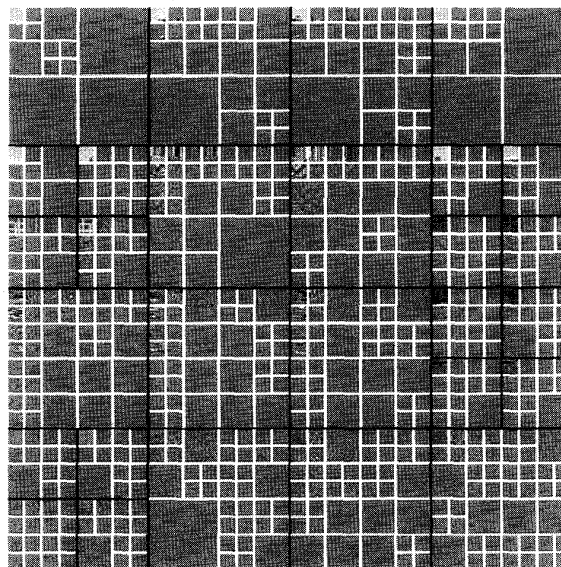
It is thus natural to use filter banks to code speech and audio signals. In the case of speech, because of the possibility of using a sophisticated speech production model, methods other than subband coding are used (e.g., adaptive linear prediction) but in the case of music, all proposed methods are variations of subband coding. The basic music coder is the MUSICAM standard [65], [66], also sometimes referred to as the MPEG-audio. Its basic form is a 32-channel short-time Fourier transform and a perceptual model to allocate bits in the channels. Variations on this theme have led to more sophisticated coders, using for example changing window sizes [67]–[69]. This leads to an adaptive change of time and frequency resolutions and allows to capture transients (attacks) precisely with short windows, while giving good compression of stationary parts using long windows. For the design of filter banks better matched to the auditory systems, see [70], and for the use of wavelet packets for audio coding, see [61].

2) *State-of-the-Art in Wavelet/Subband-Based Image Compression:* In recent years, the application of wavelets to image compression [71], [40], [72]–[75] has drawn more interest than any other application. Early wavelet-based image coding algorithms were patterned after standard transform-coding principles in that they were designed to exploit (only) the wavelet transform’s ability to decompose an image into statistically distinct *frequency bands* (i.e., to compact energy into low frequency coefficients). Coding gain was achieved by optimizing quantizers (scalar and vector) for the statistics of each subband [72], [76]. These early subband coding algorithms demonstrated modest improvements in coding efficiency over standard transform-based algorithms.

A new class of algorithms developed in the recent past has achieved significantly improved performance over the previous class [71], [40], [41], [77]–[79]. The distinctive attribute associated with this new class of coding algorithms is that while previous ones focused on only the



(a)



(b)

Fig. 10. Optimal spatial segmentation and WP frequency decomposition double tree in (a) and of the DP-based algorithm in (b) for zero-mean AR first-order Markov signal. The first quarter has variance = 100.0, and $\rho = 0.1$, the second and third quarters have variance = 100.0 and $\rho = 0.9$, and the fourth quarter has variance = 1.0, and $\rho = 0.1$. Note that the double tree degenerates to a single tree while the DP algorithm correctly segments the signal.

frequency compaction properties of the wavelet transform, the new algorithms exploit the wavelet’s space-frequency compaction properties: *the wavelet is able to both “frequency compact” energy into a small set of low frequency coefficients, and also to “spatially compact” energy into a small set of high-frequency coefficients.* In order to exploit the spatial compaction property, these coders use spatial data-structures which effectively serve as “pointing” mechanisms to identify where spatial energy is localized. While

several researchers have made active contributions to this new class of coders, we have insufficient room in this article to pay homage to them all, and we will concentrate on an illustrative few.

The most popular spatial mechanism in current state-of-the-art wavelet-based image coders is the *zerotree* data-structure that was introduced by Lewis and Knowles in [71] and popularized by Shapiro's embedded zerotree wavelet coder [40]. The zerotree data structure is conceptually very simple: it is a mechanism to indicate that a spatial region corresponding to a tree-structured set of wavelet coefficients has insignificant energy. The tree structure is evident from the spatial localization property of the wavelet transform. Following Shapiro's work in 1993, several researchers have explored the zerotree data structure for image compression. An improved modification and implementation of Shapiro's embedded zerotree coder has been recently described by Said and Pearlman [80]. A jointly optimized (in the rate-distortion sense) application of spatial (zerotree) quantization and frequency (scalar) quantization has been introduced by Xiong *et al.* [41] as a wavelet-based space-frequency quantizer (SFQ), which has obtained significant rate-distortion gains over Shapiro's coder. A novel unification of the wavelet and fractal image compression frameworks based on the wavelet zerotree data-structure has been described by Davis in [81].

The zerotree data-structure is but one mechanism to exploit the spatial compaction (or clustering) property of the wavelet representation. It accomplishes this by efficiently identifying spatial regions of insignificance (or zeros), thereby implying the clustering of significance in the remaining regions. Other mechanisms for exploiting the spatial localization of energy in the high-frequency bands have been explored by researchers, and these have resulted in high-performance coding algorithms. Servetto *et al.* [78] have proposed an image coder based on a morphological clustering data-structure that exploits the spatial localization of energy in high-frequency subbands. Joshi *et al.* [79] have introduced a spatially adaptive quantization of image subbands using block classification. There have been other image coding frameworks built around filter banks including model-based, region-based, and nonlinear approaches. Nonlinear filter bank decompositions have been proposed by Egger and Li in [82], and recently a hierarchical nonlinear filter bank based image coding algorithm has been introduced by de Queiroz *et al.* [83]. Kossentini *et al.* [84] have described a subband image coder using entropy-constrained residual vector quantization.

Finally, as a brief illustration of the incorporation of more powerful quantization strategies into adaptive transforms based on filter banks, adaptive image representations based on wavelet packets and spatially adaptive wavelet packets have been addressed in [85] and [86] using extensions of the SFQ framework of [41] to wavelet packet trees and adaptive wavelet packet trees using the single tree and double tree algorithms, respectively (see Section IV and Fig. 9).

3) *Video Compression and Other High Dimensional Signal Compression:* The problem of video compression, that is,

the compression of a sequence of images with high correlation between the images, remains an active area of research. Subband and wavelet methods are either used as a replacement of the DCT in standard motion compensated coders [87], or directly when 3-D transforms are used. The former method will not be further discussed here, since it is a variation on a classic method. Three-dimensional subband coding was proposed in [88] and has since enjoyed a certain popularity due to its simplicity and multiresolution nature.

One approach that keeps both the multiresolution nature of subband coding as well as the advantage of motion compensation is pyramid coding of video [89], [90]. This leads to a scheme for high quality coding of HDTV and digital broadcast.

True 3-D subband coding methods have been further investigated with interesting recent results [91]–[93]. The inclusion of motion models within 3-D subband coding is not easy [94], [3].

Another signal with high dimensionality that requires high quality compression is multispectral imagery as generated by remote sensing. See [95] for an investigation of transforms and subband coding for this case. Similarly, some medical imaging applications like tomography or nuclear magnetic resonance imaging require high dimensional high quality compression, and three dimensional wavelet or subband coding is a possible scheme.

C. Joint Source/Channel Coding

One of the important applications for the above-mentioned source coding methods, using wavelets and their adaptive extensions, is for transmission of images or video signals over transport channels like the telephone line or the terrestrial broadcast channel or over heterogeneous networks like asynchronous transfer mode (ATM) or the Internet. Most transmission schemes are designed using separately designed source coding and channel coding techniques. This tradition of separating the source and channel coding tasks arises from the solid theoretical foundation of Shannon's celebrated *separation principle* of source and channel coding, which basically states that this separation is optimal. For example, in a point-to-point transmission using a known, time-invariant channel (such as a telephone line), one can design the best possible channel coding method to approach channel capacity: i.e., achieve a rate R bits/second such that $R \leq C$ where C is the channel capacity in bits/second [96]. Then the task of the source coder is to simply do the best job it can in compressing the input source signal so that the compressed bit rate will match the rate of the channel. No scheme can do better theoretically in this scenario.

There are, however, several reasons not to adhere blindly to the separation principle. For one thing, the key word under which Shannon's results hold is "theoretically." This actually implies having infinitely long and complex codes which are obviously practically infeasible in real finite-complexity systems having strict finite delay constraints (e.g., for video). However there are other, even theoretical, reasons to consider doing joint source and channel coding.

If the channel is time-varying or when multiple users are present as in broadcast or multicast, it is even theoretically better to jointly design the source and channel coder.

The detailed treatment of this topic is obviously beyond the scope of this overview article on wavelets (if the reader is wondering how they are even remotely related, we beg for some patience) and falls under a well-studied branch of information theory called multiuser information theory [97]. We will simply illustrate through examples how multiresolution source coding fits naturally into joint source-channel coding principles. We will use the examples of digital broadcast and of packet video. An important concept in the transmission or channel coding scheme will use a principle that we coin *multiresolution transmission*, which can be construed as a dual of the multiresolution source coding ideas described earlier.

Multiresolution transmission is based on the idea that a transmission system can operate at different rates, depending on the channel conditions, or that certain bits will have better noise immunity than other bits in the case of adverse channel conditions. Such a transmission system can be achieved using different techniques, depending on the transport medium. For example, unequal error protection codes can be used, making the important bits more robust than others in the case of a noisy channel. The combination of such a transmission scheme with a multiresolution source coder is very natural. The multiresolution source coder segments the information into an “important” part which reconstructs a coarse, first approximation of the signal (such as the lowpass channel in a subband coder) as well as a part which gives the additional detail signal (typically, the higher frequencies). The coarse approximation is now sent using the highly protected bits and has a high probability of arriving successfully, while the detail information will only arrive if the channel condition is good. The scheme generalizes to more levels of quality in an obvious manner. This intuitive matching of successive approximation of the source to different transmission rates depending on the quality of the channel is called *multiresolution joint source-channel coding*.

1) *Digital Broadcast*: As a first example, we consider digital broadcast. This is a typical instance of a multiuser channel, since a single emitter sends to many users, each with a different channel capacity. One can of course design a digital communication channel that is tailored for the worst-case situation, but that is somewhat of a waste for the users with better channels. For simplicity, consider two classes of users U_1 and U_2 having “good” and “bad” channels, with capacities $C_1 > C_2$, respectively. Then, the idea is to superimpose information for the users with the good channel on top of the information that can be received by the users with the bad channel (which can also be decoded by the former class of users) [97]. Interestingly, this simple idea improves the joint capacity of both classes of users over simply multiplexing between the two channels (sending information at rate $R_1 \leq C$ to U_1 part of the time, and then at rate $R_2 \leq C_2$ to U_1 and U_2 the rest of the time). See Fig. 11(a) for a graphical description

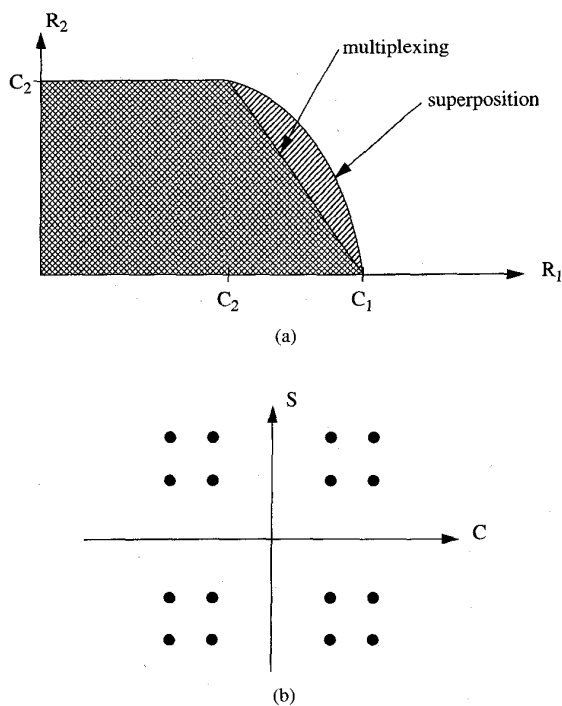


Fig. 11. Digital broadcast. (a) Joint capacity region for two classes of users with channel capacities C_1 and C_2 , respectively, and $C_1 > C_2$. Any point on or below the curves is achievable, but superposition outperforms multiplexing. (b) Example of a signal constellation (showing amplitudes of cosine and sine carriers in a digital communication system) using superposition of information. As can be seen, there are four clouds at four points each. When the channel is good, 16 points can be distinguished, (or 4-b of information), while under adverse conditions, only the clouds are seen (or 2-b of information).

of the joint capacity region and Fig. 11(b) for a typical constellation used in digital transmission where information for the users with better channels is superimposed over information which can be received by both classes of users. Now, keeping our multiresolution paradigm in mind, it is clear that we can send coarse signal information to both classes of users, while superposing detail information that can be taken by the users with the good channel. In [7], a digital broadcast system for HDTV was designed using these principles, including multiresolution video coding [90] and multiresolution transmission with graceful degradation (using constellations similar to the one in Fig. 11(b)).

2) *Time-Varying Channels*: The principles just described can be used for transmission over unknown time-varying channels. Instead of transmitting assuming the worst case channel, one can superpose information decodable on a better channel, in case the channel is actually better than worst case. On average, this will be better than simply assuming worst case all the time. As an example, consider a wireless channel without feedback (i.e., the receiver knows the state of the channel but not the transmitter). Because of the changing location of the user, the channel can vary greatly, and the worst case channel can be very poor. Superposition allows to deliver different levels of quality, depending on how good the reception actually is. An

example of multiresolution joint source channel coding for the case of time-varying channels without feedback using the idea of embedded constellations of Fig. 11(b) can be found in [98].

When there is feedback (as in two-way wireless communication), then one can use a channel coding optimized for the current channel [99]. The source coder then has to adapt to the current transmission rate, which again is easy to achieve using multiresolution source coding. A study of wireless video transmission using a two resolution video source coder can be found in [100].

3) *Packet Video*: Another example of application of multiresolution coding for transmission is found in real-time services such as voice and video over ATM networks. The problem is that packet transmission can have greatly varying delays as well as packet losses. However, it is possible to protect certain packets (for example, using priorities). Again, the natural idea is to use multiresolution source coding and put the coarse approximation into high priority so that it will almost surely be received. The detail information is carried with lower-priority packets and will only arrive when the network has enough resources to carry them. Such an approach can lead to substantial improvements over nonprioritized transmission [101]. In video compression, this approach is often called layered coding, with the layers corresponding to different levels of approximation (typically, two layers are used) and different layers having different protections for transmission.

4) *Multicast Over the Internet*: It can be argued that because of increasing interconnectivity and heterogeneity, traditional fixed-rate coding and transmission will be more and more replaced by flexible multiresolution source coding and multiple or variable-rate transmission. The main advantage is the added flexibility, which will allow users with different requirements to be interconnected through a mixture of possible channels.

An example of such flexible coding and transmission is the video coding scheme and protocol described in [102] for multicasting or broadcasting over the internet. This is built on top of a multicast facility on the internet called MBone. The Internet or rather the collection of networks that constitutes the Internet is a very heterogeneous environment. Some users might have access to MBone at several hundred kb/s, while others might only get 64 kb/s (ISDN) or even only 28 kb/s (modem).

Just like in the digital broadcast scenario discussed earlier, multicasting can then be done either at the lowest common bitrate or using multiresolution techniques. In the former case, the least possible quality is given to all users. In the latter case, each user gets the best possible quality given his bitrate. The coder in [102] uses some simple biorthogonal subband/wavelet coder with high robustness to packet loss to achieve a discrete set of bitrates to which users can subscribe, depending on their particular channel capacities.

ACKNOWLEDGMENT

The authors would like to thank Jelena Kovačević for her help and patience.

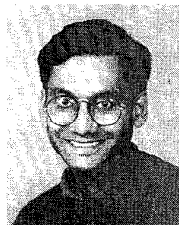
REFERENCES

- [1] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia: SIAM, 1992.
- [2] Z. Cvetković, *Overcomplete Expansions for Digital Signal Processing*. Ph.D. dissertation, Univ. Calif. Berkeley, 1995.
- [3] M. Vetterli and Kovačević, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
- [4] A. Papoulis, *Signal Analysis*. New York: McGraw-Hill, 1977.
- [5] S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Patt. Recog. and Mach. Intell.*, vol. 11, pp. 674–693, July 1989.
- [6] Y. Meyer, *Ondelettes et Opérateurs*. Paris: Hermann, 1990, vol. 1 and 2.
- [7] K. Ramchandran, A. Ortega, K. M. Uz, and M. Vetterli, "Multiresolution broadcast for digital HDTV using joint source-channel coding," *IEEE J. Select. Areas in Commun.*, vol. 11, pp. 6–23, Jan. 1993.
- [8] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston: Kluwer, 1992.
- [9] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [10] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [11] A. Croisier, D. Esteban, and C. Galand, "Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques," in *Int. Conf. on Inform. Sci. and Syst.*, Patras, Greece, Aug. 1967, pp. 443–446.
- [12] M. J. T. Smith and T. P. Barnwell III, "A procedure for designing exact reconstruction filter banks for tree structured sub-band coders," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, San Diego, CA, Mar. 1984.
- [13] ———, "Exact reconstruction for tree-structured subband coders," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. ASSP-34, pp. 431–441, June 1986.
- [14] F. Mintzer, "Filters for distortion-free two-band multirate filter banks," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. ASSP-33, pp. 626–630, June 1985.
- [15] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. on Pure and Appl. Math.*, vol. 41, pp. 909–996, Nov. 1988.
- [16] P. P. Vaidyanathan, "Quadrature mirror filter banks, M-band extensions and perfect reconstruction techniques," *IEEE ASSP Mag.*, vol. 4, pp. 4–20, July 1987.
- [17] P. P. Vaidyanathan and P.-Q. Hoang, "Lattice structures for optimal design and robust implementation of two-channel perfect reconstruction filter banks," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 36, pp. 81–94, Jan. 1988.
- [18] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser, "Signal processing and compression with wavelet packets," Tech. Rep., Dept. Math., Yale Univ., 1991.
- [19] A. Cohen, I. Daubechies, and J.-C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Commun. on Pure and Appl. Math.*, vol. 45, pp. 485–560, 1992.
- [20] M. Vetterli and C. Herley, "Wavelets and filter banks: Theory and design," *IEEE Trans. Signal Process.*, vol. 40, pp. 2207–2232, Sept. 1992.
- [21] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Process.*, vol. 1, pp. 205–220, Apr. 1992.
- [22] J. D. Villasenor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," *IEEE Trans. Image Process.*, vol. 4, pp. 1053–1060, Aug. 1995.
- [23] P. Moulin, "A multiscale relaxation algorithm for SNR maximization in 2-D nonorthogonal subband coding," *IEEE Trans. Image Process.*, vol. 4, pp. 1269–1281, Sept. 1995.
- [24] C. Herley, *Wavelets and Filter Banks*. Ph.D. dissertation, Columbia Univ., 1993.
- [25] C. Herley and M. Vetterli, "Wavelets and recursive filter banks," *IEEE Trans. Signal Process.*, vol. 41, pp. 2536–2556, Aug. 1993.
- [26] M. J. T. Smith, "IIR analysis/synthesis systems," in *Subband Image Coding*, J. W. Woods, Ed. Boston: Kluwer, 1991, pp. 101–142.
- [27] C. Heil and D. Walnut, "Continuous and discrete wavelet transforms," *SIAM Rev.*, vol. 31, pp. 628–666, 1989.
- [28] G. Battle, "A block spin construction of ondelettes. Part I: Lemarié functions," *Commun. Math. Phys.*, vol. 110, pp. 601–615, 1987.

- [29] —, "A block spin construction of ondelettes. Part II: the QFT connection," *Commun. Math. Phys.*, vol. 114, pp. 93–102, 1988.
- [30] P. G. Lemarié, "Ondelettes à localisation exponentielle," *J. Math. Pures et Appl.*, vol. 67, pp. 227–236, 1988.
- [31] J.-O. Stromberg, "A modified Franklin system and higher order spline systems on R^N as unconditional bases for Hardy spaces," in *Proc. Conf. in Honor of A. Zygmund*, W. Beckner et al., Eds., pp. 475–493, Wadsworth Mathematics series, 1982.
- [32] O. Rioul, "Regular wavelets: A discrete-time approach," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3572–3578, Dec. 1993.
- [33] P. Steffen, P. N. Heller, R. A. Gopinath, and C. S. Burrus, "Theory of m-band wavelet bases," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3497–3511, Dec. 1993.
- [34] K. Gröchenig and W. R. Madych, "Multiresolution analysis, Haar bases and self-similar tilings of R^n ," *IEEE Trans. Inform. Theory*, Special Issue on Wavelet Transforms and Multires. Signal Anal., vol. 38, pp. 556–568, Mar. 1992.
- [35] A. Cohen and I. Daubechies, "Nonseparable bidimensional wavelet bases," *Rev. Mat. Iberoamericana*, vol. 9, no. 1, pp. 51–137, 1993.
- [36] J. Kovačević and M. Vetterli, "Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for R^n ," *IEEE Trans. Inform. Theory*, Special Issue on Wavelet Transforms and Multiresol. Signal Anal., vol. 38, pp. 533–555, Mar. 1992.
- [37] L. F. Villemoes, "Continuity of nonseparable quincunx wavelets," *J. Appl. and Comput. Harmonic Anal.*, 1994.
- [38] J. S. Geronimo, D. P. Hardin, and P. R. Massopust, "Fractal functions and wavelet expansions based on several scaling functions," *J. Approx. Theory*, 1995.
- [39] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [40] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3445–3462, Dec. 1993.
- [41] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Joint optimization of scalar and tree-structured quantization of wavelet image decomposition," in *Proc. 27th Ann. Asilomar Conf. on Sig., Syst., and Comp.*, vol. 2, Pacific Grove, CA, pp. 891–895, Nov. 1993.
- [42] J. Rissanen, "Universal coding, information, prediction, and estimation," *IEEE Trans. Inform. Theory*, vol. 30, pp. 629–636, July 1984.
- [43] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3397–3415, Dec. 1993.
- [44] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 299–315, Mar. 1986.
- [45] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *Proc. ICASSP*, vol. 4, pp. 2661–2664, May 1991.
- [46] N. J. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood-Cliffs, NJ: Prentice-Hall, 1984.
- [47] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory*, Special Issue on Wavelet Transforms and Multires. Signal Anal., vol. 38, pp. 713–718, Mar. 1992.
- [48] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Process.*, vol. 2, pp. 160–175, Apr. 1993.
- [49] R. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton Univ. Press, 1957.
- [50] C. Herley, J. Kovačević, K. Ramchandran, and M. Vetterli, "Tilings of the time-frequency plane: Construction of arbitrary orthogonal bases and fast tiling algorithms," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3341–3359, Dec. 1993.
- [51] R. L. de Queiroz and K. R. Rao, "Time-varying lapped transforms and wavelet packets," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3293–3305, Dec. 1993.
- [52] A. Cohen, "Ondelettes, analyses multiresolutions et traitement numerique du signal," Ph.D. dissertation, Université Paris IX Dauphine, Paris, France, 1990.
- [53] C. Herley and M. Vetterli, "Orthogonal time-varying filter banks and wavelet packets," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 42, pp. 2650–2663, Oct. 1994.
- [54] C. Herley, Z. Xiong, K. Ramchandran, and M. Orchard, "An efficient algorithm to find a jointly optimal time-frequency segmentation using time-varying filter banks," in *Proc. ICASSP'95*, Detroit, MI, May 1995.
- [55] —, "Joint space-frequency segmentation using balanced wavelet packet trees for least-cost image representation," *IEEE Trans. Image Process.*, to be published.
- [56] J. Smith and S.-F. Chang, "Frequency and spatially adaptive wavelet packets," in *Proc. ICASSP'95*, Detroit, MI, pp. 2233–2236, May 1995.
- [57] C. M. Thiele and L. F. Villemoes, "A fast algorithm for adapted time-frequency tilings," *J. Appl. and Comput. Harmonic Anal.*, to be published.
- [58] Z. Xiong, K. Ramchandran, C. Herley, and M. T. Orchard, "Flexible time segmentations for time-varying wavelet packets," *IEEE Proc. Int. Symp. on Time-Frequency and Time-Scale Anal.*, pp. 9–12, Oct. 1994.
- [59] —, "Flexible time segmentations for time-varying wavelet packets," *IEEE Trans. Image Process.*, to be published.
- [60] P. Delsarte, B. Macq, and D. T. M. Sloock, "Signal-adapted multiresolution transform for image coding," *IEEE Trans. Inform. Theory*, Special Issue on Wavelet Transforms and Multiresolution Signal Anal., vol. 38, pp. 897–903, Mar. 1992.
- [61] D. Sinha and A. H. Tewfik, "Low bit rate transparent audio compression using adapted wavelets," *IEEE Trans. Signal Process.*, Special Issue on Wavelets and Signal Process., vol. 41, pp. 3463–3479, Dec. 1993.
- [62] P. Moulin, "A new look at signal-adapted qmf bank design," in *Proc. IEEE ICASSP*, Detroit, MI, May 1995.
- [63] P. C. Cosman, R. M. Gray, and M. Vetterli, "Vector quantization of image subbands: A survey," *IEEE Trans. Image Process.*, to be published.
- [64] W. H. Equitz and T. M. Cover, "Successive refinement of information," *IEEE Trans. Inform. Th.*, vol. 37, pp. 269–275, Mar. 1991.
- [65] Y. F. Dehery, M. Lever, and P. Urcum, "A MUSICAM source codec for digital audio broadcasting and storage," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Toronto, Canada, pp. 3605–3608, May 1991.
- [66] G. Stoll and F. Dehery, "High quality audio bit rate reduction family for different applications," in *Proc. IEEE Int. Conf. Commun.*, pp. 937–941, Apr. 1990.
- [67] J. D. Johnston and A. J. Ferreira, "Sum-difference stereo transform coding," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, San Francisco, CA, Mar. 1992, pp. 569–572.
- [68] M. Bosi and G. Davidson, "High-quality, low-rate audio transform coding for transmission and multimedia applications," in *Convent. AES*, San Francisco, CA, Oct. 1992.
- [69] C. C. Todd et al., "AC-3: Flexible perceptual coding for audio transmission and storage," in *Convent. AES*, Amsterdam, Feb. 1994.
- [70] J. Princen, "The design of nonuniform modulated filter banks," *IEEE Trans. Signal Process.*, to be published.
- [71] A. S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Process.*, vol. 1, pp. 244–250, Apr. 1992.
- [72] J. W. Woods and S. D. O'Neil, "Sub-band coupling of images," *IEEE Trans. Acoust., Speech, and Signal Process.*, vol. 34, pp. 1278–1288, May 1986.
- [73] P. H. Westerink, J. Biemond, and D. E. Boekee, "Subband coding of color images," in *Subband Image Coding*, J. W. Woods, Ed. Boston: Kluwer, 1991, pp. 193–228.
- [74] Y. H. Kim and J. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Process.*, vol. 1, pp. 31–48, Jan. 1992.
- [75] M. Antonini, M. Barlaud, and P. Mathieu, "Image coding using lattice vector quantization of wavelet coefficients," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Toronto, Canada, pp. 2273–2276, May 1991.
- [76] P. H. Westerink, J. Biemond, and D. E. Boekee, "An optimal bit allocation algorithm for sub-band coding," in *Proc. ICASSP*, pp. 757–760, 1988.

- [77] A. Said and W. Pearlman, "Image compression using the spatial-orientation tree," in *Int. Symp. Circ. and Syst.*, May 1993, Chicago, vol. 1, pp. 279-282.
- [78] S. Servetto, K. Ramchandran, and M. T. Orchard, "Morphological representation of wavelet data for image coding," in *Proc. ICASSP'95*, Detroit, MI, May 1995.
- [79] R. L. Joshi, T. R. Fisher, and R. H. Bamberger, "Optimum classification in subband coding of images," in *Proc. ICIP'94*, Austin, TX, Nov. 1994, pp. 883-887.
- [80] A. Said and W. A. Pearlman, "A new fast and efficient image coder based on set partitioning in hierarchical trees," *IEEE Trans. Circ. and Syst. for Video Technol.*, to be published.
- [81] G. Davis, "Self-quantized wavelet subtrees: a wavelet-based theory for fractal image compression," in *Data Compress. Conf. '95*, Snowbird, UT, 1995, pp. 232-241.
- [82] O. Egger and W. Li, "Very low bit rate image coding using morphological operators and adaptive decompositions," in *Proc. ICIP'94*, Nov. 1994, Austin, TX, vol. 2, pp. 326-330.
- [83] R. L. de Queiroz, D. Florencio, and R. W. Schafer, "Non-expansive pyramid for image coding using a nonlinear filter bank," *IEEE Trans. Image Process.*, to be published.
- [84] F. Kossentini, W. Chung, and M. Smith, "Subband image coding using entropy-constrained residual vector quantization," *J. Inform. Process. and Manage.*, vol. 30, July 1994.
- [85] Z. Xiong, K. Ramchandran, M. T. Orchard, and K. Asai, "Wavelet packets-based image coding using joint space-frequency quantization," in *Proc. ICIP'94*, Nov. 1994, Austin, TX, vol. 3, pp. 324-328.
- [86] K. Ramchandran, Z. Xiong, K. Asai, and M. Vetterli, "Adaptive transforms for image coding using spatially-varying wavelet packets," *IEEE Trans. Image Process.*, to be published.
- [87] A. Netravali and B. Haskell, *Digital Pictures*. New York: Plenum, 1988.
- [88] G. Karlsson and M. Vetterli, "Three-dimensional subband coding of video," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, Apr. 1988, New York, NY, pp. 1100-1103.
- [89] K. M. Uz, M. Vetterli, and D. LeGall, "Interpolative multiresolution coding of advanced television with compatible subchannels," *IEEE Trans. CAS for Video Technol., Special Issue on Signal Process. for Advanced Television*, vol. 1, pp. 86-99, Mar. 1991.
- [90] K. M. Uz, "Multiresolution systems for video coding," Ph.D. dissertation, Columbia Univ., New York, May 1992.
- [91] C. Podilchuk, N. Jayant, and N. Farvardin, "Three-dimensional subband coding of video," *IEEE Trans. Image Process.*, to be published.
- [92] C. I. Podilchuk, "Low-bit rate subband video coding," in *Proc. IEEE Int. Conf. on Image Process.*, vol. 3, Austin, TX, pp. 280-284, Nov. 1994.
- [93] D. Taubman and A. Zakhor, "Multi-rate 3-D subband coding of video," *IEEE Trans. Image Process., Special issue on Image Sequence Compress.*, vol. 3, pp. 572-588, Sept. 1994.
- [94] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Process., Special issue on Image Sequence Compress.*, vol. 3, pp. 559-571, Sept. 1994.
- [95] D. Tretter and C. A. Bouman, "Optimal transforms for multispectral and multilayer image coding," *IEEE Trans. Image Process.*, vol. 4, pp. 285-308, Mar. 1995.
- [96] C. E. Shannon, "Communications in the presence of noise," *Proc. IRE*, vol. 37, pp. 10-21, Jan. 1949.
- [97] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
- [98] I. Kozintsev and K. Ramchandran, "Multiresolution joint source-channel coding using embedded constellations for power-constrained time-varying channels," in *Proc. ICASSP'96*, Atlanta, GA, May 1996.
- [99] A. Goldsmith and P. Varaiya, "Capacity of time-varying channels with estimation and feedback," *IEEE Trans. Inform. Theory*.
- [100] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, "Robust low bit-rate video transmission over wireless access systems," in *Proc. ICC*, vol. 1, pp. 571-575, May 1994.
- [101] M. W. Garrett and M. Vetterli, "Joint source/channel coding of statistically multiplexed real-time services on packet networks," *IEEE/ACM Trans. Network.*, vol. 1, pp. 71-80, Feb. 1993.

- [102] S. McCanne and M. Vetterli, "Joint source/channel coding for multicase packet video," in *Proc. ICIP'95*, Washington, DC, Oct. 1995.



Kannan Ramchandran received the B.S.E.E. degree from the City College of New York, and the M.S.E.E. and Ph.D. (electrical engineering) degrees from Columbia University, New York, NY, in 1982, 1984, and 1993, respectively.

From 1982 to 1984 he was with IBM, and from 1984 to 1990 he was a member of the Technical Staff at AT&T Bell Laboratories' Telecommunications R&D Department. From 1990 to 1993 he was a Graduate Research Assistant at the Center for Telecommunications Research,

Columbia University. Since 1993 he has been with the University of Illinois at Urbana-Champaign, where he is currently an Assistant Professor in the Electrical and Computer Engineering Department and a Research Assistant Professor at the Beckman Institute and the Coordinated Science Laboratory. His research interests include image and video compression, multiresolution signal processing and wavelets, and image communications.

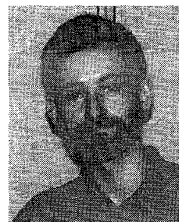
Dr. Ramchandran received the 1993 Eliahu I. Jury Award from Columbia University for the best doctoral dissertation in the area of systems, signal processing, or communications. He received the NSF Research Initiation Award in 1994.



Martin Vetterli (Fellow, IEEE) received the Dipl. El.-Ing. degree from the ETH Zürich, Switzerland, in 1981, the M.S. degree from Stanford University, Stanford, CA, in 1982, and the Doctorat ès Science degree from EPF Lausanne, Switzerland, in 1986.

He was a Research Assistant at Stanford and EPFL, and worked for Siemens and AT&T Bell Laboratories. From 1986 to 1993 he was with Columbia University, New York, NY, where he was Associate Professor of Electrical Engineering. In 1993 he became Acting Associate Professor of Electrical Engineering at the University of California at Berkeley. He is presently with the Department of Electrical Engineering and Computer Science at the Swiss Federal Institute of Technology, Lausanne, Switzerland. His research interests include wavelets, multirate signal processing, computational complexity, signal processing for telecommunications, as well as digital video processing and compression. He is an editorial board member of *Signal Processing*, *Image Communication*, *Annals of Telecommunications*, *Applied and Computational Harmonic Analysis*, and *The Journal of Fourier Analysis and Applications*. He is the co-author (with J. Kovačević) of *Wavelets and Subband Coding* (Prentice Hall, 1995).

Dr. Vetterli received the Best Paper Award of EURASIP in 1984, the Research Prize of the Brown Boverly Corporation in 1986, and the IEEE Signal Processing Society's 1991 Senior award for a 1989 paper (with D. LeGall). He is a member of SIAM and ACM.



Cormac Herley (Member, IEEE) received the B.E. degree from University College, Cork, Ireland, in 1985 and the M.S.E.E. from Georgia Institute of Technology, Atlanta, GA, in 1987. He received the M.Phil. and Ph.D. degrees from Columbia University, New York, NY, in 1992 and 1993, respectively.

Since 1994 he has been with Hewlett Packard Laboratories, Palo Alto, CA. He also teaches at the University of California at Berkeley, and is an Associate Editor for the IEEE TRANSACTIONS

ON INFORMATION THEORY.