MULTIWAVELET FILTER BANKS FOR DATA COMPRESSION

P. N. Heller[1] V. Strela[1,2] G. Strang[2] P. Topiwala[3] C. Heil[3,4] L. S. Hills[1]

[1] Aware, Inc., One Memorial Dr., Cambridge, MA 02142 USA
[2] Math Dept., Mass. Inst. of Technology, Cambridge, MA 02138 USA
[3] The MITRE Corp., Bedford, MA 01730 USA
[4] Math Dept., Georgia Inst. of Technology, Atlanta, GA 30332 USA

ABSTRACT

This paper investigates the emerging notion of multiwavelets in the context of multirate filter banks, and applies a multiwavelet system to image coding. Multiwavelets are of interest because their constituent filters can be simultaneously symmetric and orthogonal (this combination is impossible for 2-band PR-QMFs), and because one can obtain higher orders of approximation (more vanishing wavelet moments) for a given filter length. We develop symmetric extension methods for finite-length signals under multiwavelet filtering. Techniques are then presented for pre- and post-processing one-dimensional signals in order to effectively exploit multiwavelet structures. Finally, we employ these new tools in a transform coding system and compare their performance with Daubechies' scalar wavelets.

I. Introduction

Wavelets as a cascade of multirate filter banks provide a multiresolution decomposition. Discrete-time FIR filters lead to continuous-time basis functions. Such wavelet decompositions are especially effective for transform-based image coding because of their correspondence with models of human visual perception. Image compression systems work best with symmetric filters, which eliminate artifacting due to image boundaries. However, nontrivial symmetric 2-band orthogonal wavelets do not exist. In order to obtain symmetry we can choose among biorthogonal filters [1, 12], many-channel filters, and multiwavelets.

Multiwavelets – systems with two or more scaling functions spanning the "lowpass" space – offer advantages of short support, symmetry, and orthogonality. In the paragraphs to follow we develop new methods of symmetric extension and signal preprocessing which enable one to use a multiwavelet filter bank in a transform-based image coder. While block filter banks have been discussed in the past [4], issues of symmetric extension and filter design have not previously been addressed. Furthermore, the use of multiwavelet filters in a cascade algorithm leads to a novel pre- and post-processing method for block filter banks, based on the sampling/interpolation theory of wavelets. Finally, we apply multiwavelets to the compression of images.

II. Multiwavelets and Filter Banks

A multiwavelet filter bank [10] can be thought of as an M-channel filter bank with filter "taps" that are $N \times N$ matrices. In this paper, we will be working with $M = N = 2$. Our principal example, enumerated in Table 1, is the 4-coefficient symmetric multiwavelet filter bank whose lowpass filter was reported in [5]. This filter is given by four 2×2 matrices $c[k]$. Unlike a scalar 2-band paraunitary filter bank, the corresponding highpass filter (specified by four 2×2 matrices $d[k]$) cannot be obtained simply as an "alternating flip" of the lowpass filter; the $d[k]$ must be designed [10]. The resulting 2-channel, 2×2 matrix filter bank operates on *two* input data streams, filtering them into *four* output streams, each of which is downsampled by a factor of 2. This is shown in Figure 1. Each row of the multifilter is a combination of two ordinary filters, one operating on the first data stream and the other operating on the second. For example, the first lowpass multiwavelet filter in Table 1 operates as $c_{0,0}[k]$ on the first input stream and $c_{0,1}[k]$ on the second.

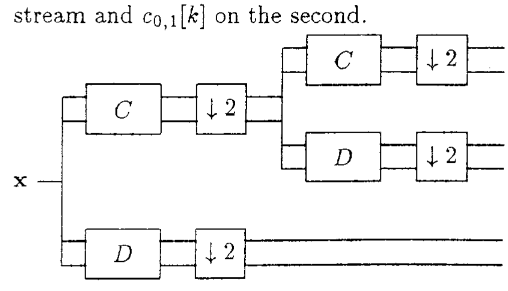

Figure 1: A multiwavelet filter bank, iterated once.

We ask that the matrix filter coefficients satisfy the orthogonality ("block-paraunitarity") condition

$$\sum_{k=0}^{N-1} c[k]c[k-2l]^T = 2\delta_{0,l} I \ . \tag{1}$$

In the time domain, filtering followed by downsampling yields an infinite lowpass matrix with "double

$$\begin{bmatrix} c[0] & c[1] & c[2] & c[3] \\ d[0] & d[1] & d[2] & d[3] \end{bmatrix} = \frac{1}{10\sqrt{2}} \left[\begin{array}{cc|cc|cc|cc} 6\sqrt{2} & 16 & 6\sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ -1 & -3\sqrt{2} & 9 & 10\sqrt{2} & 9 & -3\sqrt{2} & -1 & 0 \\ \hline -1 & -3\sqrt{2} & 9 & -10\sqrt{2} & 9 & -3\sqrt{2} & -1 & 0 \\ \sqrt{2} & 6 & -9\sqrt{2} & 0 & 9\sqrt{2} & -6 & -\sqrt{2} & 0 \end{array}\right]$$

Table 1: Linear-phase 2-band multiwavelet filter bank with 2 vanishing moments

shifts":

$$L = \begin{bmatrix} \cdots & & & & & & \\ & c[3] & c[2] & c[1] & c[0] & & \\ & & & c[3] & c[2] & c[1] & c[0] \\ & & & & & & \cdots \end{bmatrix}.$$

Each of the filter taps $c[k]$ is a 2×2 matrix. The eigenvalues of the matrix L are critical for the transition to wavelets – if L has 1 as an eigenvalue, then there is an associated 2-element vector of scaling functions $\Phi = (\phi_1(t), \phi_2(t))$. This vector is the solution to the matrix dilation equation

$$\begin{bmatrix} \phi_1(t) \\ \phi_2(t) \end{bmatrix} = \sum_k c[k] \begin{bmatrix} \phi_1(2t-k) \\ \phi_2(2t-k) \end{bmatrix}. \qquad (2)$$

These two functions (for the multiwavelet system of Table 1) are shown in Figure 2. The span of integer translates of these functions is the "lowpass" space V_0, the set of *scale-limited signals* [6]. Any continuous-time function $f(t)$ in V_0 can be expanded as a linear combination

$$f(t) = \sum_n v_{1,n}^{(0)} \phi_1(t-n) + v_{2,n}^{(0)} \phi_2(t-n). \qquad (3)$$

The superscript (0) denotes an expansion "at scale level 0." $f(t)$ is completely described by the sequences $\left\{v_{1,n}^{(0)}\right\}$, $\left\{v_{2,n}^{(0)}\right\}$. Given such a pair of sequences, their coarse approximation (component in V_1) is computed with the lowpass part of the multiwavelet filter bank:

$$\begin{bmatrix} v_{1,n}^{(1)} \\ v_{2,n}^{(1)} \end{bmatrix} = L \begin{bmatrix} v_{1,n}^{(0)} \\ v_{2,n}^{(0)} \end{bmatrix}.$$

Analogously, the details $w_{1,n}^{(1)}$, $w_{2,n}^{(1)}$ in $V_0 \ominus V_1$ are computed with the highpass part $d[k]$.

If the matrix L has eigenvalues $1, \frac{1}{2}, \ldots, \frac{1}{2^{p-1}}$, then polynomials of degree less than p belong to V_0 [7]. This holds for the multiwavelet filter of Table 1 with $p = 2$. All linear functions can be exactly reproduced by integer translates of the scaling functions ϕ_1 and ϕ_2.

A. Symmetric extension

One advantage of a multiwavelet filter bank such as that in Table 1 is the ability to combine orthogonality and linear-phase (symmetry). Linear-phase filter banks are desirable for subband image coders [1, 9] because they enable symmetric extension of finite-length signals at image boundaries. Theories of symmetric extension for 2-band and M-band subband coders have been developed in [9, 2] and elsewhere (see [2]). How do symmetric extension methods work for multiwavelets?

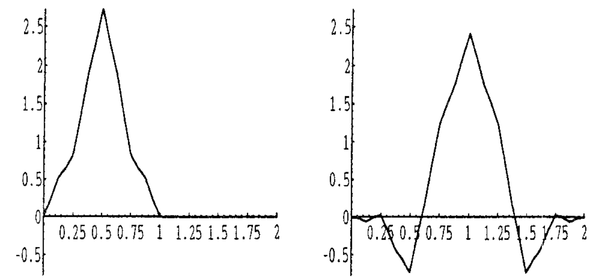

Figure 2: Multiwavelet scaling functions.

Recall the basic problem: given an input signal $f(n)$ with N samples and a linear-phase (symmetric or antisymmetric) filter, how can we symmetrically extend f before filtering and downsampling in a way that preserves the nonexpansive nature of the transform? The possibilities for such an extension have been enumerated in [2]. Depending on the parity of the input signal (even- or odd-length) and the parity and symmetry of the filter, there is a specific non-expansive symmetric extension of both the input signal and the subband outputs. For example, an even-length input signal passed through a symmetric lowpass filter should be extended by repeating the first and last samples if the filter length is even, and without any repeats if the filter length is odd.

We develop non-expansive symmetric extension of signals for multiwavelets. Each row of the multifilter in Table 1 is a linear combination of two filters, one for each input stream. One filter (applied to the first stream) is of even length; the second is of odd length. Thus we should extend the first stream using one method (e.g. repeating the first and last samples) and extend the second stream using another (e.g. *not* repeating samples). Then, when resynthesizing the input signal from the subband outputs, we must symmetrize the subband data differently depending on whether it is going into an even- or odd-length filter.

In particular, suppose we are given two input rows (one of even length, the other of odd length)

$$
\begin{array}{cccccc}
v_{0,1}^{(0)} & v_{1,1}^{(0)} & v_{1,2}^{(0)} & \cdots & v_{1,N-1}^{(0)} & \\
v_{2,0}^{(0)} & v_{2,1}^{(0)} & v_{2,2}^{(0)} & \cdots & v_{2,N-1}^{(0)} & v_{2,N}^{(0)}
\end{array} .
$$

If the rows are symmetrically extended as

$$
\begin{array}{cccccccc}
v_{1,1}^{(0)} & v_{1,0}^{(0)} & v_{1,0}^{(0)} & v_{1,1}^{(0)} & \cdots & v_{1,N-2}^{(0)} & v_{1,N-1}^{(0)} & v_{1,N-1}^{(0)} \\
v_{2,1}^{(0)} & v_{2,0}^{(0)} & v_{2,1}^{(0)} & v_{2,2}^{(0)} & \cdots & v_{2,N-1}^{(0)} & v_{2,N}^{(0)} & v_{2,N-1}^{(0)}
\end{array} \tag{4}
$$

then the subbands also prove to be symmetric:

$$
\begin{array}{ccccccc}
v_{1,1}^{(1)} & v_{1,0}^{(1)} & v_{1,0}^{(1)} & \cdots & v_{1,\frac{N}{2}-2}^{(1)} & v_{1,\frac{N}{2}-1}^{(1)} & v_{1,\frac{N}{2}-1}^{(1)} \\
v_{2,1}^{(1)} & v_{2,0}^{(1)} & v_{2,1}^{(1)} & \cdots & v_{2,\frac{N}{2}-1}^{(1)} & v_{2,\frac{N}{2}}^{(1)} & v_{2,\frac{N}{2}-1}^{(1)}
\end{array}
$$

$$
\begin{array}{ccccccc}
w_{1,1}^{(1)} & w_{1,0}^{(1)} & w_{1,1}^{(1)} & \cdots & w_{1,\frac{N}{2}-1}^{(1)} & w_{1,\frac{N}{2}}^{(1)} & w_{1,\frac{N}{2}-1}^{(1)} \\
-w_{2,1}^{(1)} & 0 & w_{2,1}^{(1)} & \cdots & w_{2,\frac{N}{2}-1}^{(1)} & 0 & -w_{2,\frac{N}{2}-1}^{(1)}
\end{array}
$$

The application of the (linear-phase) synthesis multiwavelet filters now yields the symmetric extension of the original signal. In this way we obtain a nonexpansive transform of finite-length input data which behaves well at the boundaries under quantization.

III. One input stream to two streams

Two data streams enter the multifilter. To create them from an ordinary single-stream input of length N, there are at least three possibilities:

(*i*) Separate odd and even samples (in one dimension), or use adjacent rows of the image (in two dimensions).
(*ii*) Repeat the input stream to produce two length N streams.
(*iii*) Create a consistent approximation based on equation (2) that yields two length $N/2$ streams, and a "de-approximation" that returns a length N stream.

Method (*i*) constrains the design of the multifilter and, in the case of images, introduces nontrivial two-dimensional processing. This method has yet to yield good results. Method (*ii*) is convenient to implement and consistent with the design. Its drawback is the extra calculation — we process two full-length signals before returning to one. Nevertheless, our experiments on one-dimensional signals were encouraging (see Figure 3). Accounting for the oversampled representation, and compressing twice as much, the error was comparable to compression using the $D4$ wavelet [3].

Method (*iii*) maintains a critically sampled representation. The multifilter processes two $N/2$-point data streams $\{v_{1,n}\}$ and $\{v_{2,n}\}$ using an approximation method suggested by Geronimo and described next. It follows the underlying wavelet decomposition and its sampling/interpolation theory. Examination of Figure 2 shows that the multiwavelet scaling function $\phi_1(t)$ is zero at every integer. The scaling function $\phi_2(t)$ is nonzero at $t = 1$ and vanishes at all other integers. Taking the samples of a signal $f[n]$ to be the values of

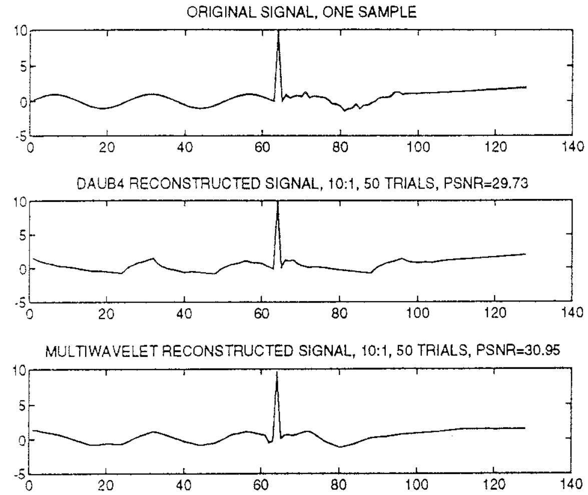

Figure 3: Method (*ii*) applied to 1-D compression.

a continuous-time function $f(t)$ at the half-integers, we can determine the coefficients $v_{k,n}$ from the behavior of the functions $\phi_k(t)$ at the integers and equation (3):

$$
\begin{aligned}
v_{1,n} &= \frac{\phi_2(1)f(2n-1) - \phi_2(1/2)(f(2n)+f(2n-2))}{\phi_2(1)\phi_1(1/2)} \\
v_{2,n} &= \frac{f(2n)}{\phi_2(1)}.
\end{aligned} \tag{5}
$$

To resynthesize the signal on output we invert (5):

$$
\begin{aligned}
f(2n) &= \phi_2(1)v_{2,n} \qquad (6)\\
f(2n-1) &= \phi_2(1/2)(v_{2,n} + v_{2,n-1}) + \phi_1(1/2)v_{1,n}.
\end{aligned}
$$

Given any $f(t) \in V_0$, the approximation (5), filtering, and de-approximation (6) will produce only lowpass output; zero in the highpass subband. For example, $f(t) \equiv 1 \in V_0$ gives $v_{1,n} = 1$ and $v_{2,n} = \sqrt{2}$, which is the eigenvalue 1 eigenvector of the matrix L^T.

If the input signal $f(n)$ has odd length $N = 2m+1$, and we extend it symmetrically as

$$
\ldots f(1),\ f(0),\ f(1),\ f(2), \ldots, f(N-1),\ f(N),\ f(N-1), \ldots
$$

then the approximation (5) yields two rows with the symmetry (4). Thus the approximation/de-approximation method is compatible with symmetric extension for multiwavelet filtering, at least for odd length inputs. If the input signal is even length, we can repeat the last value to make it odd length; this data expansion is negligible for subsequent subband coding.

IV. A Multiwavelet image coder

We now apply the multiwavelets to transform coding of two-dimensional signals. The two-dimensional multiwavelet transform is formed as a tensor product of two one-dimensional transforms. If the forward transform just approximated, then filtered with downsampling, a constant row $f[n] = 1$ would become two half-length rows, one of them the constant 1 and the other the constant $\sqrt{2}$. To overcome this, we *de-approximate* the multiwavelet filter bank output before operating in the vertical direction. Thus the multiwavelet transform works on a one-dimensional signal as in Figure 4 to yield two half-rate outputs. This operation is performed in each dimension, and iterated six times on the lowpass output only [8, 13] to obtain the full multiwavelet decomposition of an image.

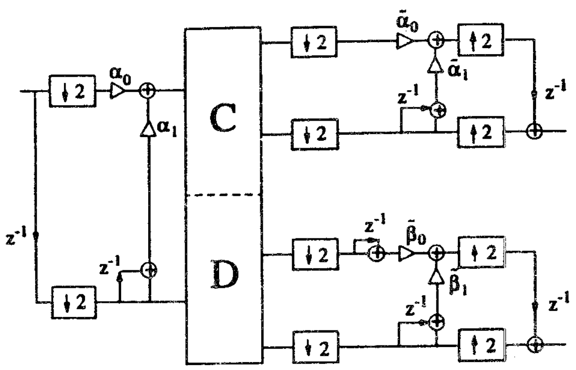

Figure 4: Approximation/de-approximation scheme for computing the 2-dimensional multiwavelet transform.

Wavelets have proven themselves as transforms for use in image coding [8, 13], because of their energy compaction properties and correspondence with models of human vision. We used the 4-coefficient linear-phase multiwavelet filter bank described above in an image coder. After transforming the data with the 4-coefficient linear-phase multiwavelet filter bank described above, we performed entropy-constrained uniform scalar quantization, followed by adaptive Huffman coding [13]. We used the same quantizer-coder combination with a wavelet transform based on the comparable-length Daubechies D4 scalar 2-band wavelets, and applied both coders to a variety of natural images. Results are tabulated in Table 2. The multiwavelet filter bank yielded comparable signal-to-noise ratios, and produced less-objectionable artifacts at edges, even in the interior of the image.

V. Conclusions and Future Directions

We have applied multiwavelets as filter banks to data compression. In doing so, we have developed symmetric extension algorithms for linear-phase multiwavelets, and a new method of applying block filter banks to one- and two-dimensional signal processing based on wavelet sampling theory (approximation and de-approximation). Finally, the multiwavelets compared well with Daubechies wavelets for image coding. Future work will include new designs for multiwavelet filter banks with symmetry and higher orders of approximation, and extensions of the sampling theory approach to multiwavelet filtering.

Image	Ratio	Multiwavelet	Daubechies 4
Lenna	16:1	31.8 dB	32.3 dB
	32:1	29.4 dB	29.3 dB
nuke (NITF7)	16:1	27.1 dB	27.5 dB
	32:1	23.8 dB	24.2 dB

Table 2: Image Coding Results (pSNR)

References

[1] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. IP,* 1 (1992), pp. 205-220.

[2] C. Brislawn, "Classification of symmetric wavelet transforms," Los Alamos Tech. Report, 1993.

[3] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pure Appl. Math.,* 41 (1988), pp. 909-996.

[4] V. M. Gadre and R. K. Patney, "Vector multirate filtering and matrix filter banks," *Proc. IEEE ISCAS*, San Diego, 1992.

[5] J. Geronimo, D. Hardin, and P. R. Massopust, "Fractal functions and wavelet expansions based on several functions," *J. Approx. Th.,* 78 (1994).

[6] R. A. Gopinath, J. E. Odegard, and C. S. Burrus, "Optimal wavelet representation of signals and the wavelet sampling theorem," *IEEE Trans. CAS II,* 41 (1994), pp. 262-277.

[7] C. Heil, G. Strang, and V. Strela, "Approximation by translates of refinable functions," preprint, 1994.

[8] S. Mallat, "A Theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. PAMI,* 11 (1989), pp. 674-693.

[9] M. J. T. Smith and S. Eddins, "Analysis-synthesis techniques for subband image coding," *IEEE Trans. ASSP,* 38 (1990), pp. 1446-1456.

[10] G. Strang and V. Strela, "Short wavelets and matrix dilation equations," *IEEE Trans. SP,* 1995.

[11] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, NJ, 1992.

[12] M. Vetterli and C. Herley, "Wavelets and filter banks," *IEEE Trans. SP*, 40 (1992), pp. 2207-2233.

[13] W. Zettler, J. Huffman, and D. Linden. "The Application of compactly supported wavelets to image compression," *Proc. SPIE*, 1244 (1990), pp. 150-160.