Resolution scalability for arbitrary wavelet transforms in the JPEG-2000 standard

Christopher M. Brislawn, Brendt E. Wohlberg, and Allon G. Percus

Los Alamos National Laboratory, Los Alamos, NM 87545, USA

ABSTRACT

A new set of boundary-handling algorithms has been developed for discrete wavelet transforms in the ISO/IEC JPEG-2000 Still Image Coding Standard. Two polyphase component extrapolation policies are specified: a constant extension policy and a symmetric extension policy. Neither policy requires any computations to generate the extrapolation. The constant extension policy is a low-complexity option that buffers just one sample from each end of the input being extrapolated. The symmetric extension policy has slightly higher memory and conditional-logic requirements but is mathematically equivalent to whole-sample symmetric pre-extension when used with whole-sample symmetric filter banks. Both policies can be employed with arbitrary lifted filter banks, and both policies preserve resolution scalability and reversibility. These extension policies will appear in Annex H, "Transformation of images using arbitrary wavelet transformations," in Part 2 ("Extensions") of the JPEG-2000 standard.

Keywords: JPEG 2000, wavelet transform, lifting, filter bank, boundary conditions, image coding, resolution scalability

1. INTRODUCTION

One of the main technical advances in the recently released JPEG-2000 Still Image Coding Standard[1,2] is the use of discrete wavelet transforms,[3–7] which are performed by two-channel perfect reconstruction multirate filter banks like the one shown in Figure 1. In conjunction with JPEG-2000's embedded, packetized codestream, wavelet transforms allow resolution-progressive decoding of a compressed image codestream; i.e., an image can be decoded at any level of resolution created in the encoding process by decoding only the packets containing wavelet coefficients up to the desired level of resolution. Progressively reconstructing the image at the next higher level of spatial resolution only requires decoding the packets containing the wavelet coefficients representing the next resolution level and synthesizing them with the previously decoded coefficients. This is a significant improvement in scalability over the original JPEG standard.[8,9]

Resolution-progressive decoding is complicated, however, by the presence of tiling options in JPEG-2000. If tiling is used, wavelet transforms are performed independently on each tile, enabling each tile to be decoded independently but also creating the potential for tiling artifacts. Tiling is administered in JPEG-2000 through a global reference grid coordinate system, as depicted in Figure 2. The term *resolution scalability* is defined as the ability to decode reduced-resolution versions of an image simply by decoding the lowpass coefficients from the desired resolution level in each tile; implicit in the concept of resolution scalability is the absence of any phase distortion at tile boundaries attributable to reduced-resolution decoding.

E-mail: `brislawn@lanl.gov`, `brendt@lanl.gov`, `percus@lanl.gov`. Los Alamos National Laboratory is operated by the University of California for the U. S. Department of Energy under contract W-7405-ENG-36. This work was supported in part by the DOE Applied Mathematical Sciences Program, contract KC-07-01-01. Report #LAUR-03-2299. To appear in *Visual Commun. & Image Process.*, Proc. SPIE (Lugano, Switzerland), July 2003. This preprint is distributed with permission of SPIE. One print or electronic copy may be made for personal use only. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper are prohibited. © 2003 Society of Photo-Optical Instrumentation Engineers.

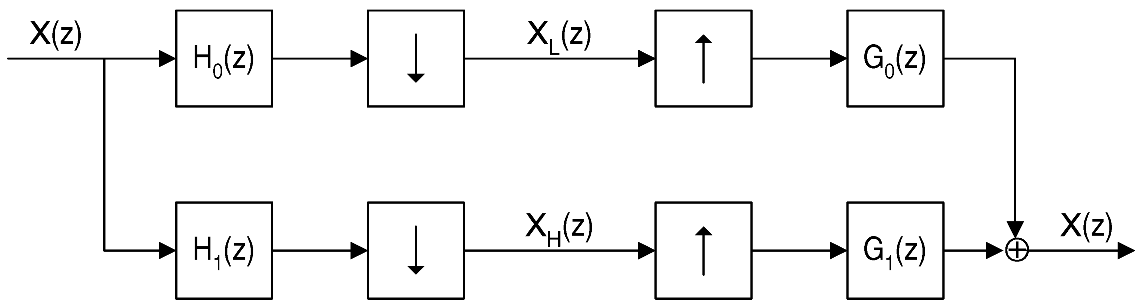

Figure 1. Two-channel perfect reconstruction multirate filter bank.

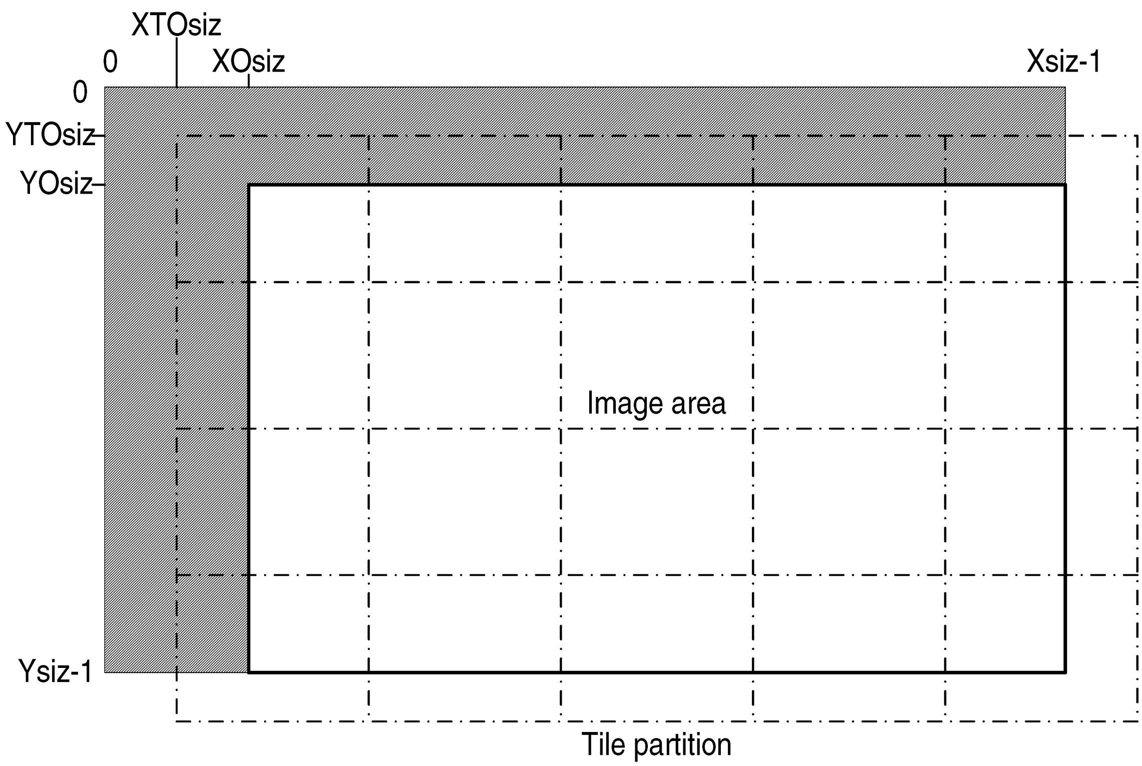

Figure 2. Reference grid coordinate system with a tile partition.

We begin by discussing resolution scalability on the reference grid. We then examine the problem in one dimension in the context of direct-form implementation of linear phase filter banks using symmetric extension techniques,[10] an approach popular in image coding applications. The standard symmetric extension technique for whole-sample symmetric (i.e., odd-length linear phase FIR) filter banks is shown to provide resolution scalability, while the standard symmetric extension approach for half-sample symmetric (i.e., even-length linear phase FIR) filter banks fails to do so. One potential solution to this problem is discussed, along with the reason why it was not incorporated into the JPEG-2000 standard. Finally, we look at the problem from the perspective of lifting factorizations of the polyphase matrix representation of a filter bank. This leads to a better boundary-handling method that actually accommodates *arbitrary* FIR filter banks; this is the method adopted in Part 2, “Extensions,” of the JPEG-2000 standard.[11]

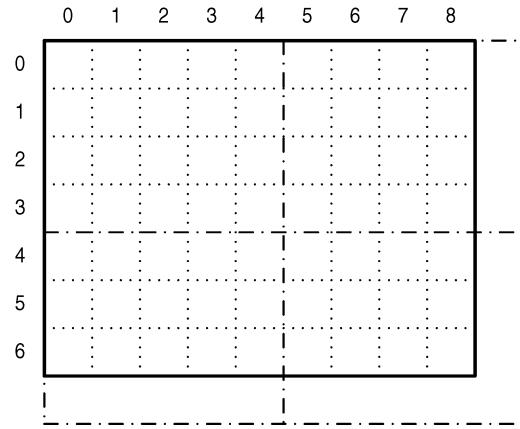

Figure 3. Reference grid with pixel indices divided into four tiles.

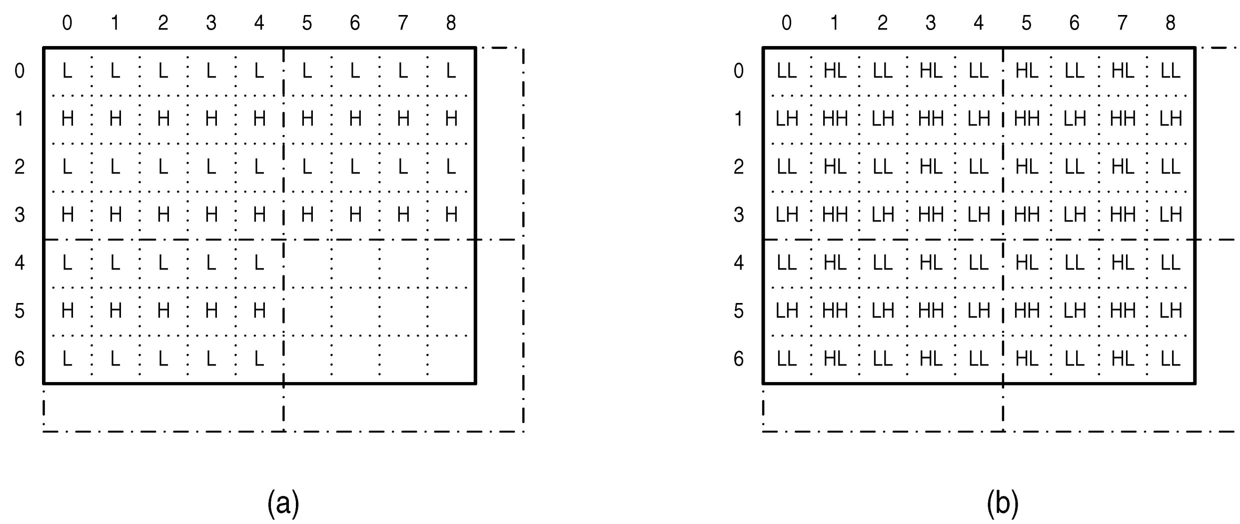

Figure 4. (a) Interleaved data after vertical filtering of the first three tiles. (b) Interleaved first-level subbands after both vertical and horizontal filtering.

2. RESOLUTION SCALABILITY ON THE REFERENCE GRID

Figure 3 depicts an image indexed with reference grid coordinates and partitioned into four tiles. Each small cell in Figure 3 represents a single pixel; the tiles measure 5 pixels wide by 4 pixels high, and the tiles along the bottom and right edges of the image are incomplete. The goal of resolution scalable image coding is to enable the extraction of reduced-resolution partial reconstructions irrespective of possible tiling partitions. Towards this end, JPEG-2000 defines a mapping of wavelet transform subband coefficients onto the reference coordinate system that codifies the concept of lowpass subbands as reduced-resolution versions of the image. An intermediate stage of the first level of wavelet transform decomposition is shown in Figure 4(a). The vertical (column-wise) filtering of the first 3 tiles is shown interleaved onto the reference grid, beginning with lowpass samples in row 0. Note that the alternation of lowpass and highpass outputs is preserved across the horizontal tile boundary, and that lowpass samples have even vertical coordinate indices (row numbers) while highpass samples have odd vertical indices.

Figure 4(b) shows the result of completing one level of two-dimensional decomposition. The second letter in each pixel cell indicates which vertically filtered subband (from Figure 4(a)) the sample is in, while the first letter indicates which horizontally filtered subband a sample is in. For instance, a sample labelled "HL" has been highpass filtered in the horizontal direction and lowpass filtered in the vertical direction. As with Figure 4(a), the alternation of lowpass and highpass row filtered samples is preserved across the vertical tile boundary in Figure 4(b). Samples that have been lowpass-filtered in the horizontal direction have even horizontal indices, and samples that have been highpass-filtered in the horizontal direction have odd horizontal indices. As shown in Figure 4(b), the two-dimensional wavelet transform decomposition is *nonexpansive*; i.e., the number of output subband coefficients for a given tile is exactly equal to the number of input samples in that tile.

To extract a reduced-resolution version of the image at one-half the original resolution, a user only needs to reconstruct the "LL" samples from each tile and composite them as shown. If both image dimensions are even, the convention for alternating lowpass- and highpass-filtered samples ensures that the reduced-resolution image will have exactly one-fourth as many pixels as the original. The key technical problem is to define invertible, nonexpansive wavelet transform algorithms for tiles of arbitrary sizes and offsets while ensuring that the reduced-resolution reconstructions are free of phase distortion across tile boundaries. The possibility of such distortion in reduced-resolution reconstructions is a separate issue from concerns about tiling artifacts in quantized full-resolution reconstructions: a losslessly encoded image will be perfect when reconstructed to full resolution, yet reduced-resolution reconstructions may still show phase distortion at tile boundaries if phase continuity is not maintained across tile boundaries in the LL subbands.

3. DIRECT-FORM ANALYSIS

From this point on, we limit discussion to one-dimensional filter banks that can be used in product filtering schemes like the ones used in image coding. The construction of (one-dimensional) symmetric extension boundary-handling methods that result in nonexpansive transforms was studied in detail by Brislawn.[10] Such transforms are referred to here as symmetric *pre*-extension transforms to emphasize the fact that they are defined via preprocessing operations that extend a finite-length input vector prior to filtering it. Specifically, the symmetric pre-extension method used with whole-sample symmetric (WS) filter banks involves extending an input vector, $x(n)$, $0 \le n < N_0$, to an infinite, whole-sample symmetric signal of period $N = 2N_0 - 2$, as shown in Figure 5. Symmetric pre-extension for half-sample symmetric (HS) filter banks uses the half-sample symmetric extension of period $N = 2N_0$ also shown in Figure 5. When the appropriate linear phase extension is passed through a linear phase filter bank like the one shown in Figure 1, the decimated subbands X_L and X_H are also periodic linear phase signals with just N_0 degrees of freedom; i.e., both X_L and X_H can be reproduced exactly from a total of just N_0 subband samples, even if N_0 is odd.

It is known[12] that the only "nontrivial" classes of two-channel, linear phase, finite impulse response perfect reconstruction filter banks are the WS and HS classes mentioned above. While nonexpansive symmetric pre-extension transforms have been derived for both of these classes,[10] the analysis in that paper did not consider the issue of resolution scalability in tiled images, so we now take a closer look at what happens in these two cases.

3.1. Whole-sample symmetric filter banks.

We illustrate the issues using a simple example of a (non-normalized) WS filter bank:

$$H_0(z) = z + 2 + z^{-1} \quad ; \quad H_1(z) = z \quad . \tag{1}$$

The group delays (which coincide with the axes of symmetry) of these filters are $\gamma_0 = 0$, $\gamma_1 = -1$; in general, perfect reconstruction is only possible for WS filter banks if $\gamma_0 + \gamma_1$ is odd. Figure 6(a) shows the time-reversed and translated lowpass impulse responses that generate the lowpass subband from a tiled one-dimensional signal; e.g., a row vector from Figure 3. Both tiles 1 and 2 have been filtered (independently)

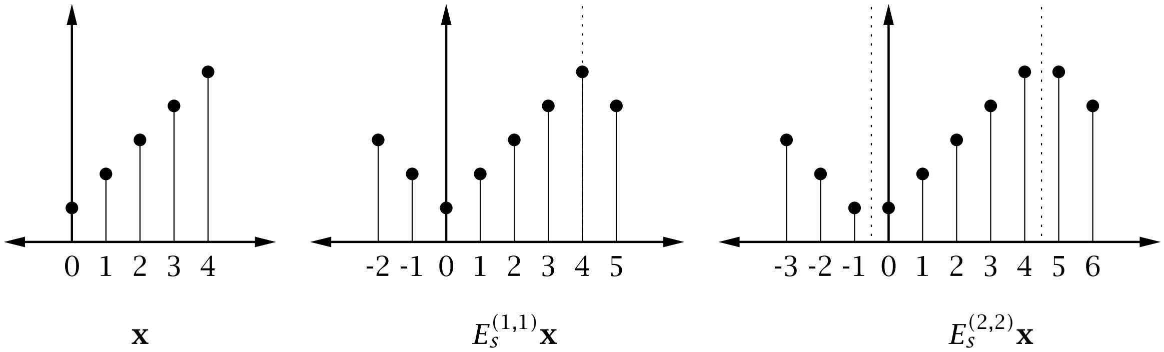

Figure 5. A finite-length signal and single periods of its whole-sample symmetric extension ($E_s^{(1,1)}\mathbf{x}$) and its half-sample symmetric extension ($E_s^{(2,2)}\mathbf{x}$).

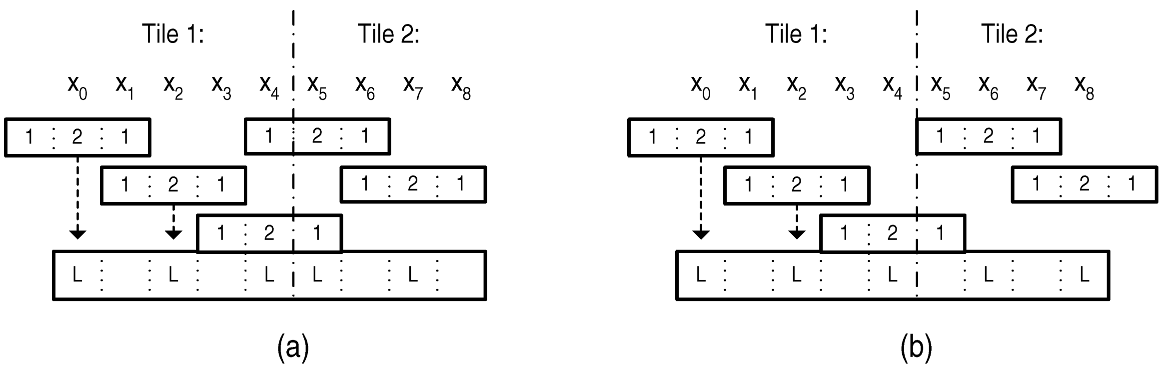

Figure 6. (a) Failure of resolution scalability. (b) Correct phasing of lowpass filters in tiles 1 and 2.

using the zero-phase lowpass filter from (1), and the lowpass subbands have been composited onto a global coordinate grid at the bottom of Figure 6. (Boundary-handling details are ignored in Figure 6). Because tile 1 has an odd-length input, the last subband sample from tile 1 and the first from tile 2 are both lowpass samples, resulting in a disruption of the period-2 alternation of lowpass samples on the reference grid. In a visualization application, displaying the LL subband from this decomposition would reveal artifacts similar to what one would see if a single column of pixels were deleted from the middle of an image. Such artifacts would be particularly noticeable if the discontinuity were crossed by sharp diagonal lines or edges.

The solution to this problem is illustrated in Figure 6(b), and the input samples in the tiles have been offset so we can display the whole-sample symmetric extension of the input signals in both tiles. Observe how the lowpass impulse responses in tile 2 are phased so as to maintain period-2 alternation of lowpass samples across the tile boundary. Achieving this effect has a significant consequence: specifically, tile 2 is processed by a lowpass filter having group delay $\gamma_0 = -1$ *with respect to the first sample in tile 2*, as can be seen in Figure 6(b). The corresponding highpass filter can then be given a group delay of $\gamma_1 = 0$ (again, with respect to the first sample in tile 2). In effect, when we think of transforms as being applied independently to distinct tiles, tile 2 is being transformed by the filter bank

$$H_0(z) = z^2 + 2z + 1 \quad ; \quad H_1(z) = 1 \quad , \tag{2}$$

which is a *different* symmetric pre-extension transform than the one given by (1). The complete transform for both tiles, with both lowpass and highpass subbands, is shown in Figure 7. Note that the first subband sample on the reference grid in tile 2 is a highpass output; for this reason, the combination of (2) together

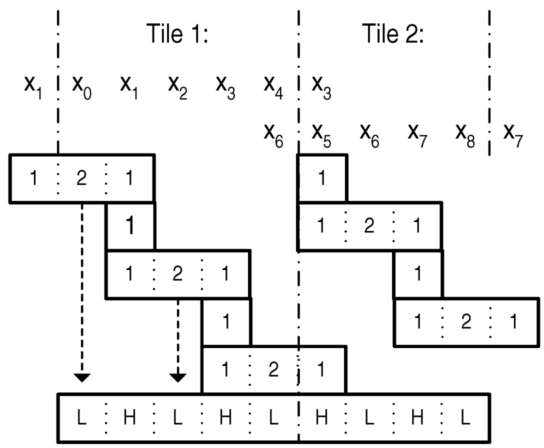

Figure 7. Resolution scalable implementation of whole-sample symmetric filter banks with symmetric extension.

with the interleaving convention of the reference grid coordinate system is referred to as a *highpass-first* transform. The nonexpansiveness and invertibility of both transforms is derived in detail by Brislawn.[10]

We should point out that this is *not* the way that resolution-scalable wavelet transforms are specified in the JPEG-2000 standard. The standard document[1] specifies a single filtering operation (equivalent to (1)) whose output is solely a function of the reference grid coordinates of each sample, an approach that maintains period-2 alternation of interleaved subbands across tile boundaries. At tile boundaries, the input signal is redefined by symmetric extension to achieve the effect of independently transformed tiles. The net effect is mathematically equivalent to the combination of transforms depicted in Figure 7.

3.2. Half-sample symmetric filter banks.

For this case, we use the familiar example of the (non-normalized) Haar filter bank:

$$H_0(z) = z + 1 \quad ; \quad H_1(z) = z - 1 \quad . \tag{3}$$

The group delays of these filters are equal: $\gamma_0 = -1/2 = \gamma_1$. Let's try the same trick that was used in Figure 7 to maintain resolution scalability. Change the group delays of the filters used on tile 2 to $\gamma_0 = +1/2 = \gamma_1$; the result is shown in Figure 8. Note the use of half-sample symmetric extension at tile boundaries; because of this, the first highpass output in tile 2 is identically 0. This implies that the transformation of tile 2 is not invertible since it maps 4 inputs to 3 transformed samples!

According to Brislawn,[10] the correct way to get an invertible symmetric pre-extension transform for HS filter banks with $\gamma_0 = +1/2 = \gamma_1$ on an even-length input is to compute *lowpass* coefficients at both ends of the signal, as shown in Figure 9(a). (The corresponding highpass coefficients, if we were to compute them, would both be zero.) This transform, while invertible, has 2 more lowpass than highpass coefficients, even though the input length is even. This violates the requirements of resolution scalability, and for this reason the JPEG-2000 committee initially thought that HS filter banks were unsuitable for inclusion in the JPEG-2000 standard.

There is an extremely simple solution, however, which is depicted in Figure 9(b) and which we refer to as the "two-point transform" approach. It merely involves computing the difference of the two lowpass samples at the left end of tile 2 in order to generate an additional highpass sample. The final decomposition is resolution scalable, and the two-point difference operation (which can be used in conjunction with any HS filter bank, not just the Haar) is invertible in either floating point or integer arithmetic. Perceptual evaluations of the two-point transform method[13] showed that it produced image quality on tiled images that was comparable to the image quality of whole-sample symmetric pre-extension transforms when employed

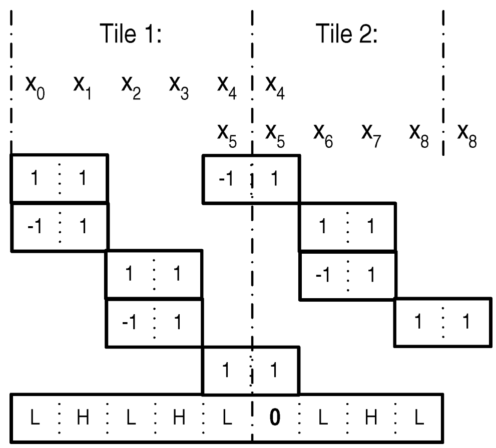

Figure 8. Noninvertible attempt to achieve resolution scalability for half-sample symmetric filter banks with symmetric extension.

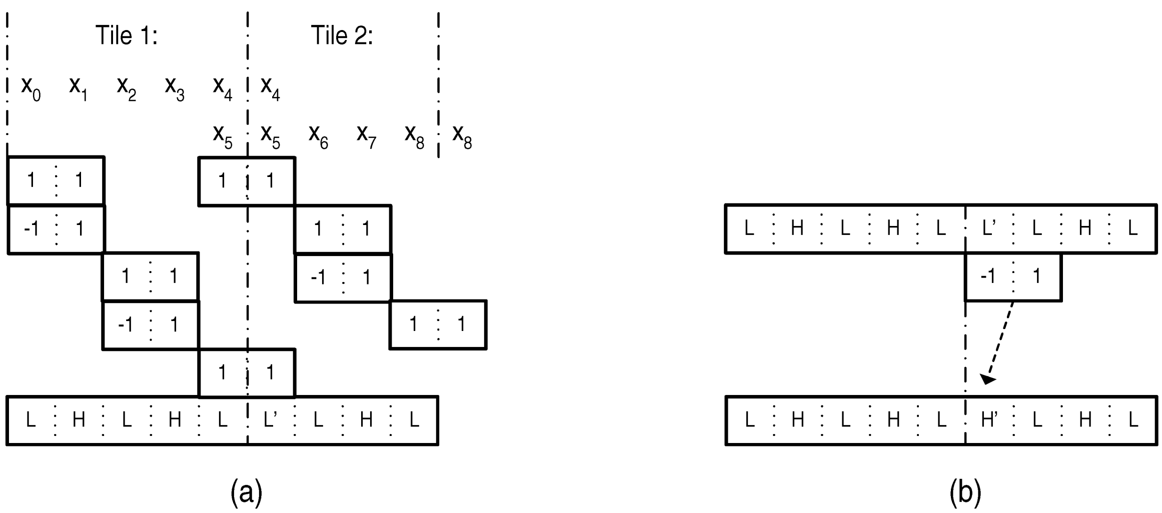

Figure 9. (a) Invertible nonexpansive transforms for half-sample symmetric filter banks. (b) The 2-point transform technique for achieving resolution scalability.

with filters of similar image-coding performance (e.g., when comparing the 2-10 HS filter bank to the 9-7 WS filter bank).

We thought this would solve the problem of adding HS filter banks to Part 2 of the JPEG-2000 standard,[11] but in the course of solving the resolution scalability problem we discovered two additional problems with symmetric pre-extension transforms for HS filter banks that have nothing to do with resolution scalability. Since both of these problems are related to lifted implementations of HS filter banks, we now turn to the polyphase representation and lifting factorizations.

4. THE POLYPHASE REPRESENTATION AND LIFTING

The JPEG-2000 standard uses a variant form of the polyphase representation based on the work of Daubechies and Sweldens.[14] The input signal is demultiplexed into even and odd components,

$$x_i(n) = x(2n+i) \quad , \quad i = 0, 1 \quad . \tag{4}$$

The *polyphase vector form* of a signal is defined to be

$$\mathbf{x}(n) = \begin{bmatrix} x_0(n) \\ x_1(n) \end{bmatrix} \tag{5}$$

in the time domain and

$$\mathbf{X}(z) = \begin{bmatrix} X_0(z) \\ X_1(z) \end{bmatrix} \tag{6}$$

in the transform domain. The polyphase components of the analysis filter bank are defined as

$$h_{a_{ij}}(n) = h_i(2n - j) \quad , \quad i, j = 0, 1 \quad , \tag{7}$$

which implies

$$H_i(z) = H_{a_{i0}}(z^2) + zH_{a_{i1}}(z^2) \quad , \quad i = 0, 1 \quad . \tag{8}$$

With these definitions, the analysis bank in Figure 1 can be written

$$\begin{bmatrix} X_L(z) \\ X_H(z) \end{bmatrix} = \mathbf{H}_a(z)\mathbf{X}(z) \quad , \tag{9}$$

where $\mathbf{H}_a(z)$ is the transform (polyphase) matrix of the impulse responses given by (7).

A *lifting factorization* of the filter bank is a factorization of $\mathbf{H}_a(z)$ into a matrix product,

$$\mathbf{H}_a(z) = \mathrm{diag}(1/K, K)\mathbf{S}_N(z) \cdots \mathbf{S}_1(z)\mathbf{S}_0(z) \quad , \tag{10}$$

where each factor, $\mathbf{S}_i(z)$, is either lower or upper triangular, with ones on the diagonal; e.g.,

$$\mathbf{S}_0(z) = \begin{bmatrix} 1 & S_0(z) \\ 0 & 1 \end{bmatrix} \quad . \tag{11}$$

The above factor (11) "lifts" (or updates) the lowpass (even-indexed) subsequence, $x_0(n)$, while leaving the highpass (odd-indexed) subsequence, $x_1(n)$, unchanged. A schematic diagram of a lifting factorization is depicted in Figure 10 for both analysis and synthesis. Figure 10 shows a situation in which a finite-length vector is being transformed in the analysis bank by a symmetric pre-extension transform, for which the box labelled "Extend" is either the $E_s^{(1,1)}$ operator (for WS filter banks) or the $E_s^{(2,2)}$ operator (for HS filter banks) depicted in Figure 5. The "Extend" operators extending the input vectors to the synthesis bank are chosen to be the appropriate extensions that restore the linear phase characteristics of the subbands; see Brislawn[10] for details.

This is where the difficulties resume. One of the significant features of the JPEG-2000 standard is the ability to perform lossless transform coding using so-called "reversible" filter banks. A filter bank is reversible if the impulse response coefficients in its lifting steps are dyadic rationals. In this case the filter bank can be implemented losslessly by applying a rounding operation immediately following each lifting filter, $S_i(z)$, in Figure 10 before accumulating the update onto the channel being lifted. Dyadic rational coefficients allow the divide-by-2-and-round operations to be performed in integer arithmetic using bit-shifts. If the filter bank is half-sample symmetric then the highpass filter, $H_1(z)$, is actually antisymmetric, as is the highpass filtered channel, $X_H(z)$. The symmetric pre-extension approach depends critically on knowing the exact symmetry properties of the subbands so that these symmetries can be used to restore (extrapolate) the finite-length subband vectors going into the synthesis bank, as mentioned in the preceding paragraph. Unfortunately, we have shown[15] that there exist lifting factorizations of reversible HS filter banks for which *no* rounding rule preserves the antisymmetry of the highpass channel. In other words, the transform is still reversible but the highpass subband no longer possesses the antisymmetry required by the symmetric pre-extension method. This pathology occurs for some, but not all, HS lifting factorizations.

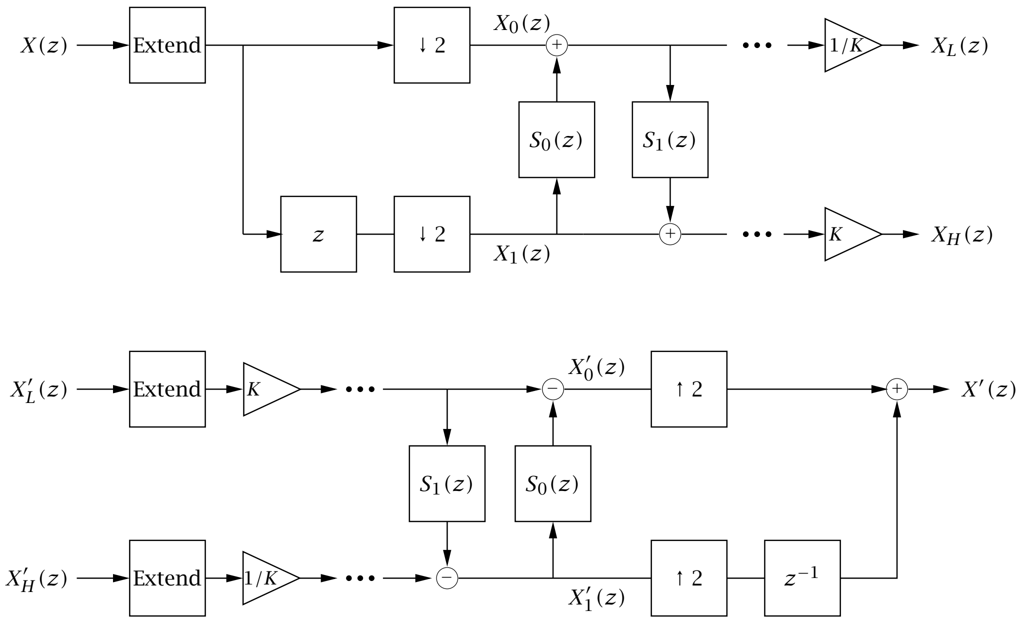

Figure 10. Lifting factorization of a filter bank with symmetric pre-extension.

5. ITERATED POLYPHASE COMPONENT EXTENSIONS

In the course of studying this problem, we considered an alternative to symmetric pre-extension in which extension operations are not performed until immediately before each lifting update, as shown in Figure 11. We call this method *iterated polyphase component extension.* For WS filter banks, it was known[2] that polyphase component extension operators could be chosen so as to make the iterated extension scheme in Figure 11 mathematically equivalent to symmetric pre-extension transforms using the $E_s^{(1,1)}$ extension. This requires the use of linear phase lifting steps in the lifting factorization of the WS filter bank; details and derivations can be found in Brislawn and Wohlberg.[16]

One observation is immediately clear from Figure 5: there is no hope of defining an iterated extension scheme for HS filter banks that will be mathematically equivalent to symmetric pre-extension using $E_s^{(2,2)}$, even for non-reversible HS filter banks. Examine the extrapolation to the left of 0 in the picture of $E_s^{(2,2)}\mathbf{x}$: the odd-indexed, extrapolated samples are equal to *even*-indexed samples in the input vector; e.g., $E_s^{(2,2)}\mathbf{x}(-1) = \mathbf{x}(0)$, $E_s^{(2,2)}\mathbf{x}(-3) = \mathbf{x}(2)$, etc. Similarly, the even-indexed, extrapolated samples are equal to *odd*-indexed samples in the input vector. This means that the polyphase components of a half-sample symmetrically pre-extended input vector are not symmetric but, rather, are time-reversed (mirrored) images of one another. Unlike the case of whole-sample symmetric pre-extension, *half-sample symmetric pre-extension cannot be achieved by applying iterated extension operations to polyphase component vectors as indicated in Figure 11.*

Since there was no chance of performing reversible half-sample symmetric pre-extension or anything equivalent to it using iterated polyphase component extension, it was decided to drop the notion that HS filter banks should be implemented in JPEG-2000 via symmetric extension. In retrospect, this was an extremely liberating decision because *any* extension method will work with *any* type of lifting steps in an iterated polyphase component extension scheme, provided the same extension operations are used in both the analysis and synthesis filter banks. (This includes lossless implementation of reversible filter banks.) Since there was no longer a special boundary-handling procedure just to support HS filter banks, it became possible to expand the universe of admissible filter banks in JPEG-2000 Part 2 to include *all* finite impulse response two-channel filter banks. It is known[14] that all such filter banks have lifting factorizations, and

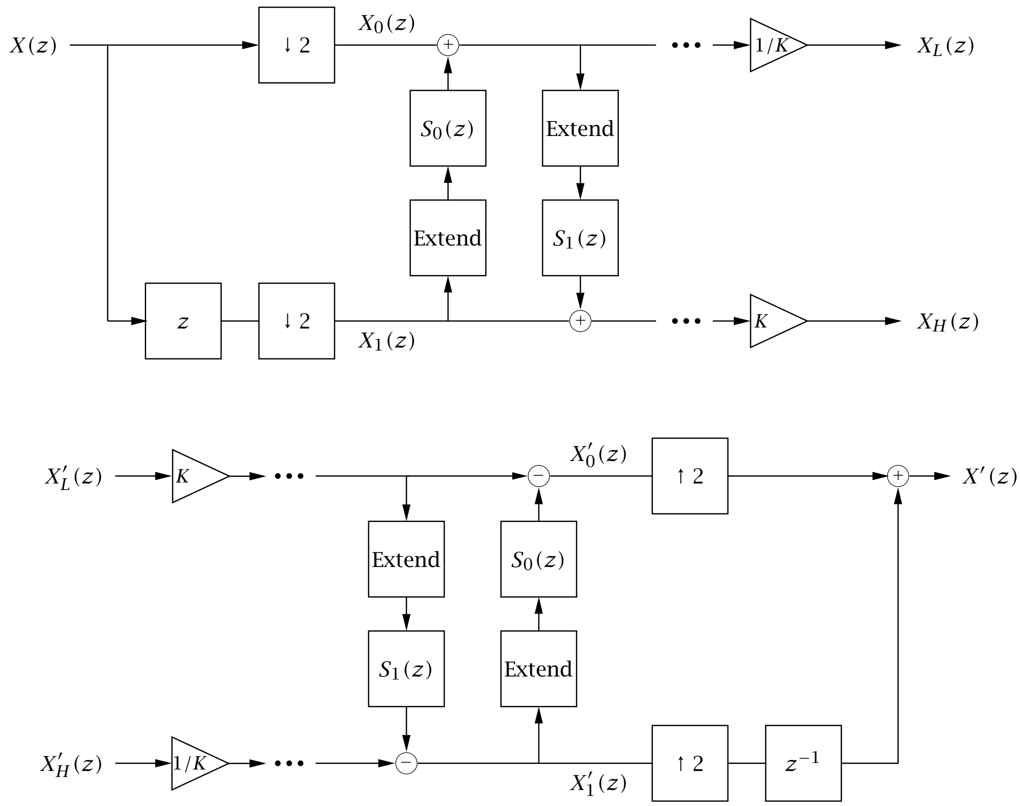

Figure 11. Lifting factorization of a filter bank with iterated polyphase component extension.

this expanded universe includes other interesting classes of filter banks that had previously been excluded from consideration, such as paraunitary (orthogonal) filter banks.

The only remaining task was to decide which iterated polyphase component extension methods should be supported in Part 2 of the standard. In the interest of backwards compatibility, an obvious candidate was the extension rule that furnishes a transform equivalent to whole-sample symmetric pre-extension when used in conjunction with a linear phase lifting factorization of a WS filter bank. This extension rule is most easily described in terms of interleaved polyphase components because the linear phase lifting steps for a WS filter bank have the property that they are, themselves, WS filter banks, and WS filter banks preserve whole-sample symmetry. This means that if a whole-sample symmetric signal is input to a WS filter bank then the *interleaved* output subbands will also form a whole-sample symmetric signal. Thus, the correct extension to apply at the input to each lifting step in a WS filter bank is the extension that will lead to whole-sample symmetry for the interleaved polyphase components at that point in the lifting cascade. (It isn't actually necessary to form the whole-sample symmetric extension of the interleaved polyphase components prior to each lifting step because only one of the two components is actually being filtered and therefore actually needs to be extended.) This extension scheme is denoted the "WS" iterated extension method in JPEG-2000 Part 2, although it can be used with any filter bank (not just WS filter banks).

In the interest of providing a low-complexity alternative, JPEG-2000 Part 2 also allows for the use of a "constant" ("CON") iterated extension scheme. This is conceptually simpler than the whole-sample symmetric scheme and requires almost no memory to implement. In the constant extension scheme, the boundary samples of the input polyphase component vector to a lifting filter are simply replicated as far to the left (or to the right) as necessary to compute the needed lifting filter outputs. Thus, only one cache register is needed for the left (resp., for the right) end of the input vector. In perceptual tests on natural imagery,[17] the constant extension scheme used with both WS and HS filter banks was found to perform subjectively on a par with WS iterated extension (no statistically significant differences in viewer preference) at rates of 0.5 bits per pixel. Statistically significant differences in viewer preference began to appear at rates of 0.25 bpp, and we expect that WS extension will generally outperform CON extension at extremely low rates, but the CON extension appears to be an attractive alternative for low-complexity implementations operating at moderate to high rates.

6. CONCLUSIONS

The boundary handling options incorporated into Part 2 of the ISO/IEC JPEG-2000 Still Image Coding Standard have been described and compared. Extension schemes are specified in Part 2 in terms of iterated polyphase domain extensions, in contrast to the symmetric pre-extension approach followed in the JPEG-2000 baseline (Part 1). Part 2 contains two extension options that can be used with lifting factorizations of completely arbitrary two-channel FIR filter banks. Both extension options support resolution scalability and both support lossless transforms when used with reversible filter banks. The WS extension scheme is mathematically equivalent to whole-sample symmetric pre-extension when used with linear phase lifting factorizations of WS filter banks; it is backwards-compatible with JPEG-2000 Part 1 when employed with either of the Part 1 filter banks. The CON extension is a low-memory alternative to WS iterated extension that appears to be perceptually as good as WS extension at rates of 0.5 bpp or higher.

REFERENCES

1. *JPEG 2000 Image Coding System, Part 1*, ISO/IEC Int'l. Standard 15444-1, ITU-T Rec. T.800, Int'l. Org. for Standardization, Dec. 2000.
2. D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, Kluwer, Boston, MA, 2002.
3. I. C. Daubechies, *Ten Lectures on Wavelets*, no. 61 in CBMS-NSF Regional Conf. Series in Appl. Math., (Univ. Mass.—Lowell, June 1990), Soc. Indust. Appl. Math., Philadelphia, 1992.
4. P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, Englewood Cliffs, NJ, 1993.
5. M. Vetterli and J. Kovačević, *Wavelets and Subband Coding*, Prentice Hall, Englewood Cliffs, NJ, 1995.
6. G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge, Wellesley, MA, 1996.
7. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 2nd ed., 1999.
8. *Digital Compression and Coding of Continuous-Tone Still Images, Part 1: Requirements and Guidelines*, ISO/IEC Int'l. Standard 10918-1, Int'l. Org. for Standardization, 1994.
9. W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*, Van Nostrand Reinhold, New York, NY, 1992.
10. C. M. Brislawn, "Classification of nonexpansive symmetric extension transforms for multirate filter banks," *Appl. Comput. Harmonic Anal.* **3**, pp. 337–357, 1996.
11. *JPEG 2000 Image Coding System, Part 2 (Extensions)*, ISO/IEC Int'l. Standard 15444-2, ITU-T Rec. T.801, Int'l. Org. for Standardization, Dec. 2001.
12. T. Q. Nguyen and P. P. Vaidyanathan, "Two-channel perfect-reconstruction FIR QMF structures which yield linear-phase analysis and synthesis filters," *IEEE Trans. Acoust., Speech, Signal Process.* **37**, pp. 676–690, May 1989.
13. C. M. Brislawn, B. E. Wohlberg, and A. G. Percus, "Even-length filter bank options: Report on core experiment CodEff03," Tech. Rep. WG1N1911, ISO/IEC Standards Committee JTC1/SC29/WG1, New Orleans, LA, Dec. 2000.
14. I. C. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.* **4**(3), pp. 245–267, 1998.
15. B. E. Wohlberg and C. M. Brislawn, "Reversible integer-to-integer transforms and symmetric extension of even-length filter banks," in *Visual Commun. & Image Proc.*, Proc. SPIE, SPIE, (Lugano, Switzerland), July 2003.

16. C. M. Brislawn and B. E. Wohlberg, "Resolution scalability for multirate filter banks," tech. rep., Los Alamos Nat'l. Lab, 2003. preprint.
17. C. M. Brislawn, S. M. Mniszewski, M. D. Pal, A. G. Percus, B. E. Wohlberg, T. Acharya, P.-S. Tsai, and M. Lepley, "Report on core experiment CodEff03: Even-length filter bank option," Tech. Rep. WG1N2209, ISO/IEC Standards Committee JTC1/SC29/WG1, Stockholm, Sweden, July 2001.