Balanced Multiresolution for Symmetric/Antisymmetric Filters

Mahmudul Hasan, Faramarz F. Samavati, Mario C. Sousa

Department of Computer Science, University of Calgary, Alberta, Canada
{mhasan, samavati, smcosta}@ucalgary.ca

Abstract

Given a set of symmetric/antisymmetric filter vectors containing only regular multiresolution filters, the method we present in this article can establish a *balanced multiresolution* scheme for images, allowing their *balanced decomposition* and subsequent perfect reconstruction without the use of any extraordinary boundary filters. We define balanced multiresolution such that it allows balanced decomposition i.e. decomposition of a high-resolution image into a low-resolution image and corresponding *details* of equal size. Such a balanced decomposition makes on-demand reconstruction of *regions of interest* efficient in both computational load and implementation aspects. We find this balanced decomposition and perfect reconstruction based on an appropriate combination of symmetric/antisymmetric extensions near the image and detail boundaries. In our method, exploiting such extensions correlates to performing sample (pixel/voxel) split operations. Our general approach is demonstrated for some commonly used symmetric/antisymmetric multiresolution filters. We also show the application of such a balanced multiresolution scheme in real-time focus+context visualization.

Keywords: multiresolution, reverse subdivision, balanced decomposition, perfect reconstruction, lossless reconstruction, focus+context visualization, contextual close-up, symmetric extension, antisymmetric extension

1. Introduction

Applications that facilitate multiscale 2D and 3D image visualization and exploration (see [LHJ99, WS05], for example) benefit from multiresolution schemes that decompose high-resolution images into low-resolution approximations and corresponding *details* (usually, wavelet coefficients). Several subsequent applications of such a decomposition constructs the corresponding *wavelet transform*. This wavelet transform can then be used to derive low-resolution approximations of the entire image, as well as high-resolution approximations of a *region of interest* (ROI), on demand. Reconstructing the high-resolution approximation of a ROI involves locating the corresponding details from a hierarchy of details within the wavelet transform. One such hierarchy of details resulting from only two levels of decomposition of a 512×256 Blue Marble image [VEN02] is shown in Figure 1.

For the purpose of demonstration, we created the wavelet transform in Figure 1 using the *short* filters of quadratic B-spline presented by Samavati *et al.* [SB04, SBO07]. In practice, images that require multiscale visualization are larger in size and may require more levels of decomposition. For each level of decomposition in this particular example, the image was first decomposed heightwise and then widthwise. The rectangles that contain details are numbered in the order of their creation during the two levels of decomposition.

Given such a hierarchy of details, if we need to reconstruct the high-resolution approximation of a ROI located in the low-resolution (coarse) image shown in the top-left rectangle in Figure 1, we have to locate the corresponding details in some or all of the rectangles that contain details (numbered $1.1, \ldots, 2.2$) depending on the expected level of resolution. Locating these details will be straightforward if each of the heightwise and widthwise decomposition steps decomposes an image into two halves of equal size – one half corresponding to the coarse image and the other half corresponding to the details. Among B-spline wavelets, only the filters obtained from Haar wavelets provide such a balance in decomposition [Haa10, SDS96]. However, because Haar wavelets and the associated scaling functions are not continuous, it would be beneficial to achieve such a balanced decomposition for the filters obtained from higher order scaling functions and their wavelets.

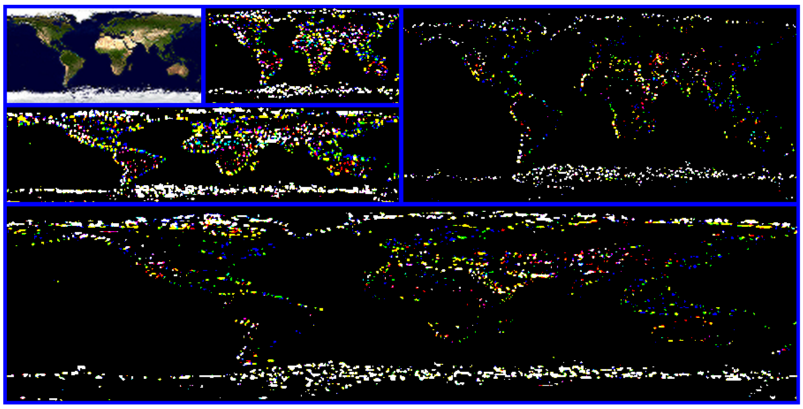


Figure 1: Hierarchy of details in a wavelet transform resulting from two levels of decomposition of a 512×256 Blue Marble image. The size of the image in this figure is not representative of its original size.

There exist many sets of local regular multiresolution filters obtained from higher order scaling functions and their wavelets (see [SBO07, CDF92, Dau92, Mey90], for example). They require special treatments to handle image and detail boundaries (such as the use of extraordinary boundary filters) and result in unequal numbers of coarse and detail samples after decomposition. Therefore, in this article, we present a method to devise balanced multiresolution schemes for such filters, using an appropriate combination of symmetric/antisymmetric extensions near the image and detail boundaries, which correlate to sample split operations. To guarantee a perfect (lossless) reconstruction without the use of any extraordinary boundary filters, our method requires each of the given decomposition and reconstruction filter vectors (kernels) to be either symmetric or antisymmetric about their centers. Many existing sets of local regular multiresolution filters, such as those associated with the B-spline wavelets [SBO07], biorthogonal and reverse biorthogonal wavelets [CDF92, Dau92], and Meyer wavelets [Mey90, Dau92], exhibit such symmetric/antisymmetric structures. Additionally, we show the use of such a balanced multiresolution scheme in an application that exploits real-time focus+context visualization.

This article is organized as follows. In section 2, we present the notations used throughout the article. Next, we formulate the problem definition in section 3, which is followed by a brief survey of the existing related work in section 4. Section 5 presents our method for devising a balanced multiresolution scheme accompanied by two examples. We demonstrate the application of a balanced multiresolution scheme devised by our method in real-time focus+context visualization with experimental results in section 6. In section 7, we discuss some characteristics of our method with possible directions for future work. Finally, section 8 concludes the article. We also provide two appendices with additional examples of balanced multiresolution schemes devised by our method.

2. Notation

In this article, we adopted the notations for representing multiresolution operations used by Samavati *et al.* in [SBO07]. Multiresolution operations are specified in terms of analysis filter matrices $\mathbf{A}^k$ and $\mathbf{B}^k$ and synthesis filter matrices $\mathbf{P}^k$ and $\mathbf{Q}^k$. Given a column vector of samples C^k, a lower-resolution sample vector C^{k-1} is obtained by the application of a downsampling filter on C^k. This can be expressed by the matrix equation

$$C^{k-1} = \mathbf{A}^k C^k.$$

The *details* D^{k-1}, lost after downsampling, are captured using $\mathbf{B}^k$ as follows:

$$D^{k-1} = \mathbf{B}^k C^k.$$

This process of obtaining the low-resolution sample vector C^{k-1} and the corresponding details D^{k-1} from a given high-resolution sample vector C^k is known as *decomposition*. Note that the sequences of samples along each dimension of an image can be treated independently during decomposition. Therefore, any such sequence of samples can form the column vector of samples C^k for decomposition.

The process of recovering the original high-resolution sample vector C^k from the previously obtained low-resolution sample vector C^{k-1} and the corresponding details D^{k-1} is known as *reconstruction*. The reconstruction process requires the refinement of the low-resolution sample vector C^{k-1} and the corresponding details D^{k-1} by the application of synthesis filters $\mathbf{P}^k$ and $\mathbf{Q}^k$ as follows:

$$C^k = \mathbf{P}^k C^{k-1} + \mathbf{Q}^k D^{k-1}.$$

This equation reverses the prior application of $\mathbf{A}^k$ and $\mathbf{B}^k$ on the given high-resolution sample vector C^k. Therefore, decomposition and reconstruction are inverse processes satisfying

$$\begin{bmatrix} \mathbf{A}^k \\ \mathbf{B}^k \end{bmatrix} \begin{bmatrix} \mathbf{P}^k & \mathbf{Q}^k \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

If we recursively decompose a high-resolution sample vector C^k into its coarser approximations $C^l, C^{l+1}, \ldots, C^{k-1}$ and details $D^l, D^{l+1}, \ldots, D^{k-1}$, then the sequence $C^l, D^l, D^{l+1}, \ldots, D^{k-1}$ is known as a *wavelet transform*. Here, $l < k$ and C^l is the very coarse approximation of the dataset. Each of $C^{l+1}, \ldots, C^{k-1}, C^k$ can be reconstructed from the wavelet transform $C^l, D^l, D^{l+1}, \ldots, D^{k-1}$.

To simply the notations for the rest of this article, we may omit the superscript k for the kth level of resolution assuming $F = C^k$, $C = C^{k-1}$, $D = D^{k-1}$, $\mathbf{A} = \mathbf{A}^k$, and $\mathbf{B} = \mathbf{B}^k$, $\mathbf{P} = \mathbf{P}^k$, and $\mathbf{Q} = \mathbf{Q}^k$. We further assume that the matrices are of appropriate size to satisfy the following equations:

$$C = \mathbf{A}F, \tag{1}$$

$$D = \mathbf{B}F, \tag{2}$$

$$F = \mathbf{P}C + \mathbf{Q}D. \tag{3}$$

Let $\mathbf{a}$ and $\mathbf{b}$ denote the filter vectors containing the nonzero entries in a representative row of $\mathbf{A}$ and $\mathbf{B}$, respectively. Similarly, let $\mathbf{p}$ and $\mathbf{q}$ stand for the filter vectors containing the nonzero entries in a representative column of $\mathbf{P}$ and $\mathbf{Q}$, respectively. Furthermore, let $sizeof(V)$ represent the number of elements in vector V and the widths of filter vectors $\mathbf{a}$ and $\mathbf{b}$ be represented by w_a and w_b, respectively, i.e. $sizeof(\mathbf{a}) = w_\mathrm{a}$ and $sizeof(\mathbf{b}) = w_\mathrm{b}$.

3. Problem

As mentioned earlier, the method we present in this article can be used to devise a balanced multiresolution scheme for a given set of regular multiresolution filters comprised of filter vectors **a**, **b**, **p**, and **q**, such as the set given in (4). If each of **a**, **b**, **p**, and **q** is either symmetric or antisymmetric, our method will guarantee a perfect reconstruction without the use of any extraordinary boundary filters.

Due to the regularity present in an image structure, the multiresolution methods applicable to curves and regular meshes are also applicable to images (see subsection 4). For end point and boundary interpolations, extraordinary filters (as opposed to regular filters) are used in multiresolution methods for curves and regular meshes, respectively. However, the use of extraordinary filters at image boundaries for boundary interpolation assigns incongruous importance to the image boundaries. So for 2D or 3D image decomposition, the general practice is to use symmetric extensions near the image boundaries to avoid boundary case evaluations using extraordinary filters [SBO07].

However, an arbitrary choice of symmetric extension for decomposition while using a given set of multiresolution filters may eventually lead to the use of extraordinary boundary filters for a perfect reconstruction. This can also make on-demand reconstruction of image parts corresponding to a ROI computationally untidy near the image boundaries. Therefore, a careful setup of symmetric/antisymmetric extensions for both decomposition and reconstruction is required, which can be obtained by our presented method. Hence, we define our problem statement as follows.

Problem definition. Given a set of regular multiresolution filters in the form of symmetric/antisymmetric filter vectors **a**, **b**, **p**, and **q**, devise a balanced multiresolution scheme applicable to a high-resolution column vector of samples F that satisfies:

(i) $C = \mathbf{A}F'$ and $D = \mathbf{B}F'$, analogous to equations (1) and (2), where $F \rightarrow F'$ through symmetric extensions at its boundaries and the nonzero entries in each row of $\mathbf{A}$ and $\mathbf{B}$ correspond to the regular filters in the given filter vectors $\mathbf{a}$ and $\mathbf{b}$, respectively;
(ii) $sizeof(C) = sizeof(D)$ i.e. a balanced decomposition;
(iii) $sizeof(C) + sizeof(D) = sizeof(F)$ i.e. a compact representation of the resulting wavelet transform, hereafter referred to as a *balanced wavelet transform*; and
(iv) $F = \mathbf{P}C' + \mathbf{Q}D'$, analogous to equation (3), where $C \rightarrow C'$ and $D \rightarrow D'$ through symmetric/antisymmetric extensions at their boundaries and the nonzero entries in each column of $\mathbf{P}$ and $\mathbf{Q}$ correspond to the regular filters in the given filter vectors $\mathbf{p}$ and $\mathbf{q}$, respectively.

4. Related Work

In the next three subsections, we review the existing related work within the following three categories: multiresolution, symmetric and antisymmetric extensions, and focus+context visualization.

4.1. Multiresolution

Regular meshes. Here we review the multiresolution methods applicable to curves and tensor-product meshes (surfaces and volumes) given their applicability to multidimensional images due to their regular structure.

Hierarchical representation of multiresolution tensor-product surfaces was made possible due to the pioneering work of Forsey and Bartels [FB88]. They localized the editing effect in a desired manner on tensor-product surfaces through hierarchically controlled subdivisions. This was done by adding finer sets of B-splines onto existing coarse sets. However, it resulted in an over-representation because the union of the sets of basis functions from different resolutions did not form a set of basis functions. Adding complementary basis functions to the coarse set of basis functions is a possible way to resolve the problem of over-representation. This means of supporting multiresolution is closely aligned to the wavelet theory approach to multiresolution [SDS96]. Wavelet representations of details may, however, introduce undesired undulations, as pointed out by Gortler and Cohen [GC95]. Furthermore, under this approach, optimizing the behaviour of the analysis (decomposition) using least squares is difficult due to the need to support interactive mesh manipulations [ZSS97].

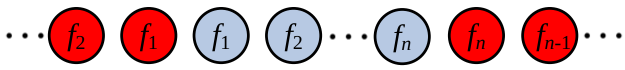
Samavati and Bartels pioneered in their work on a mathematically clean and efficient approach to multiresolution based on reverse subdivision [SB99, BS00, BGS06, BS11]. Under this approach, during the analysis, each coarse vertex is obtained by efficiently solving a local least squares optimization problem. The use of least squares optimization reduces the undesired undulations. Additionally, the resulting wavelets provide a much more compact support compared to the conventional wavelets for curves and regular surfaces. Some of the examples demonstrating the application of our proposed method use multiresolution filters resulting from this approach (see the examples in section 5, for instance).

Images. Notable existing approaches obtaining a multiresolution representation supporting context-aware visualization of 3D images include the wavelet tree [WS05], segmentation of texture-space into an octree [LHJ99, PTCF02, PHF07], octree-based tensor approximation hierarchy [SGM*11], and trilinear resampling on the Graphics Processing Unit (GPU) coupled with the deformation of regularly partitioned image regions [WWLM11]. For 4D images, the wavelet-based time-space partitioning (WTSP) tree was proposed in [WS05]. In [WS05], Haar [Haa10, SDS96] and Daubechies's D4 [Dau88] wavelets were used to construct the wavelet transforms in each node of the wavelet and WTSP trees.

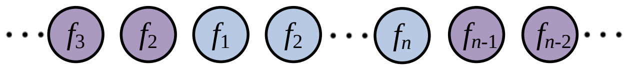
4.2. Symmetric and Antisymmetric Extensions

As mentioned earlier, we achieve balanced decomposition and subsequent perfect reconstruction based on the use of an appropriate combination of symmetric and antisymmetric extensions near the image and detail boundaries. In the literature, symmetric and antisymmetric extensions were used in the context of various types of wavelet transforms [LL00, KZT02, AW03, LS08]. In contrast, our proposed method allows the construction of a *balanced wavelet transform*.

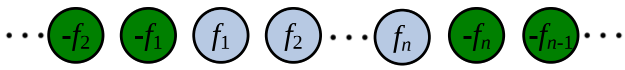
Figure 2 shows three types of extensions as defined in [KNI94]. Consider a sequence of n samples $(f_1, f_2, \ldots, f_n)$, corresponding to a column vector of samples $\begin{bmatrix} f_1 & f_2 & \ldots & f_n \end{bmatrix}^T$, where $n \in \mathbb{N}$ and $n \geq 3$. Figure 2(a), 2(b), and 2(c)



(a) Half-sample symmetry.



(b) Whole-sample symmetry.



(c) Half-sample antisymmetry.

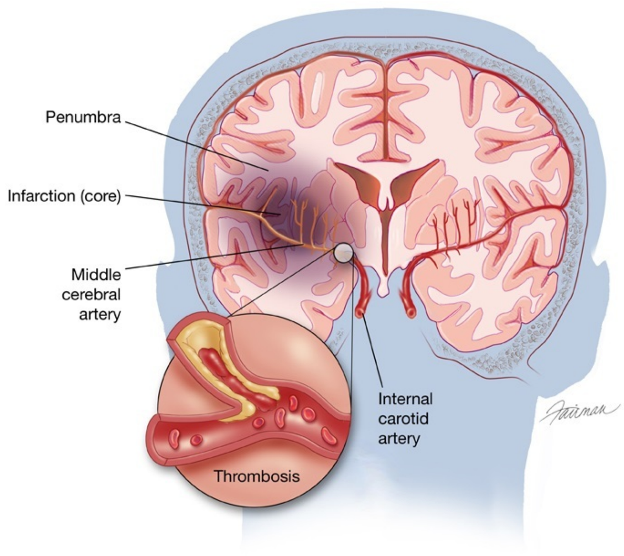
Figure 2: Symmetric and antisymmetric extensions.

show the extended sequences obtained through half-sample symmetric, whole-sample symmetric, and half-sample antisymmetric extensions, respectively, at both ends of $(f_1, f_2, \ldots, f_n)$. Whole-sample antisymmetry, not shown in Figure 2, can be obtained by negating the samples in the extensions of Figure 2(b). Note that in real applications, the types of extensions at both ends of a sequence do not necessarily have to be the same (as used in Figure 11, for example).

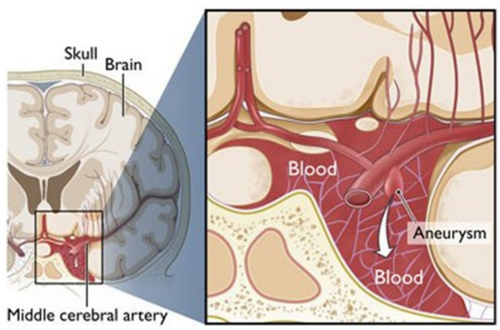
To be consistent with the coloring used in Figure 2, from this point forward in this article, notations and figures may use red, purple, and green to denote the samples introduced by half-sample symmetric, whole-sample symmetric, and half-sample antisymmetric extensions, respectively.

4.3. Focus+Context Visualization

Because we chose to demonstrate the use of a balanced multiresolution scheme resulting from our method in a real-time focus+context visualization application, here we review some of the notable related work.



(a) A circular ROI.



(b) A rectangular ROI.

Figure 3: Traditional focus+context visualization in medical illustrations. (a) Thrombosis in human brain. Copyright Fairman Studios, LLC. Used with permission. (b) Aneurysm in human brain. Copyright University of Washington Medicine Stroke Center.

In many visualization tasks, it is useful to simultaneously visualize both the local and global views of the data, *possibly at different scales*, which is known as focus+context visualization. One approach to implement focus+context is to use the metaphor of lenses [TSS*06, WWLM11, HMC11]. This metaphor is inspired by techniques used in traditional medical (see Figure 3), technical, and scientific illustrations [Hod03].

Our implemented approach to focus+context visualization of multidimensional images is closest to the technique presented by Taerum *et al.* for the visualization of small-scale clinical volumetric datasets [TSS*06]. In their approach, the resolution of a given 3D image is reduced by one level using reverse subdivision [SB99, BS00], which is rendered during user interactions to achieve interactive frame rates. The 3D image is rendered in the original resolution while there is no user-interaction. The ROI identified by a query window is enlarged by the application of B-spline subdivision to allow different levels of smoothness. Therefore, the authors used only three different levels of resolution. In contrast, our implementation for multiresolution visualization of images provides a true multiresolution framework, where the resolutions of both the coarse image (providing context information) and the enlarged ROI (providing focus information) can be controlled by the user.

5. Methodology: Balanced Multiresolution

In this section, we explain and demonstrate by examples how our method achieves balanced decomposition and subsequent perfect reconstruction by choosing an appropriate combination of symmetric and antisymmetric extensions near the image and detail boundaries.

5.1. Balanced Decomposition

We defined balanced decomposition as the task of decomposing a high-resolution image into a low-resolution image and corresponding details of equal size. Balanced decomposition of a 3D image of dimensions $2w \times 2h \times 2s$ results in an image of dimensions $w \times h \times s$ after one level of widthwise, heightwise, and depthwise decomposition. To allow l levels of balanced decomposition, we need the following conditions to be satisfied: $2w = 2^l m$, $2h = 2^l n$, and $2s = 2^l p$, where $m, n, p \in \mathbb{Z}^+$. Disregarding the third dimension infers the same idea for a 2D image. Once the ideal dimensions are known, the high-resolution image should be uniformly resampled to those dimensions before the application of our balanced decomposition procedure.

Given the decomposition filter vectors $\mathbf{a}$ and $\mathbf{b}$, to achieve a balanced decomposition of a column vector containing an even number of fine samples F, we first decide on the type of symmetric extension to use for decomposition based on the parity of w_{a} and w_{b}. Then an extended column vector of fine samples F' is obtained from F, through the chosen type of symmetric extension, such that $sizeof(F')$ ensures the generation of $sizeof(F)/2$ coarse samples and $sizeof(F)/2$ detail samples by a subsequent application of filter vectors $\mathbf{a}$ and $\mathbf{b}$ on F', respectively.

Demonstration by example. Before we outline the general construction for the balanced decomposition process, here we demonstrate how it works for a given set of decomposition filter vectors. In this example, we consider the decomposition filter vectors $\mathbf{a}$ and $\mathbf{b}$ from following set of local regular multiresolution filters [SB04, SBO07]:

$$\begin{cases} \mathbf{a} = \begin{bmatrix} -\frac{1}{4} & \frac{3}{4} & \frac{3}{4} & -\frac{1}{4} \end{bmatrix}, \\ \mathbf{b} = \begin{bmatrix} \frac{1}{4} & -\frac{3}{4} & \frac{3}{4} & -\frac{1}{4} \end{bmatrix}, \\ \mathbf{p} = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} & \frac{3}{4} & \frac{1}{4} \end{bmatrix}, \\ \mathbf{q} = \begin{bmatrix} -\frac{1}{4} & -\frac{3}{4} & \frac{3}{4} & \frac{1}{4} \end{bmatrix}. \end{cases} \tag{4}$$

The filter vectors in (4) are known as the *short* filters of quadratic (third order) B-spline [SBO07] and were constructed by reversing Chaikin subdivision [Cha74]. Recall from section 2 that filter vectors $\mathbf{a}$ and $\mathbf{b}$ contain the nonzero entries in a representative row of analysis filter matrices $\mathbf{A}$ and $\mathbf{B}$, respectively.

For the purpose of demonstration, assume that we are given a fine column vector of 8 samples $F = \begin{bmatrix} f_1 & f_2 & \dots & f_8 \end{bmatrix}^T$, on which we have to perform a balanced decomposition. Provided $sizeof(F) = 8$, a balanced decomposition should result in column vectors of coarse samples $C = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}^T$ and detail samples $D = \begin{bmatrix} d_1 & d_2 & d_3 & d_4 \end{bmatrix}^T$.

In Figure 4, we present one possible setup to obtain such a balanced decomposition. It shows the application of equations $C = \mathbf{A}F'$ and $D = \mathbf{B}F'$, analogous to equations (1) and (2), where $F' = \begin{bmatrix} f_1 & f_1 & f_2 & \dots & f_8 & f_8 \end{bmatrix}^T$. First, note that F' was obtained by extending the given sample vector F by 2 extra samples. In general, when the dilation

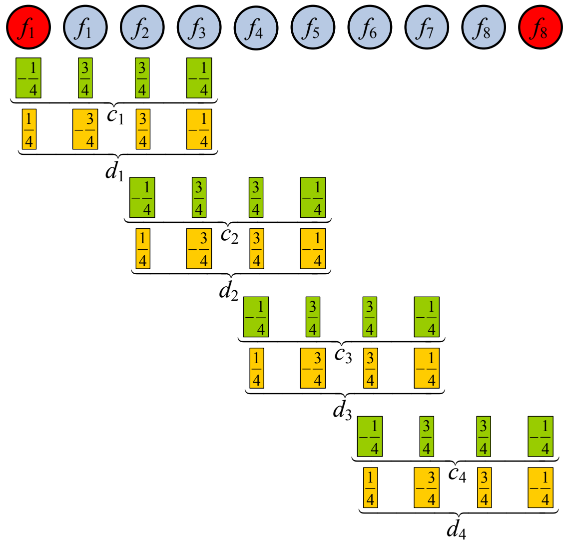


Figure 4: Balanced decomposition of 8 fine samples.

factor is 2, a given column vector of fine samples F, with $sizeof(F) = 2n$ for $n \in \mathbb{Z}^+$, does not have enough samples to accommodate n shifts of both $\mathbf{a}$ and $\mathbf{b}$ for generating n coarse and n detail samples, respectively. The number of extra samples x, required for a balanced decomposition can be obtained by the general formula:

$$\begin{aligned} x &= max(w_a, w_b) + 2(n-1) - 2n && (5) \\ \Rightarrow x &= max(w_a, w_b) - 2. && (6) \end{aligned}$$

Here we explain how equation 5 evaluates x. We need at least $max(w_a, w_b)$ fine samples to obtain both c_1 and d_1, which explains the first term on the right-hand side of equation 5. Next, because the dilation factor is 2, every 2 additional samples will guarantee the generation of an additional pair of c_i and d_i. Here, $i \in \{2, \ldots, n\}$ because we want to generate $|\{2, \ldots, n\}| = n - 1$ more coarse samples and $n - 1$ more detail samples to achieve a balanced decomposition. This indicates the need for an additional $2(n - 1)$ fine samples, justifying the addition of the second term on the right-hand side of equation 5. Therefore, subtracting $2n$ i.e. the $sizeof(F)$ in the third term gives us the required number of extra samples.

For the families of multiresolution filters we consider in this article, w_a and w_b are either both even or both odd. For example, see the decomposition filter vectors obtained from B-spline wavelets [SBO07], biorthogonal and reverse biorthogonal wavelets [CDF92, Dau92], and Meyer wavelets [Mey90, Dau92]. The multiresolution filter vectors obtained from most such wavelets and their scaling functions are available in commonly used mathematical software packages such as MATLAB [MAT14]. For the given filter vectors $\mathbf{a}$ and $\mathbf{b}$ in (4), because both w_a and w_b are even, observe that the extension of F by 2 extra samples to obtain F' was achieved by half-sample symmetric extension at both ends of F. Here we would have used whole-sample symmetric extension instead if both w_a and w_b were odd. Use of an appropriate type of symmetric extension is required to avoid the use of any extraordinary boundary filters for a perfect reconstruction. We justify our choice of symmetric extension for a balanced decomposition later in subsection 5.3.

Finally, as shown in Figure 4, the filter vectors $\mathbf{a}$ and $\mathbf{b}$ in (4) are applied to the samples in F' to obtain C and D in order to complete the balanced decomposition process. For instance, the coarse sample c_1 and the detail sample d_1 are computed from the first 4 samples in F' as follows:

$$\begin{cases} c_1 = -\frac{1}{4}f_1 + \frac{3}{4}f_1 + \frac{3}{4}f_2 - \frac{1}{4}f_3, \\ d_1 = \frac{1}{4}f_1 - \frac{3}{4}f_1 + \frac{3}{4}f_2 - \frac{1}{4}f_3. \end{cases} \quad (7)$$

Note that the total contribution of f_1 in the construction of c_1 is $\frac{1}{2}f_1$, written as $-\frac{1}{4}f_1 + \frac{3}{4}f_1$ in (7) through an implicit sample split operation. A similar sample split is observed in the construction of d_1, as shown in (7). Therefore, the symmetric extensions at both ends of F implicitly lead to a number of sample split operations during decomposition.

Therefore, for $n \in \mathbb{Z}^+$, a balanced multiresolution scheme based on the *short* filters of quadratic B-spline given in (4) can make use of the matrix equations

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} -\frac{1}{4} & \frac{3}{4} & \frac{3}{4} & -\frac{1}{4} & 0 & 0 & 0 & \cdots \\ 0 & 0 & -\frac{1}{4} & \frac{3}{4} & \frac{3}{4} & -\frac{1}{4} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_1 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n} \end{bmatrix}$$

and

$$\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & -\frac{3}{4} & \frac{3}{4} & -\frac{1}{4} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \frac{1}{4} & -\frac{3}{4} & \frac{3}{4} & -\frac{1}{4} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_1 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n} \end{bmatrix}$$

for the decomposition process, analogous to equations (1) and (2).

General construction. Now we present our general approach for achieving a balanced decomposition. Given the symmetric/antisymmetric decomposition filter vectors **a** and **b** containing only regular filters, carry out the following steps to achieve a balanced decomposition of a fine column vector of samples F, where $sizeof(F) = 2n$ for a suitably large $n \in \mathbb{Z}^+$.

1. Determine x, the number of extra samples required for a balanced decomposition using equation (6).
2. If both w_a and w_a are even, extend F with x extra samples using half-sample symmetric extension to obtain F'. Use whole-sample symmetric extension instead if both w_a and w_a are odd. Justification of our choice of symmetric extension can be found in subsection 5.3. To avoid giving inconsistent importance to any end (boundary) of F:
   (a) If x is even, introduce $x/2$ samples at each end of F.
   (b) If x is odd, introduce $\lfloor x/2 \rfloor$ samples at one end and $\lfloor x/2 \rfloor + 1$ samples at the other end of F. Let us refer to the end at which $\lfloor x/2 \rfloor + 1$ samples are introduced as the *odd end*. Alternate between the ends of F as the choice of the odd end during multiple levels of decomposition.
3. To obtain C and D such that $sizeof(C) = sizeof(D)$, use equations $C = \mathbf{A}F'$ and $D = \mathbf{B}F'$, analogous to equations (1) and (2).

5.2. *Perfect Reconstruction*

Given the reconstruction filter vectors **p** and **q** that can reverse the application of the decomposition filter vectors **a** and **b**, to achieve a perfect reconstruction of the column vector of fine samples F from its prior balanced decomposition into C and D, we first reconstruct as many interior samples of F as possible by the application of **p** and **q** on C and D, using equation (3). To evaluate the samples near each boundary (end) of F, we form a square system of linear equations based on the prior construction of corresponding boundary samples in C and D, where the unknowns constitute the boundary samples of F yet to be reconstructed. Symbolically solving two such square systems for the two boundaries of F reveals the extended versions of C and D (denoted by C' and D', respectively) required for a perfect reconstruction by the application of **p** and **q** using equation $F = \mathbf{P}C' + \mathbf{Q}D'$, analogous to equation (3).

Demonstration by example. Here we demonstrate how we perform a perfect reconstruction of F following its balanced decomposition to C and D by means of an example, before giving the general construction for our perfect reconstruction process. In this example, we consider the reconstruction filter vectors **p** and **q** given in (4). Recall from section 2 that filter vectors **p** and **q** contain the nonzero entries in a representative column of synthesis filter matrices **P** and **Q**, respectively.

This example to demonstrate our perfect reconstruction process is an extension of the example shown in Figure 4. So, from the resulting column vectors coarse samples $C = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}^T$ and detail samples $D = \begin{bmatrix} d_1 & d_2 & d_3 & d_4 \end{bmatrix}^T$ in subsection 5.1, we now want to reconstruct the corresponding column vector of fine samples $F = \begin{bmatrix} f_1 & f_2 & \dots & f_8 \end{bmatrix}^T$.

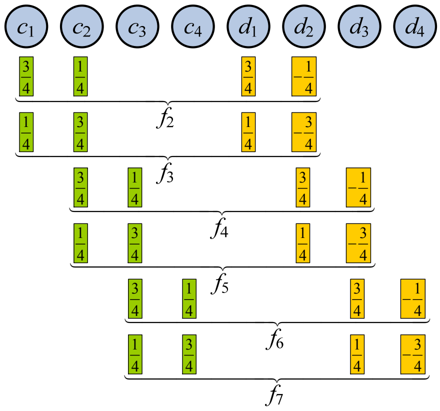


Figure 5: Reconstruction of 6 of the 8 fine samples.

In Figure 5, we show the application of the filter vectors **p** and **q** to the samples in C and D, respectively. For instance, the fine sample f_2 is reconstructed from the first two coarse samples and the first two detail samples as follows:

$$f_2 = \frac{3}{4}c_1 + \frac{1}{4}c_2 + \frac{3}{4}d_1 - \frac{1}{4}d_2.$$

Note that the application of the filter vectors **p** and **q** to the samples in C and D in Figure 5 left two samples, f_1 and f_8, near the two ends of F not reconstructed. Note that having two samples near the boundaries of F yet to reconstruct is specific to this example. The example in subsection 5.4 receives 5 samples yet to reconstruct at this stage. Now, to reconstruct f_1, we form the following 1×1 system of linear equations based on the prior construction of c_1 (as shown in Figure 4) to which f_1 made some contribution during decomposition:

$$\begin{aligned}
c_1 &= -\frac{1}{4}f_1 + \frac{3}{4}f_1 + \frac{3}{4}f_2 - \frac{1}{4}f_3 && (8)\\
\Rightarrow f_1 &= 2c_1 - \frac{3}{2}f_2 + \frac{1}{2}f_3\\
\Rightarrow f_1 &= 2c_1 - \frac{3}{2}\left(\frac{3}{4}c_1 + \frac{1}{4}c_2 + \frac{3}{4}d_1 - \frac{1}{4}d_2\right)\\
&\quad + \frac{1}{2}\left(\frac{1}{4}c_1 + \frac{3}{4}c_2 + \frac{1}{4}d_1 - \frac{3}{4}d_2\right)\\
\Rightarrow f_1 &= c_1 - d_1. && (9)
\end{aligned}$$

Although it appears from equation (9) that f_1 is not reconstructed using *regular filters*, our prior appropriate choice of symmetric extension to obtain F' from F (justified later in subsection 5.3) guarantees that we can rewrite f_1 using the regular filter values from **p** and **q** in (4). This is achieved by a rearrangement of the right-hand side of equation (9), which is implicitly equivalent to performing two sample split operations:

$$f_1 = \tfrac{1}{4}c_1 + \tfrac{3}{4}c_1 + \tfrac{1}{4}(-d_1) - \tfrac{3}{4}d_1. \tag{10}$$

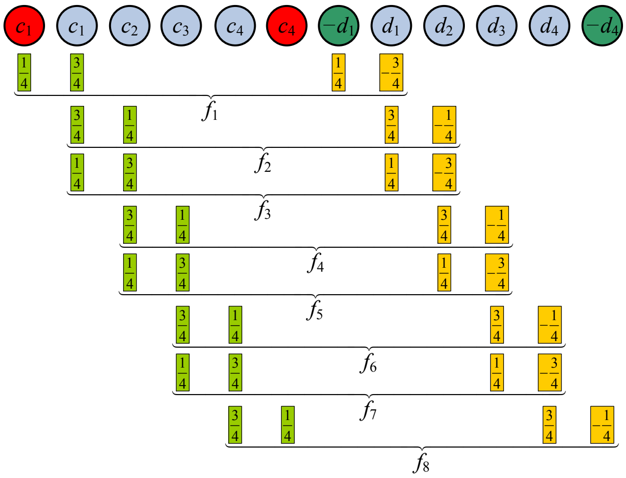


Figure 6: Perfect reconstruction of 8 fine samples.

This rewriting step is important because it allows the reconstruction of fine samples near the boundaries of F without the use of any extraordinary boundary filters. Equation (10) now yields the introduction of one extra coarse sample through half-sample symmetric extension and one extra detail sample through half-sample antisymmetric extension for the reconstruction of f_1, as shown in Figure 6. We use a similar approach to determine how to reconstruct the boundary sample f_8, resulting in

$$f_8 \;=\; \tfrac{3}{4}c_4 + \tfrac{1}{4}c_4 + \tfrac{3}{4}d_1 - \tfrac{1}{4}(-d_4), \tag{11}$$

as reflected in Figure 6. This concludes the perfect reconstruction process.

Therefore, based on our findings from (10) and (11), for a given column vector of $2n$ fine samples for a suitably large $n \in \mathbb{Z}^+$, we get

$$\begin{cases} f_1 &= \tfrac{1}{4}c_1 + \tfrac{3}{4}c_1 + \tfrac{1}{4}(-d_1) - \tfrac{3}{4}d_1, \\ f_{2n} &= \tfrac{3}{4}c_n + \tfrac{1}{4}c_n + \tfrac{3}{4}d_n - \tfrac{1}{4}(-d_n). \end{cases}$$

So a balanced multiresolution scheme based on the *short* filters of quadratic B-spline given in (4) will make use of the matrix equation

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & \frac{3}{4} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{3}{4} & \frac{1}{4} & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{1}{4} & \frac{3}{4} & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & \cdots & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4} & \frac{3}{4} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \frac{3}{4} & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & \cdots & \frac{1}{4} & \frac{3}{4} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{3}{4} & \frac{1}{4} \end{bmatrix} \begin{bmatrix} c_1 \\ c_1 \\ c_2 \\ \vdots \\ c_n \\ c_n \end{bmatrix} + \begin{bmatrix} \frac{1}{4} & -\frac{3}{4} & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{3}{4} & -\frac{1}{4} & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{1}{4} & -\frac{3}{4} & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{4} & -\frac{1}{4} & \cdots & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4} & -\frac{3}{4} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \frac{3}{4} & -\frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & \cdots & \frac{1}{4} & -\frac{3}{4} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{3}{4} & -\frac{1}{4} \end{bmatrix} \begin{bmatrix} -d_1 \\ d_1 \\ d_2 \\ \vdots \\ d_n \\ -d_n \end{bmatrix}$$

for the reconstruction process, analogous to equation (3).

General construction. Now we describe our general approach to achieve perfect reconstruction. Given the symmetric/antisymmetric reconstruction filter vectors **p** and **q** containing only regular filters that can reverse the application of the decomposition filter vectors **a** and **b**, carry out the following steps to perfectly reconstruct the column vector of fine samples F from its prior balanced decomposition into C and D.

1. Assume that $F = \left[\begin{array}{ccc} F_l{}^T & F_m{}^T & F_r{}^T \end{array}\right]^T$, where F_l and F_r respectively contain some samples at the left and right boundaries of F, and F_m contains the remaining interior samples of F. To reconstruct the samples in F_m, use the equation $F_m = \mathbf{P}C + \mathbf{Q}D$, analogous to equation (3). The samples in F_l and F_r are yet to be reconstructed.
   (In the example above, we had $F_l = [f_1]$, $F_m = \left[\begin{array}{cccc} f_2 & f_3 & \dots & f_7 \end{array}\right]^T$, and $F_r = [f_8]$. Note that F_l and F_r may contain more samples; for instance, the F_l and F_r encountered in 5.4 have 2 and 3 samples, respectively.)
2. To reconstruct the samples in F_l:
   (a) Form a system of linear equations based on the prior construction of some coarse and detail boundary samples, to which the fine samples in F_l made some contributions during the decomposition process. It should be a $q \times q$ system, where $q = sizeof(F_l)$ and the unknowns are the samples of F_l.
   (For example, see the 1×1 system formed by equation (8) and the 2×2 system formed by the two equations in (19).)
   (b) Solving the system formed in step 2(a) symbolically will evaluate the samples in F_l as a linear combination of some samples from C and D.
   (For example, see equation (9) and the two equations in (20).)
   (c) Rewrite the linear combination(s) of coarse and detail samples on the right-hand side(s) of the equation(s) obtained in step 2(b) using the regular filter values from the filter vectors **p** and **q** as coefficients. Such rewriting of fine samples here correlates to performing sample split operations. This will reveal the following two pieces of information applicable to the left boundaries of C and D for a perfect reconstruction: (i) the type of symmetric/antisymmetric extension that must be used and (ii) the number of extra samples that must be to introduced.
   (For example, see equation (10) and the equations in (21).)
3. Use an approach similar to that in step 2 to reconstruct the samples in F_r.

Note that steps 2-3 above allow the generation of C' and D' respectively from C and D, such that condition (iv) of the problem definition given in section 3 is satisfied.

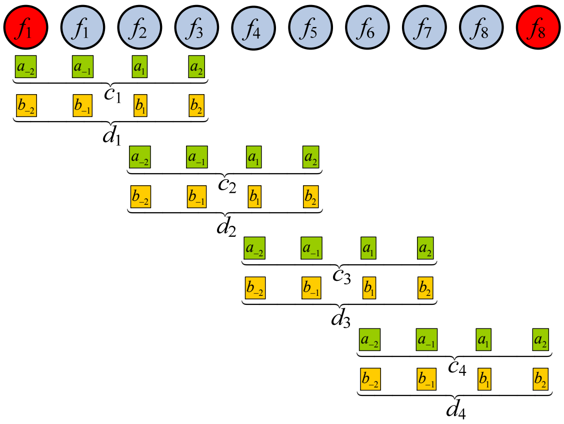
5.3. Choice of Symmetric Extension for Decomposition

Claim. For a given set of symmetric/antisymmetric multiresolution filter vectors **a**, **b**, **p**, and **q**, even values of w_a and w_b imply the use of half-sample symmetric extensions at the image boundaries during a balanced decomposition to ensure a perfect reconstruction only using the regular reconstruction filters from **p** and **q**. On the other hand, odd values of w_a and w_b imply the use of whole-sample symmetric extensions instead.

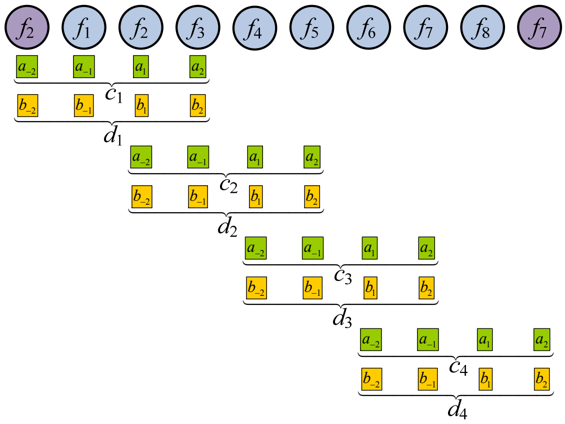
Proof outline. We outline the proof by means of an example that makes use of the filter vectors containing only regular filters,

$$\left\{\begin{array}{lcl} \mathbf{a} & = & \left[\begin{array}{cccc} a_{-2} & a_{-1} & a_1 & a_2 \end{array}\right], \\ \mathbf{b} & = & \left[\begin{array}{cccc} b_{-2} & b_{-1} & b_1 & b_2 \end{array}\right], \\ \mathbf{p} & = & \left[\begin{array}{cccc} p_{-2} & p_{-1} & p_1 & p_2 \end{array}\right], \\ \mathbf{q} & = & \left[\begin{array}{cccc} q_{-2} & q_{-1} & q_1 & q_2 \end{array}\right]. \end{array}\right. \tag{12}$$

The widths of the filter vectors **a**, **b**, **p**, and **q** in (12) are assumed to be 4 as in the case of the filter vectors containing the *short* filters of quadratic B-spline in (4). So, here w_a and w_b are even. Next, two possible balanced decompositions of a fine column vector of 8 samples $F = \left[\begin{array}{cccc} f_1 & f_2 & \dots & f_8 \end{array}\right]^T$ are shown by the use of half-sample and whole-sample symmetric extensions at its boundaries in Figures 7(a) and 7(b), respectively.



(a) Balanced decomposition using half-sample symmetric extension.



(b) Balanced decomposition using whole-sample symmetric extension.

Figure 7: Balanced decomposition of 8 fine samples.

Now, our goal is to perfectly reconstruct F from the column vectors of coarse samples $C = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}^T$ and detail samples $D = \begin{bmatrix} d_1 & d_2 & d_3 & d_4 \end{bmatrix}^T$ using only the regular reconstruction filters vectors $\mathbf{p}$ and $\mathbf{q}$ from (12) as shown in Figure 8.

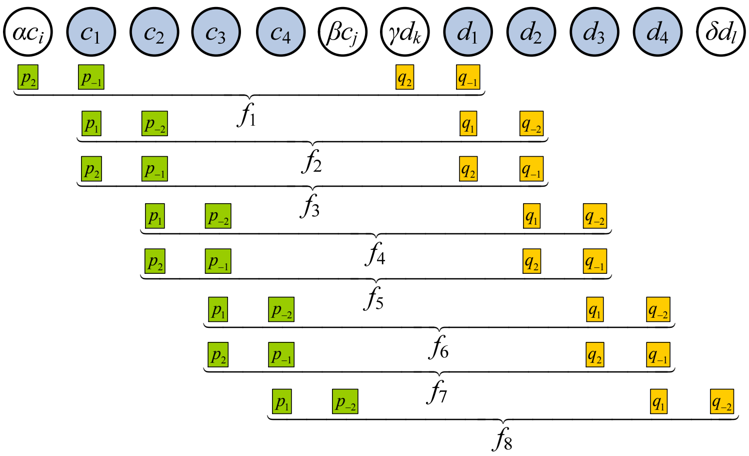


Figure 8: Perfect reconstruction of 8 fine samples.

We intend to evaluate the unknowns in Figure 8, which are $\alpha c_i \in \{-c_1, c_1, -c_2, c_2\}$, $\beta c_j \in \{-c_3, c_3, -c_4, c_4\}$, $\gamma d_k \in \{-d_1, d_1, -d_2, d_2\}$, and $\delta d_l \in \{-d_3, d_3, -d_4, d_4\}$ near the boundaries of C and D. Once evaluated, these will reveal the type of symmetric/antisymmetric extensions to be used at the boundaries of C and D to ensure a perfect reconstruction using only the regular reconstruction filters. Here $\alpha, \beta, \gamma, \delta \in \{+, -\}$ represent the signs of c_i, c_j, d_k and d_l, respectively. When negative, they allow the representation of antisymmetric extensions.

Now, let us try to evaluate αc_i. As shown in Figure 8, αc_i contributes to the reconstruction of f_1. If we consider the balanced decomposition shown in Figure 7(a) and try to evaluate f_1 following our general approach from subsection 5.2, we get

$$\begin{aligned} c_1 &= a_{-2} f_1 + a_{-1} f_1 + a_1 f_2 + a_2 f_3 \\ \Rightarrow f_1 &= \frac{1}{a_{-2} + a_{-1}} c_1 - \frac{a_1}{a_{-2} + a_{-1}} f_2 - \frac{a_2}{a_{-2} + a_{-1}} f_3 \end{aligned}$$

$$
\begin{aligned}
\Rightarrow f_1 &= \frac{1}{a_{-2}+a_{-1}} c_1 - \frac{a_1}{a_{-2}+a_{-1}} \left(p_1 c_1 + p_{-2} c_2 + q_1 d_1 + q_{-2} d_2\right) \\
&\quad - \frac{a_2}{a_{-2}+a_{-1}} \left(p_2 c_1 + p_{-1} c_2 + q_2 d_1 + q_{-1} d_2\right) \\
\Rightarrow f_1 &= \left(\frac{1 - a_1 p_1 - a_2 p_2}{a_{-2}+a_{-1}}\right) c_1 + \left(\frac{-a_1 p_{-2} - a_2 p_{-1}}{a_{-2}+a_{-1}}\right) c_2 \\
&\quad + \left(\frac{-a_1 q_1 - a_2 q_2}{a_{-2}+a_{-1}}\right) d_1 + \left(\frac{-a_1 q_{-2} - a_2 q_{-1}}{a_{-2}+a_{-1}}\right) d_2. \qquad (13)
\end{aligned}
$$

Next, if we consider the balanced decomposition shown in Figure 7(b) and try to evaluate f_1 following our general approach from subsection 5.2, we get

$$
\begin{aligned}
c_1 &= a_{-2} f_2 + a_{-1} f_1 + a_1 f_2 + a_2 f_3 \\
\Rightarrow f_1 &= \frac{1}{a_{-1}} c_1 - \frac{a_{-2}+a_1}{a_{-1}} f_2 - \frac{a_2}{a_{-1}} f_3 \\
\Rightarrow f_1 &= \frac{1}{a_{-1}} c_1 - \frac{a_{-2}+a_1}{a_{-1}} \left(p_1 c_1 + p_{-2} c_2 + q_1 d_1 + q_{-2} d_2\right) \\
&\quad + \frac{a_2}{a_{-1}} \left(p_2 c_1 + p_{-1} c_2 + q_2 d_1 + q_{-1} d_2\right) \\
\Rightarrow f_1 &= \left(\frac{1 - a_{-2} p_1 - a_1 p_1 - a_2 p_2}{a_{-1}}\right) c_1 + \left(\frac{-a_{-2} p_2 - a_1 p_2 - a_2 p_{-1}}{a_{-1}}\right) c_2 \\
&\quad + \left(\frac{-a_{-2} q_1 - a_1 q_1 - a_2 q_2}{a_{-1}}\right) d_1 + \left(\frac{-a_{-2} q_2 - a_1 q_2 - a_2 q_{-1}}{a_{-1}}\right) d_2. \qquad (14)
\end{aligned}
$$

Let the filter values multiplied to c_1 and c_2 in the reconstruction of f_1 be denoted by $w(c_1)$ and $w(c_2)$, respectively. In (13),

$$
\begin{cases}
w(c_1) = \frac{1 - a_1 p_1 - a_2 p_2}{a_{-2}+a_{-1}}, \\
w(c_2) = \frac{-a_1 p_{-2} - a_2 p_{-1}}{a_{-2}+a_{-1}},
\end{cases} \qquad (15)
$$

which result from using half-sample symmetric extension at the left boundary F for a balanced decomposition. On the other hand, in (14),

$$
\begin{cases}
w(c_1) = \frac{1 - a_{-2} p_1 - a_1 p_1 - a_2 p_2}{a_{-1}}, \\
w(c_2) = \frac{-a_{-2} p_2 - a_1 p_2 - a_2 p_{-1}}{a_{-1}},
\end{cases} \qquad (16)
$$

which result from using whole-sample symmetric extension instead. Now, according to Figure 8, f_1 is reconstructed as follows:

$$
f_1 = p_2(\alpha c_i) + p_{-1} c_1 + q_2(\alpha d_k) - q_{-1} d_1. \qquad (17)
$$

If we consider $\alpha c_i = -c_1$ in equation (17) for example, then $w(c_1) = -p_2 + p_{-1}$ and $w(c_2) = 0$. If $-c_1$ is substituted in Figure 8 in place of αc_i, it would then reveal the need for half-sample antisymmetric extension for the left boundary of C to be used during reconstruction. In this manner, Table 1 lists the sufficient conditions for all possible values of αc_i. Note that each possible value of αc_i yields a particular type of extension (listed in Table 1) for the left boundary of C.

Now, if we substitute the actual values of the corresponding regular filters of quadratic B-spline from (4) in (15) and (16), we find that (15) only satisfies the sufficient conditions under case I (i.e. $\alpha c_i = c_1$) in Table 1 and (16) does not satisfy the sufficient conditions under any of the cases. Recall that (15) was obtained by the use of half-sample symmetric extension on the left boundary of F for a balanced decomposition. This implies that the use of half-sample symmetric extension at the left boundary of F for a balanced decomposition will ensure the perfect reconstruction of that boundary olely using regular reconstruction filters. Similarly, for the regular filters of quadratic B-spline from (4), we can show that $\beta c_j = c_4$, $\gamma d_k = -d_1$, and $\delta d_l = -d_4$; and they all require the use of half-sample symmetric extension at the boundaries of F for a balanced decomposition.

Table 1: Sufficient conditions for symmetric and antisymmetric extensions.

Case	Sufficient Conditions	αc_i	Type of Extension
I	$\begin{cases} w(c_1) = p_2 + p_{-1} \\ w(c_2) = 0 \end{cases}$	c_1	Half-sample symmetry
II	$\begin{cases} w(c_1) = -p_2 + p_{-1} \\ w(c_2) = 0 \end{cases}$	$-c_1$	Half-sample antisymmetry
III	$\begin{cases} w(c_1) = p_{-1} \\ w(c_2) = p_2 \end{cases}$	c_2	Whole-sample symmetry
IV	$\begin{cases} w(c_1) = p_{-1} \\ w(c_2) = -p_2 \end{cases}$	$-c_2$	Whole-sample antisymmetry

In the above manner, we can show that for any set of symmetric/antisymmetric filter vectors **a**, **b**, **p**, and **q**, where w_a and w_b are even, half-sample symmetric extension can be used at the boundaries of a column vector of fine samples for a balanced decomposition to ensure a perfect reconstruction only using the regular reconstruction filters from **p** and **q**. A similar proof can be outlined to show that odd values of w_a and w_b imply the use of whole-sample symmetric extension instead.

5.4. *Further Demonstration by Example*

The example in this subsection illustrates the use of decomposition filter vectors of odd width for a balanced decomposition as opposed to the even width of decomposition filter vectors in the previous example (subsections 5.1 and 5.2). Further examples are provided in Appendix A.

Balanced decomposition. Here we demonstrate our general approach described in subsection 5.1 using the decomposition filter vectors **a** and **b** from following set of local regular multiresolution filters [BS00, SBO07]:

$$\begin{cases} \mathbf{a} = \begin{bmatrix} \frac{1}{8} & -\frac{1}{2} & \frac{3}{8} & 1 & \frac{3}{8} & -\frac{1}{2} & \frac{1}{8} \end{bmatrix}, \\ \mathbf{b} = \begin{bmatrix} -\frac{1}{8} & \frac{1}{2} & -\frac{3}{4} & \frac{1}{2} & -\frac{1}{8} \end{bmatrix}, \\ \mathbf{p} = \begin{bmatrix} \frac{1}{8} & \frac{1}{2} & \frac{3}{4} & \frac{1}{2} & \frac{1}{8} \end{bmatrix}, \\ \mathbf{q} = \begin{bmatrix} \frac{1}{8} & \frac{1}{2} & \frac{3}{8} & -1 & \frac{3}{8} & \frac{1}{2} & \frac{1}{8} \end{bmatrix}. \end{cases} \tag{18}$$

The filter vectors in (18) are known as the *inverse powers of two* filters of cubic (fourth order) B-spline [SBO07]. We explain the balanced decomposition process using the decomposition filter vectors in (18) through the example shown in Figure 9. Similar to the previous example shown in Figure 4, here we have a column vector of 8 fine samples $F = \begin{bmatrix} f_1 & f_2 & \dots & f_8 \end{bmatrix}^T$ that we want to decompose into the column vectors of coarse samples $C = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}^T$ and detail samples $D = \begin{bmatrix} d_1 & d_2 & d_3 & d_4 \end{bmatrix}^T$.

Figure 9 shows one possible balanced decomposition using our general approach presented in subsection 5.1. Step 1 of our *general construction* given in subsection 5.1 reveals that 5 extra samples are required to ensure a balanced decomposition. As noted earlier, w_a and w_b for the filter vectors in (18) are odd. So according to step 2, whole-sample symmetric extension is used to introduce 2 extra samples at one end and 3 extra samples at the other end of F to obtain the extended column vector of fine samples $F' = \begin{bmatrix} f_3 & f_2 & f_1 & f_2 & \dots & f_8 & f_7 & f_6 & f_5 \end{bmatrix}^T$. Finally, according to step 3, the filter vectors **a** and **b** from (18) are applied to F' to obtain C and D by means of the equations $C = \mathbf{A}F'$ and $D = \mathbf{B}F'$, analogous to equations (1) and (2).

Therefore, for $n \in \mathbb{Z}^+$, a balanced multiresolution scheme based on the *inverse powers of two* filters of cubic B-spline given in (18) can make use of the matrix equations

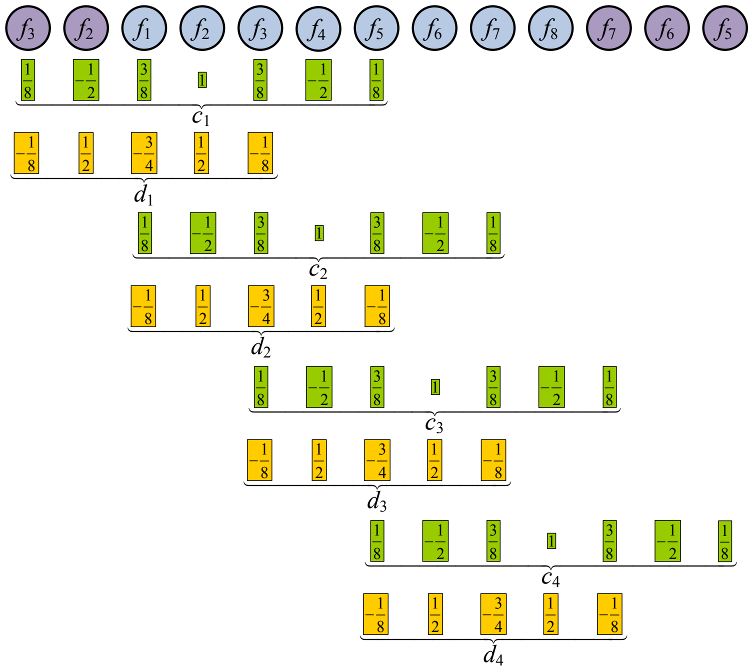


Figure 9: Balanced decomposition of 8 fine samples.

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} \frac{1}{8} & -\frac{1}{2} & \frac{3}{8} & 1 & \frac{3}{8} & -\frac{1}{2} & \frac{1}{8} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \frac{1}{8} & -\frac{1}{2} & \frac{3}{8} & 1 & \frac{3}{8} & -\frac{1}{2} & \frac{1}{8} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_3 \\ f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \\ f_{2n-2} \\ f_{2n-3} \end{bmatrix}$$

and

$$\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} -\frac{1}{8} & \frac{1}{2} & -\frac{3}{4} & \frac{1}{2} & -\frac{1}{8} & 0 & 0 & 0 & \cdots \\ 0 & 0 & -\frac{1}{8} & \frac{1}{2} & -\frac{3}{4} & \frac{1}{2} & -\frac{1}{8} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_3 \\ f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \end{bmatrix}$$

for the decomposition process, analogous to equations (1) and (2).

Perfect reconstruction. Here we demonstrate our general approach described in subsection 5.2 using the reconstruction filter vectors **p** and **q** given in (18). They can reverse the application of the decomposition filters vectors **a** and **b** from (18). Given the column vectors of coarse samples $C = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 \end{bmatrix}^T$ and detail samples

$D = \begin{bmatrix} d_1 & d_2 & d_3 & d_4 \end{bmatrix}^T$ (obtained as shown in Figure 9), we now want to perfectly reconstruct the column vector fine samples $F = \begin{bmatrix} f_1 & f_2 & \dots & f_8 \end{bmatrix}^T$.

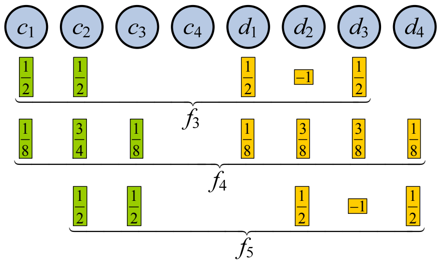


Figure 10: Reconstruction of 3 of the 8 fine samples.

Figure 10 shows the reconstruction of $F_m = \begin{bmatrix} f_3 & f_4 & f_5 \end{bmatrix}^T$ according to step 1 of our *general construction* given in subsection 5.2. $F_l = \begin{bmatrix} f_1 & f_2 \end{bmatrix}^T$ and $F_r = \begin{bmatrix} f_6 & f_7 & f_8 \end{bmatrix}^T$ are yet to be reconstructed.

Next, following step 2(a) of our given *general construction*, we form the following system of 2 linear equations in 2 unknowns (f_1 and f_2 in F_l):

$$\begin{cases} c_1 = \frac{1}{8}f_3 - \frac{1}{2}f_2 + \frac{3}{8}f_1 + f_2 + \frac{3}{8}f_3 - \frac{1}{2}f_4 + \frac{1}{8}f_5, \\ d_1 = -\frac{1}{8}f_3 + \frac{1}{2}f_2 - \frac{3}{4}f_1 + \frac{1}{2}f_2 - \frac{1}{8}f_3. \end{cases} \tag{19}$$

The equations in (19) were obtained from Figure 9, which shows how c_1 and d_1 were computed during decomposition. Note that in (19), we can replace f_3, f_4, and f_5 with the corresponding linear combinations of coarse and detail samples from Figure 10. Then following step 2(b), solving the 2×2 system formed by the equations in (19) gives

$$\begin{cases} f_1 = c_1 - d_1 + d_2, \\ f_2 = \frac{7}{8}c_1 + \frac{1}{8}c_2 + \frac{3}{8}d_1 + \frac{1}{2}d_2 + \frac{1}{8}d_3. \end{cases} \tag{20}$$

Now, according to step 2(c), the equations in (20) can be rewritten as follows such that the coefficients of the coarse and detail samples are all regular filters from (18):

$$\begin{cases} f_1 = \frac{1}{2}c_1 + \frac{1}{2}c_1 + \frac{1}{2}d_2 + (-1)d_1 + \frac{1}{2}d_2, \\ f_2 = \frac{1}{8}c_1 + \frac{3}{4}c_1 + \frac{1}{8}c_2 + \frac{1}{8}d_2 + \frac{3}{8}d_1 + \frac{3}{8}d_2 + \frac{1}{8}d_3. \end{cases} \tag{21}$$

This rewriting required two implicit sample split operations on the right-hand side of each equation in (21).

Finally, following step 3 of our *general construction* to reconstruct F_r, we form the following system of 3 linear equations in 3 unknowns (f_6, f_7, and f_8 in F_r):

$$\begin{cases} c_3 = \frac{1}{8}f_3 - \frac{1}{2}f_4 + \frac{3}{8}f_5 + f_6 + \frac{3}{8}f_7 - \frac{1}{2}f_8 + \frac{1}{8}f_7, \\ c_4 = \frac{1}{8}f_5 - \frac{1}{2}f_6 + \frac{3}{8}f_7 + f_8 + \frac{3}{8}f_7 - \frac{1}{2}f_6 + \frac{1}{8}f_5, \\ d_4 = -\frac{1}{8}f_5 + \frac{1}{2}f_6 - \frac{3}{4}f_7 + \frac{1}{2}f_8 - \frac{1}{8}f_7. \end{cases} \tag{22}$$

The equations in (22) were obtained from Figure 9, which shows how c_3, c_4, and d_4 were evaluated during decomposition. Observe that in (22), we can replace f_3, f_4, and f_5 with the corresponding linear combinations of coarse and detail samples from Figure 10. Then solving the 3×3 system formed by the equations in (22) gives

$$\begin{cases} f_6 = \frac{1}{8}c_2 + \frac{3}{4}c_3 + \frac{1}{8}c_4 + \frac{1}{8}d_2 + \frac{3}{8}d_3 + \frac{1}{2}d_4, \\ f_7 = \frac{1}{2}c_3 + \frac{1}{2}c_4 + \frac{1}{2}d_3 - \frac{1}{2}d_4, \\ f_8 = \frac{1}{4}c_3 + \frac{3}{4}c_4 + \frac{1}{4}d_3 + \frac{3}{4}d_4. \end{cases} \tag{23}$$

Now, the equations in (23) can be rewritten as follows such that the coefficients of the coarse and detail samples are all regular filters from (18):

$$\begin{cases} f_6 &= \frac{1}{8}c_2 + \frac{3}{4}c_3 + \frac{1}{8}c_4 + \frac{1}{8}d_2 + \frac{3}{8}d_3 + \frac{3}{8}d_4 + \frac{1}{8}d_4, \\ f_7 &= \frac{1}{2}c_3 + \frac{1}{2}c_4 + \frac{1}{2}d_3 + (-1)d_4 + \frac{1}{2}d_4, \\ f_8 &= \frac{1}{8}c_3 + \frac{3}{4}c_4 + \frac{1}{8}c_3 + \frac{1}{8}d_3 + \frac{3}{8}d_4 + \frac{3}{8}d_4 + \frac{1}{8}d_3. \end{cases} \tag{24}$$

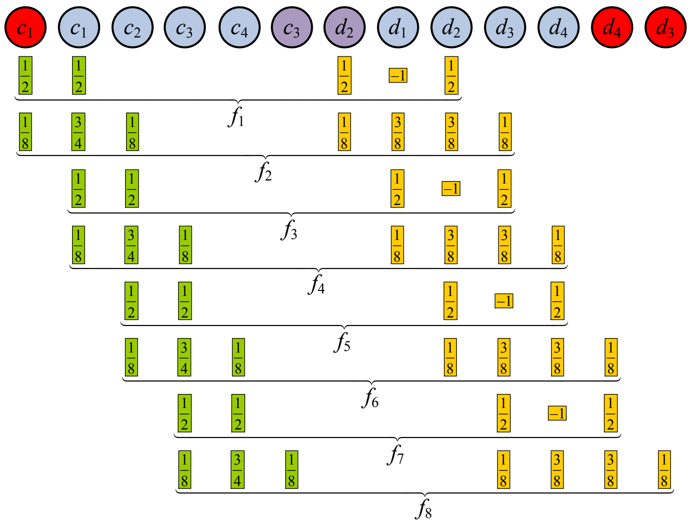


Figure 11: Perfect reconstruction of 8 fine samples.

As we mentioned in the *general construction* given in subsection 5.2, note that the equations in (21) and (24) yield a specific type of symmetric extension for each boundary of C and D as shown in Figure 11. Therefore, based on (21) and (24), for a given column vector of $2n$ fine samples ($n \in \mathbb{Z}^+$), we get

$$\begin{cases} f_1 &= \frac{1}{2}c_1 + \frac{1}{2}c_1 + \frac{1}{2}d_2 + (-1)d_1 + \frac{1}{2}d_2, \\ f_2 &= \frac{1}{8}c_1 + \frac{3}{4}c_1 + \frac{1}{8}c_2 + \frac{1}{8}d_2 + \frac{3}{8}d_1 + \frac{3}{8}d_2 + \frac{1}{8}d_3, \\ f_{2n-2} &= \frac{1}{8}c_{n-2} + \frac{3}{4}c_{n-1} + \frac{1}{8}c_n + \frac{1}{8}d_{n-2} + \frac{3}{8}d_{n-1} + \frac{3}{8}d_n + \frac{1}{8}d_n, \\ f_{2n-1} &= \frac{1}{2}c_{n-1} + \frac{1}{2}c_n + \frac{1}{2}d_{n-1} + (-1)d_n + \frac{1}{2}d_n, \\ f_{2n} &= \frac{1}{8}c_{n-1} + \frac{3}{4}c_n + \frac{1}{8}c_{n-1} + \frac{1}{8}d_{n-1} + \frac{3}{8}d_n + \frac{3}{8}d_n + \frac{1}{8}d_{n-1}. \end{cases}$$

So a balanced multiresolution scheme based on the *inverse powers of two* filters of cubic B-spline given in (18) can make use of the matrix equation

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{bmatrix} \begin{bmatrix} c_1 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \\ c_{n-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2} & -1 & \frac{1}{2} & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{1}{2} & -1 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix} \begin{bmatrix} d_2 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \\ d_n \\ d_{n-1} \end{bmatrix}$$

for the reconstruction process, analogous to equation (3).

6. Application in Focus+Context Visualization

Multiscale 2D and 3D image visualization applications often exploit query window-based focus+context visualization for image exploration and navigation purposes. A low-resolution approximation is rendered to provide the context and a selected portion of that low-resolution approximation defining the focus, also known as the ROI, is rendered as a close-up in high-resolution. While such visualization is supported by an underlying wavelet transform, it is necessary to reconstruct the high-resolution approximation of the ROI on demand from the low-resolution approximation and corresponding details. Here the use of a *balanced wavelet transform* constructed by our proposed method makes locating the details straightforward. For instance, observe the reconstruction of interior samples in Figures 5 and 10. If the first coarse sample for the reconstruction of a fine sample is c_i, then first detail sample to use in the reconstruction of that fine sample is d_i. This may not have been the case if we had an unequal number of coarse and detail samples from decomposition. Also, the only additional step required to reconstruct the fine samples near the boundaries is the use of specific symmetric/antisymmetric extensions, because our method completely eliminates the need for extraordinary boundary filters.

6.1. Overview of Visualization Tool

We have implemented a visualization tool prototype named *Focus+Context Studio* to test our presented balanced multiresolution framework for images. It robustly allows real-time multilevel focus+context visualization of large-scale 2D and 3D images, supported by multiple movable query windows defining ROIs at different resolutions. It currently uses the balanced multiresolution scheme we devised using the *short* filters of quadratic B-spline in (4), as described in the examples shown in subsections 5.1 and 5.2. At the moment, all the query windows are 32×32 samples in dimension.

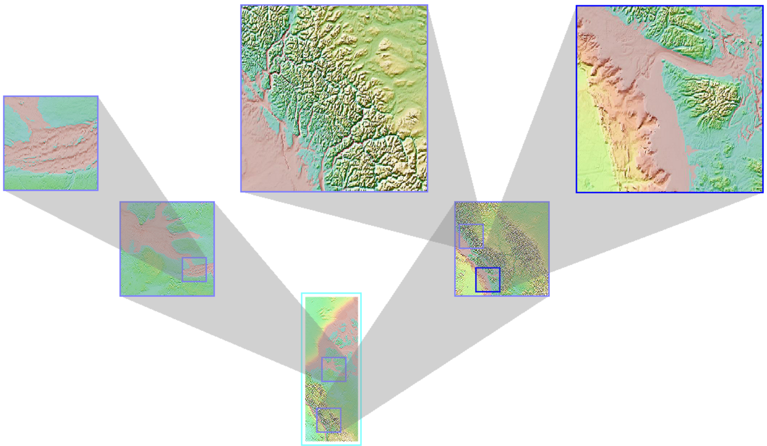
To facilitate focus+context visualization and exploration of a 3D image, our prototype currently allows the query windows identifying the ROIs to move back and forth through sequential slices interactively by the use of mouse scroll wheel and alternatively, the up and down arrow keys on the keyboard. When the query windows move from one slice to another, the low-resolution approximation of the context and the high-resolution approximations of the ROIs are updated on the fly in real-time. For 3D images, currently it only performs widthwise and heightwise decompositions, which keeps the number of 2D slices intact for depthwise volume exploration.

6.2. Experimental Results

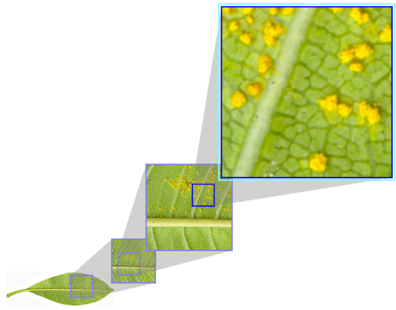
Here we present the experimental results produced by our *Focus+Context Studio* prototype. The n-tuples ($n \geq 3$) used in the captions of Figures 12, 13 and 14 are defined as follows: (image dimensions, $C_d, F^{r_1}, F^{r_2}, \ldots, F^{r_m}$), where d is the number of levels of (widthwise and heightwise) decomposition for the context and r_i ($1 \leq i \leq m$) is the number of levels of reconstruction for deriving the high-resolution approximation of the ith ROI. F^{r_i} appears in the n-tuple in a position determined by the left-to-right and top-to-bottom ordering of placement for the high-resolutions approximations of the ROIs.

Figure 12 shows various scenarios for focus+context visualization of 2D images using our prototype. Figures 12(a) and 12(b) show multilevel focus+context visualization of large-scale 2D images showing the topographic and bathymetric shading of northwestern North America and the topographic shading of Long Island, respectively. Similar multilevel focus+context visualization is shown for a diseased leaf in Figure 12(c). Such multilevel focus+context visualization is motivated by the need for more manageable utilization of screen space and visualization of the context at a higher resolution while maintaining interactive frame rates.

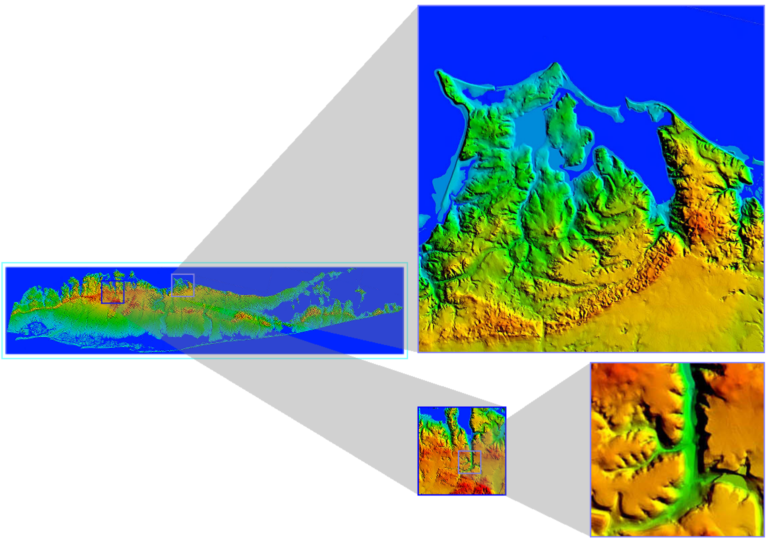
Next, for a 2D image, Figures 12(d) and 12(e) show different levels of decomposition for the context and different levels of reconstruction for the high-resolution approximation of the ROI using our developed tool. The 2D image used in this example is an abdomen slice from the male from the Visible Human Project [NLM12]. One advantage of allowing multiple query windows corresponding to multiple ROIs is the ability to draw comparisons between similar ROIs when required. Figure 12(f) shows such a comparison scenario between the tropical storm Parma (on the left) and typhoon Melor (on the right). Another such scenario comparing the ice near the coasts of Greenland and Alexander Island is shown in Figure 12(g).



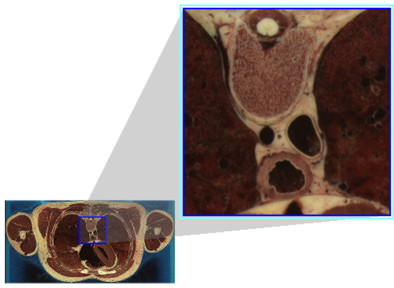
(a) Topographic and bathymetric shading of northwestern North America: $(4672{\times}12800, C_6, F^4, F^5, F^5, F^2, F^2)$.



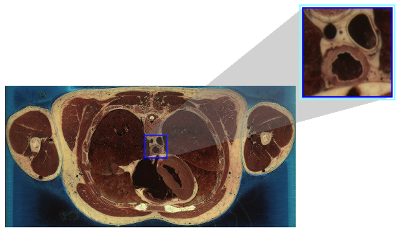
(c) Frangipani rust: $(10496{\times}3328, C_6, F^6, F^3, F^1)$.



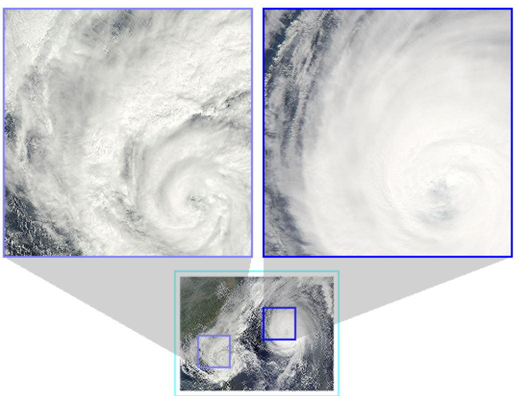
(b) Topographic shading of Long Island: $(18944{\times}4224, C_5, F^4, F^2, F^3)$.



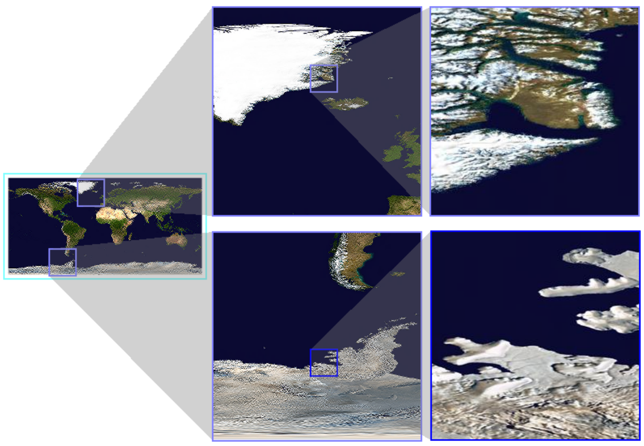
(d) Male abdomen: $(1728{\times}832, C_3, F^3)$.



(e) Male abdomen: $(1728{\times}832, C_2, F^2)$



(f) Comparison of Parma (on the left) and Melor (on the right): $(10240{\times}7680, C_6, F^3, F^3)$.



(g) Comparison of ice near the coasts of Greenland and Alexander Island on a Blue Marble image: $(15360{\times}7680, C_6, F^3, F^3, F^3, F^3)$.

Figure 12: Focus+context visualization of 2D images at various resolutions.

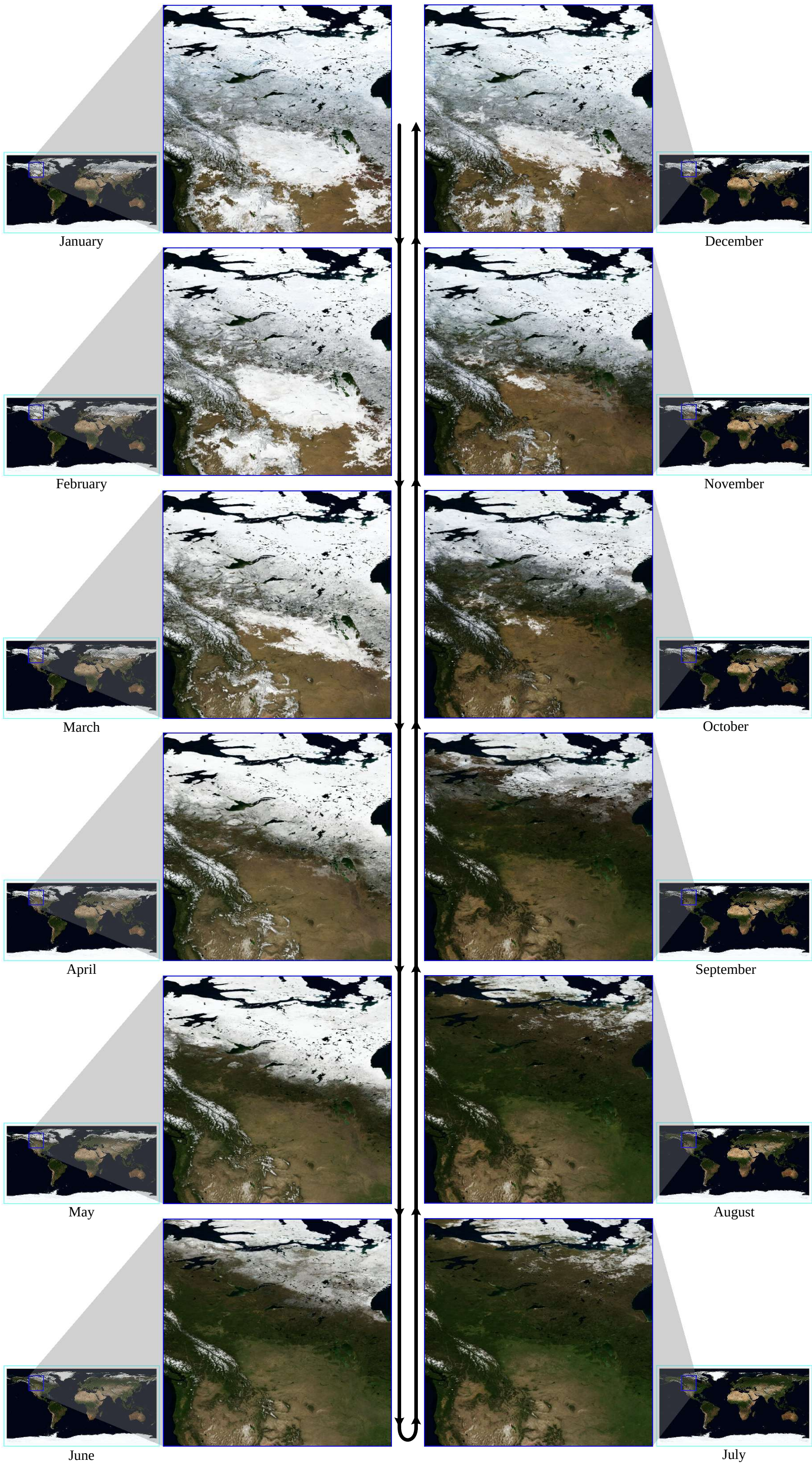


Figure 13: Focus+context visualization of time-lapse imagery – monthly global images: $(5440 \times 2752 \times 12, C_4, F^4)$.

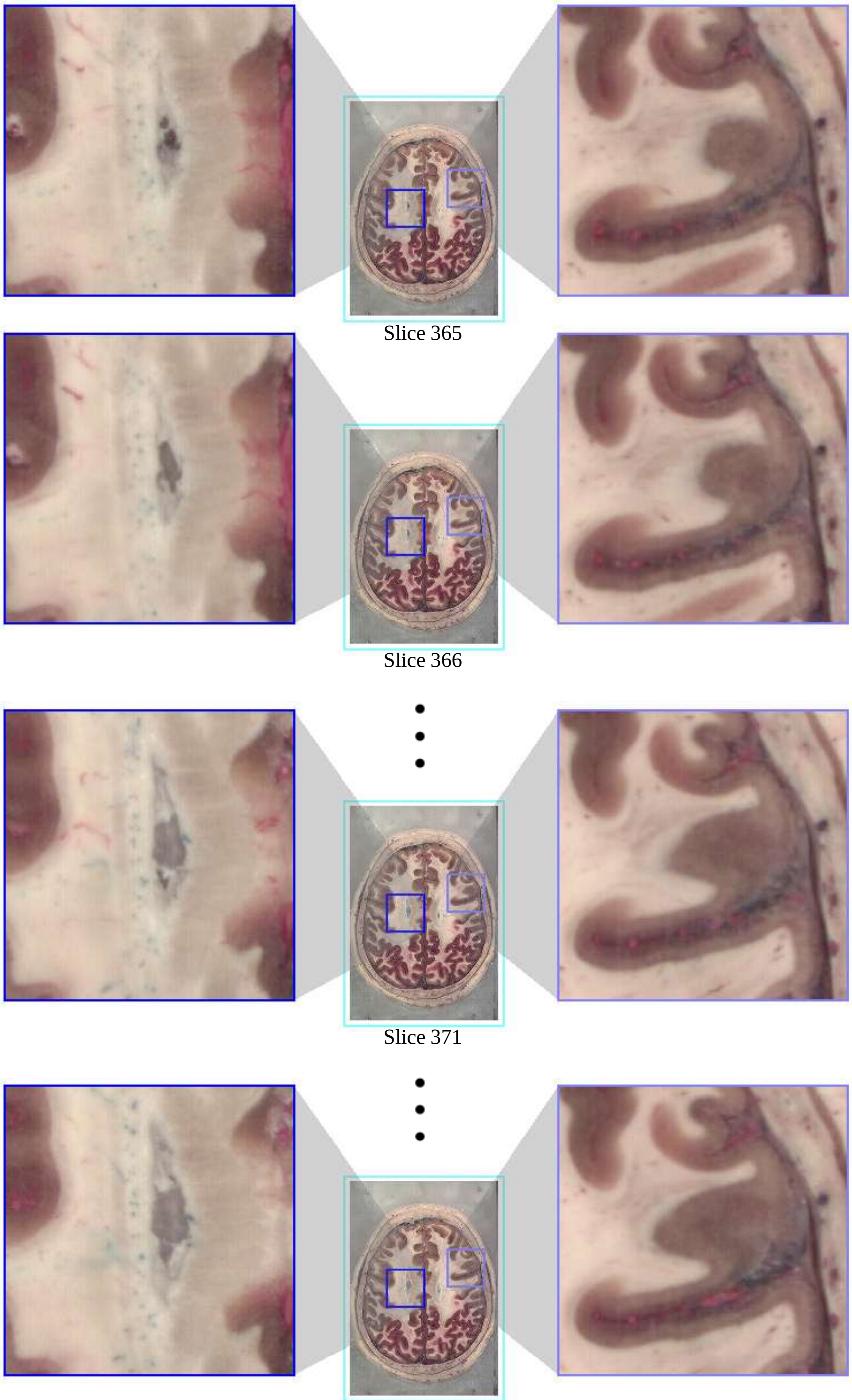


Figure 14: Focus+context visualization of a 3D image ($1056\times1528\times150, C_3, F^3, F^3$).

Our developed prototype is also suitable for the visualization and exploration of time-lapse imagery. For instance, Figure 13 shows 12 unique frames from the interactive transition through the 12 slices of monthly global images. The order of frames is shown by directions marked on the curved-arrow in the middle. The ROI covers most of northwestern North America and shows the transition from one winter to the following winter.

Figure 14 shows an example of visualization and exploration of a 3D image in our prototype. For the purpose of demonstration, the transition through 10 of the 150 slices that the query windows were constrained to move back and forth through are shown in Figure 14. The original dataset corresponds to the head of the female from the Visible Human Project [NLM12]. This head dataset contains a total of 1477 2D slices, each of dimensions 1056×1528, among which 150 sequential slices were loaded into our prototype for this example.

7. Discussion and Future Work

Our method can be used to devise a balanced multiresolution scheme for any set of given regular multiresolution filter vectors. However, if the scheme would only make use of regular reconstruction filters is determined by the properties of the given multiresolution filter vectors. If the given filter vectors are symmetric/antisymmetric, then our method can devise a balanced multiresolution scheme that only uses regular filters. Otherwise, some extraordinary boundary reconstruction filters are introduced (see Appendix B, for instance).

The balanced multiresolution schemes devised by our approach can also be applied to open curves and tensor product meshes (surfaces and volumes) in applications where boundary interpolation is not important but a balanced decomposition is preferred, for reasons such as partitioning the curve or the mesh into *even* and *odd* vertices. Such a partitioning allows the storage of coarse vertices and details in even and odd vertices, respectively, as proposed in [OSB07]. However, some of the devised balanced multiresolution schemes may support boundary interpolation only in the context of subdivision i.e. when we only consider the result of $\mathbf{P}C'$ in order to increase the resolution of C. For example, the filters of second order B-spline in (A.1) and the *short* filters of third order B-spline in (4) lead to such boundary interpolating subdivisions.

There is a number of directions for future research. In this article, we covered the commonly used types of symmetric and antisymmetric extensions. It would be useful to investigate and develop extension types that can be utilized to devise balanced multiresolution schemes for near symmetric and asymmetric filter vectors in order to ensure a perfect reconstruction solely by the use of regular filters. To start with, the devised balanced multiresolution scheme given in Appendix B for Daubechies' asymmetric D4 filters may provide some insights.

In addition, further investigations are needed for an in-depth understanding of the relations between the symmetry/antisymmetry exhibited by the filter vectors, parity of their widths, and the determined types of symmetric/antisymmetric extensions required for a perfect reconstruction using only regular filters. For instance, compare the multiresolution filter vectors containing the *inverse powers of two* filters of fourth order B-spline in (18) and the *wide* and *optimal* filters of fourth order B-spline in (A.3). In these two sets, the corresponding filter vectors have the same widths and they are all symmetric. Now, observe that the two balanced multiresolution schemes we devised using these two sets of filter vectors suggest exactly the same type of symmetric extensions for the column vectors of fine, coarse, and detail samples. Therefore, the determined types of symmetric/antisymmetric extensions are not dependant on actual filter values. Several other such scenarios are shown in Table A.2.

8. Conclusion

In this article, we presented a novel method for devising a *balanced multiresolution* scheme, primarily applicable to images, using a given a set of symmetric/antisymmetric filter vectors containing regular multiresolution filters. A balanced multiresolution scheme resulting from our method allows balanced decomposition and subsequent perfect reconstruction of images without using any extraordinary boundary filters. This is achieved by the use of an appropriate combination of symmetric and antisymmetric extensions at the image and detail boundaries, correlating to implicit sample split operations. Balanced decomposition of an image makes locating the details corresponding to a ROI straightforward in a hierarchy of details within the wavelet transform representation of that image.

In order to support smooth multiresolution representations of images beyond Haar wavelets and the associated scaling functions, and still exploit the advantages of a balanced decomposition, we used our method to devise balanced multiresolution schemes for some commonly used sets of local multiresolution filters obtained from higher order scaling functions and their wavelets. Any such balanced multiresolution scheme can be used to generate a *balanced wavelet transform* representation of a multidimensional image in a preprocessing phase, which can then be utilized to support its focus+context visualization in an efficient manner.

We also presented a set of experimental results produced using our developed *Focus+Context Studio* prototype that allows focus+context visualization of large-scale 2D and 3D images. It exploits the balanced multiresolution scheme we devised from the *short* filters of quadratic B-spline in (4). We envision the integration of the key functionalities of our prototype in visualization systems and application programming interfaces (APIs) to enable users to visualize and explore the contents of complex imagery such as large-scale satellite images, clinical data, seismic data, etc.

Acknowledgements

This research was supported by the *Natural Sciences and Engineering Research Council* (NSERC) of Canada, *Alberta Innovates – Technology Futures* (AITF), *Alberta Enterprise and Advanced Education*, *Networks of Centres of Excellence* (NCE) of Canada in *Graphics, Animation and New Media* (GRAND), and the *NSERC/AITF/Foundation CMG Industrial Research Chair* program in *Scalable Reservoir Visualization*. We would like to thank Javad Sadeghi for providing the multiresolution filter vectors containing the local filters of second order B-spline in (A.1) and Jennifer E. Fairman for providing the traditional medical illustration shown in Figure 3(a). We would also like to thank Troy Alderson for his helpful editorial comments. The following list includes sources for the datasets we used in producing the experimental results: topographic and bathymetric shading of northwestern North America [SSB08], topographic shading of Long Island [Han02], diseased leaf [FS10], Visible Human dataset [NLM12], Parma and Melor [MRR09], Blue Marble image [VEN02], and monthly global images [Sto04].

Appendix A. Further Examples with Symmetric/Antisymmetric Filter Vectors

Our first example here involves the multiresolution filter vectors containing the local regular filters of second order B-spline,

$$\begin{cases} \mathbf{a} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{3} & \frac{2}{3} & \frac{1}{3} & -\frac{1}{6} \end{bmatrix}, \\ \mathbf{b} = \begin{bmatrix} -\frac{1}{2} & 1 & -\frac{1}{2} \end{bmatrix}, \\ \mathbf{p} = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}, \\ \mathbf{q} = \begin{bmatrix} -\frac{1}{6} & -\frac{1}{3} & \frac{2}{3} & -\frac{1}{3} & -\frac{1}{6} \end{bmatrix}, \end{cases} \tag{A.1}$$

derived by Sadeghi [Sad11] by reversing Faber subdivision [Fab09] based on the construction procedure presented by Samavati and Bartels in [SB99, BS00]. For the filter vectors in (A.1), the matrix equations for a balanced multiresolution scheme we devised using our method for $n \in \mathbb{Z}^+$ are

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} -\frac{1}{6} & \frac{1}{3} & \frac{2}{3} & \frac{1}{3} & -\frac{1}{6} & 0 & 0 & 0 & \cdots \\ 0 & 0 & -\frac{1}{6} & \frac{1}{3} & \frac{2}{3} & \frac{1}{3} & -\frac{1}{6} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \\ f_{2n-2} \end{bmatrix},$$

$$\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 1 & -\frac{1}{2} & 0 & 0 & 0 & \cdots \\ 0 & 0 & -\frac{1}{2} & 1 & -\frac{1}{2} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix},$$

and

$$
\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & \vdots & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} + \begin{bmatrix} -\frac{1}{6} & \frac{2}{3} & -\frac{1}{6} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{1}{3} & -\frac{1}{3} & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{1}{6} & \frac{2}{3} & -\frac{1}{6} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\frac{1}{6} & \frac{2}{3} & -\frac{1}{6} \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{1}{3} & -\frac{1}{3} \end{bmatrix} \begin{bmatrix} d_2 \\ d_1 \\ d_2 \\ \vdots \\ d_n \\ d_n \end{bmatrix},
$$

analogous to equations (1), (2), and (3), respectively.

The next example involves the following multiresolution filter vectors containing the local regular filters of cubic (fourth order) B-spline from [SBO07]:

$$
\begin{cases} \mathbf{a} = \begin{bmatrix} -\frac{1}{2} & 2 & -\frac{1}{2} \end{bmatrix}, \\ \mathbf{b} = \begin{bmatrix} \frac{1}{4} & -1 & \frac{3}{2} & -1 & \frac{1}{4} \end{bmatrix}, \\ \mathbf{p} = \begin{bmatrix} \frac{1}{8} & \frac{1}{2} & \frac{3}{4} & \frac{1}{2} & \frac{1}{8} \end{bmatrix}, \\ \mathbf{q} = \begin{bmatrix} \frac{1}{4} & 1 & \frac{1}{4} \end{bmatrix}. \end{cases} \tag{A.2}
$$

The filter vectors in (A.2) are called the *short* filters of cubic B-spline. For these filter vectors, the matrix equations for a balanced multiresolution scheme we devised using our method for $n \in \mathbb{Z}^+$ are

$$
\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 2 & -\frac{1}{2} & 0 & 0 & 0 & \cdots \\ 0 & 0 & -\frac{1}{2} & 2 & -\frac{1}{2} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix},
$$

$$
\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} \frac{1}{4} & -1 & \frac{3}{2} & -1 & \frac{1}{4} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \frac{1}{4} & -1 & \frac{3}{2} & -1 & \frac{1}{4} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \\ f_{2n-2} \end{bmatrix},
$$

and

$$
\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix} = \begin{bmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 & 0 & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & 0 & 0 & 0 & \cdots & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} c_2 \\ c_1 \\ c_2 \\ \vdots \\ c_n \\ c_n \end{bmatrix} + \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \vdots & 0 & 1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix},
$$

analogous to equations (1), (2), and (3), respectively.

Our last example uses the following multiresolution filter vectors containing the local regular filters of cubic B-spline from [SBO07]:

$$\begin{cases} \mathbf{a} = \left[\tfrac{23}{196} \;\; -\tfrac{23}{49} \;\; \tfrac{9}{28} \;\; \tfrac{52}{49} \;\; \tfrac{9}{28} \;\; -\tfrac{23}{49} \;\; \tfrac{23}{196} \right], \\ \mathbf{b} = \left[\tfrac{13}{98} \;\; -\tfrac{26}{49} \;\; \tfrac{39}{49} \;\; -\tfrac{26}{49} \;\; \tfrac{13}{98} \right], \\ \mathbf{p} = \left[\tfrac{1}{8} \;\; \tfrac{1}{2} \;\; \tfrac{3}{4} \;\; \tfrac{1}{2} \;\; \tfrac{1}{8} \right], \\ \mathbf{q} = \left[-\tfrac{23}{208} \;\; -\tfrac{23}{52} \;\; -\tfrac{63}{208} \;\; 1 \;\; -\tfrac{63}{208} \;\; -\tfrac{23}{52} \;\; -\tfrac{23}{208} \right]. \end{cases} \tag{A.3}$$

The filter vectors in (A.3) are known as the *wide* and *optimal* filters of cubic B-spline. For these filter vectors, the matrix equations for a balanced multiresolution scheme we devised using our method for $n \in \mathbb{Z}^+$ are

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} \frac{23}{196} & -\frac{23}{49} & \frac{9}{28} & \frac{52}{49} & \frac{9}{28} & -\frac{23}{49} & \frac{23}{196} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \frac{23}{196} & -\frac{23}{49} & \frac{9}{28} & \frac{52}{49} & \frac{9}{28} & -\frac{23}{49} & \frac{23}{196} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_3 \\ f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \\ f_{2n-2} \\ f_{2n-3} \end{bmatrix},$$

$$\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} \frac{13}{98} & -\frac{26}{49} & \frac{39}{49} & -\frac{26}{49} & \frac{13}{98} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \frac{13}{98} & -\frac{26}{49} & \frac{39}{49} & -\frac{26}{49} & \frac{13}{98} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} f_3 \\ f_2 \\ f_1 \\ f_2 \\ \vdots \\ f_{2n-1} \\ f_{2n} \\ f_{2n-1} \end{bmatrix},$$

and

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{2n} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{bmatrix} \begin{bmatrix} c_1 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \\ c_{n-1} \end{bmatrix} + \begin{bmatrix} -\frac{23}{52} & 1 & -\frac{23}{52} & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ -\frac{23}{208} & -\frac{63}{208} & -\frac{63}{208} & -\frac{23}{208} & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -\frac{23}{208} & -\frac{63}{208} & -\frac{63}{208} & -\frac{23}{208} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{23}{52} & 1 & -\frac{23}{52} & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & -\frac{23}{208} & -\frac{63}{208} & -\frac{63}{208} & -\frac{23}{208} \end{bmatrix} \begin{bmatrix} d_2 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \\ d_n \\ d_{n-1} \end{bmatrix},$$

analogous to equations (1), (2), and (3), respectively.

In Table A.2, we summarize all the balanced multiresolution schemes presented in this article so far, in addition to six other sets of symmetric/antisymmetric regular multiresolution filters. The biorthogonal and reverse biorthogonal filters [CDF92, Dau92] we referred to in the table are available in MATLAB [MAT14]. $\mathbf{q}^R$ in the first column of the table denotes the reversed filter vector $\mathbf{q}$. The second through fifth columns specify the symmetric (S)/antisymmetric (A) structure of the $\mathbf{a}$, $\mathbf{b}$, $\mathbf{p}$, and $\mathbf{q}$ filter vectors, respectively, for each set of filters in the table. The next column mentions the parity of w_a and w_b, based on which we decide on the type of symmetric extension to use for F.

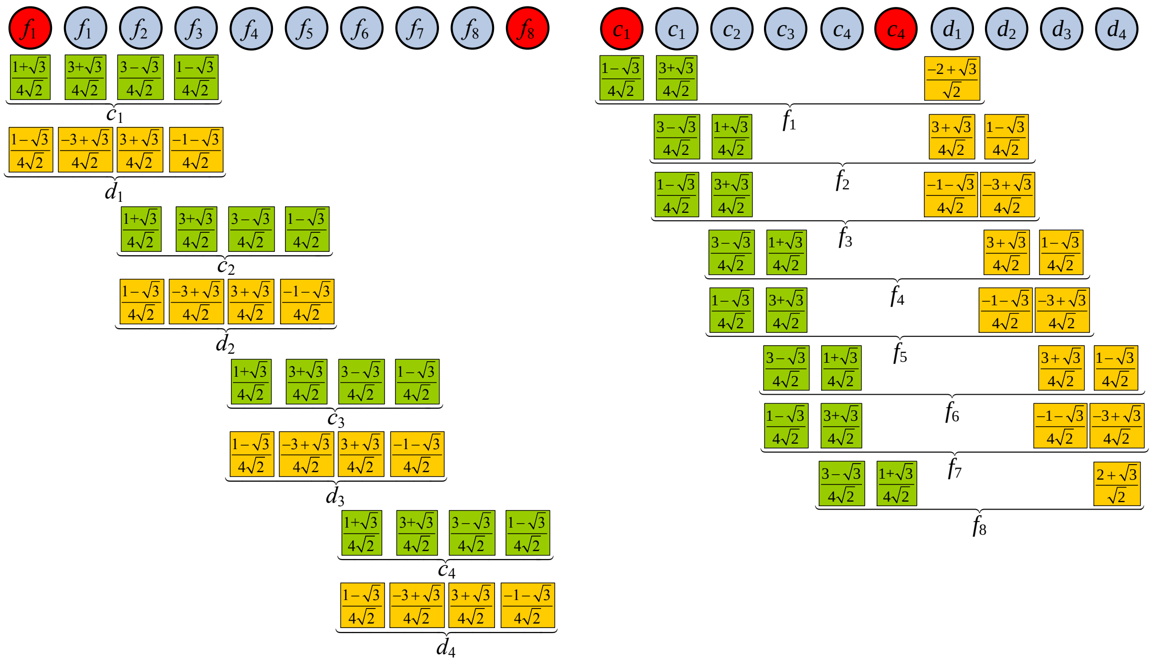
For $n \in \mathbb{Z}^+$, the second-to-last column of Table A.2 illustrates the proposed extended vector of fine sample F' and the construction of the first coarse sample c_1 and detail sample d_1, applicable to one possible balanced multiresolution scheme for each set of filters in the table. Here, we only give the construction of c_1 and d_1 because the remaining pairs of coarse and detail samples can be obtained by subsequent shifts of the filter vectors **a** and **b** by two fine samples along F' (as shown in Figure 9, for example). Finally, the last column shows the corresponding extended vectors of coarse samples C' and detail samples D', in addition to the reconstruction of the fine samples in F_l and F_r as defined in subsection 5.2. In this column, filter vectors of odd and even width are assumed to have formats similar to $\left[\ \dots\ v_{-2}\ \ v_{-1}\ \ v_0\ \ v_1\ \ v_2\ \ \dots\ \right]$ and $\left[\ \dots\ v_{-2}\ \ v_{-1}\ \ v_1\ \ v_2\ \ \dots\ \right]$, respectively.

Appendix B. An Example with Asymmetric Filter Vectors

An attempt to apply our general approach for devising a balanced multiresolution scheme described in section 5 to Daubechies' asymmetric D4 filters [Dau88, SDS96],

$$\begin{cases} \mathbf{a} \ = \ \mathbf{p} \ = \ \left[\ \frac{1+\sqrt{3}}{4\sqrt{2}} \quad \frac{3+\sqrt{3}}{4\sqrt{2}} \quad \frac{3-\sqrt{3}}{4\sqrt{2}} \quad \frac{1-\sqrt{3}}{4\sqrt{2}}\ \right], \\ \mathbf{b} \ = \ \mathbf{q} \ = \ \left[\ \frac{1-\sqrt{3}}{4\sqrt{2}} \quad \frac{-3+\sqrt{3}}{4\sqrt{2}} \quad \frac{3+\sqrt{3}}{4\sqrt{2}} \quad \frac{-1-\sqrt{3}}{4\sqrt{2}}\ \right], \end{cases} \tag{B.1}$$

produces the balanced multiresolution setup shown in Figure B.15. Note that it introduces two extraordinary boundary filter values, $\frac{-2+\sqrt{3}}{\sqrt{2}}$ and $\frac{2+\sqrt{3}}{\sqrt{2}}$ in the reconstruction of f_1 and f_8, respectively. Because the filter vectors in (B.1) are not symmetric/antisymmetric, the rewriting task suggested in step 2(c) and that of step 3 in our *general construction* given in subsection 5.2 were not entirely successful. Therefore, our approach could not ensure a perfect reconstruction using only the regular filters from (B.1).



(a) Balanced decomposition.

(b) Perfect reconstruction.

Figure B.15: Balanced decomposition and perfect reconstruction of 8 fine samples.

As shown in Figure B.15(b), this particular example does not require any extraordinary boundary filters for the subdivision matrix **P**. This may not always be the case while using other asymmetric filter vectors.

Table A.2: Balanced multiresolution schemes.

Filters	**a**	**b**	**p**	**q**	$w_{\mathrm{a}}, w_{\mathrm{b}}$	**Decomposition:** F', c_1, d_1	**Reconstruction:** $C', D', f_i \in \left[{F_l}^T \; {F_r}^T\right]^T$
Filters of second order B-spline (A.1) Biorthogonal 2.2 filters (**a**, **b**, **p**, and **q** in [MAT14])	S	S	S	S	Odd	$F' = [f_2 \; f_1 \; \ldots \; f_{2n} \; f_{2n-1} \; f_{2n-2}]^T,$ $\begin{cases} c_1 &= a_{-2}f_2 + a_{-1}f_1 + a_0 f_2 \\ &\quad + a_1 f_3 + a_2 f_4, \\ d_1 &= b_{-1}f_2 + b_0 f_1 + b_1 f_2. \end{cases}$	$C' = [c_1 \; c_1 \; \ldots \; c_n]^T, \quad D' = [d_2 \; d_1 \; \ldots \; d_n \; d_n]^T,$ $\begin{cases} f_1 &= p_1 c_1 + p_{-1}c_1 + q_2 d_2 + q_0 d_1 + q_{-2}d_2, \\ f_{2n-1} &= p_1 c_{n-1} + p_{-1}c_n + q_2 d_{n-1} + q_0 d_n + q_{-2}d_n, \\ f_{2n} &= p_0 c_n + q_1 d_n + q_{-1}d_n. \end{cases}$
Short filters of quadratic B-spline (4) Biorthogonal 3.1 filters (**a**, **b**, **p**, and $\mathbf{q}^R$ in [MAT14]) Reverse biorthogonal 3.1 filters (**a**, **b**, **p**, and $\mathbf{q}^R$ in [MAT14])	S	A	S	A	Even	$F' = [f_1 \; f_1 \; \ldots \; f_{2n} \; f_{2n}]^T,$ $\begin{cases} c_1 &= a_{-2}f_1 + a_{-1}f_1 + a_1 f_2 + a_2 f_3, \\ d_1 &= b_{-2}f_1 + b_{-1}f_1 + b_1 f_2 + b_2 f_3. \end{cases}$	$C' = [c_1 \; c_1 \; \ldots \; c_n \; c_n]^T, \quad D' = [-d_1 \; d_1 \; \ldots \; d_n \; -d_n]^T,$ $\begin{cases} f_1 &= p_2 c_1 + p_{-1}c_1 + q_2(-d_1) + q_{-1}d_1, \\ f_{2n} &= p_1 c_n + p_{-2}c_n + q_1 d_n + q_{-2}(-d_n). \end{cases}$
Wide filters of quadratic B-spline (**a**, **b**, **p**, and **q** in [SBO07]) Biorthogonal 3.3 filters (**a**, **b**, **p**, and $\mathbf{q}^R$ in [MAT14])	S	A	S	A	Even	$F' = [f_3 \; f_2 \; f_1 \; f_1 \; \ldots \; f_{2n} \; f_{2n} \; f_{2n-1} \; f_{2n-2}]^T,$ $\begin{cases} c_1 &= a_{-4}f_3 + a_{-3}f_2 + a_{-2}f_1 + a_{-1}f_1 \\ &\quad + a_1 f_2 + a_2 f_3 + a_3 f_4 + d_4 f_5, \\ d_1 &= b_{-2}f_1 + b_{-1}f_1 + b_1 f_2 + b_2 f_3. \end{cases}$	$C' = [c_1 \; c_1 \; \ldots \; c_n \; c_n]^T, \quad D' = [-d_2 \; -d_1 \; d_1 \; \ldots \; d_n \; -d_n \; -d_{n-1}]^T,$ $\begin{cases} f_1 &= p_2 c_1 + p_{-1}c_1 + q_4(-d_2) + q_2(-d_1) + q_{-1}d_1 + q_{-3}d_2, \\ f_2 &= p_1 c_1 + p_{-2}c_2 + q_3(-d_1) + q_1 d_1 + q_{-2}d_2 + q_{-4}d_3, \\ f_3 &= p_2 c_1 + p_{-1}c_2 + q_4(-d_1) + q_2 d_1 + q_{-1}d_2 + q_{-3}d_3, \\ f_{2n-2} &= p_1 c_{n-1} + p_{-2}c_n + q_3 d_{n-2} + q_1 d_{n-1} + q_{-2}d_n + q_{-4}(-d_n), \\ f_{2n-1} &= p_2 c_{n-1} + p_{-1}c_n + q_4 d_{n-2} + q_2 d_{n-1} + q_{-1}d_n + q_{-3}(-d_n), \\ f_{2n} &= p_1 c_n + p_{-2}c_n + q_3 d_{n-1} + q_1 d_n + q_{-2}-d_n + q_{-4}(-d_{n-1}). \end{cases}$
Short filters of cubic B-spline (A.2) Reverse biorthogonal 2.2 filters (**a**, **b**, **p**, and **q** in [MAT14])	S	S	S	S	Odd	$F' = [f_2 \; f_1 \; \ldots \; f_{2n} \; f_{2n-1} \; f_{2n-2}]^T,$ $\begin{cases} c_1 &= a_{-1}f_2 + a_0 f_1 + a_1 f_2, \\ d_1 &= b_{-2}f_2 + b_{-1}f_1 + b_0 f_2 \\ &\quad + b_1 f_3 + b_2 f_4. \end{cases}$	$C' = [c_2 \; c_1 \; \ldots \; c_n \; c_n]^T, \quad D' = [d_1 \; d_1 \; \ldots \; d_n]^T,$ $\begin{cases} f_1 &= p_2 c_2 + p_0 c_1 + p_{-2}c_2 + q_1 d_1 + q_{-1}d_1, \\ f_{2n-1} &= p_2 c_{n-1} + p_0 c_n + p_{-2}c_n + q_1 d_{n-1} + q_{-1}d_n, \\ f_{2n} &= p_1 c_n + p_{-1}c_n + q_0 d_n. \end{cases}$
Inverse powers of two filters of cubic B-spline (18) *Wide* and *optimal* filters of cubic B-spline (A.3)	S	S	S	S	Odd	$F' = [f_3 \; f_2 \; f_1 \; \ldots \; f_{2n} \; f_{2n-1} \; f_{2n-2} \; f_{2n-3}]^T,$ $\begin{cases} c_1 &= a_{-3}f_3 + a_{-2}f_2 + a_{-1}f_1 + a_0 f_2 \\ &\quad + a_1 f_3 + a_2 f_4 + a_3 f_5, \\ d_1 &= b_{-2}f_3 + b_{-1}f_2 + b_0 f_1 \\ &\quad + b_1 f_2 + b_2 f_3. \end{cases}$	$C' = [c_1 \; c_1 \; \ldots \; c_n \; c_{n-1}]^T, \quad D' = [d_2 \; d_1 \; \ldots \; d_n \; d_n \; d_{n-1}]^T,$ $\begin{cases} f_1 &= p_1 c_1 + p_{-1}c_1 + q_2 d_2 + q_0 d_1 + q_{-2}d_2, \\ f_2 &= p_2 c_1 + p_0 c_1 + p_{-2}c_2 + q_3 d_2 + q_1 d_1 + q_{-1}d_2 + q_{-3}d_3, \\ f_{2n-2} &= p_2 c_{n-2} + p_0 c_{n-1} + p_{-2}c_n + q_3 d_{n-2} + q_1 d_{n-1} + q_{-1}d_n + q_{-3}d_n, \\ f_{2n-1} &= p_1 c_{n-1} + p_{-1}c_n + q_2 d_{n-1} + q_0 d_n + q_{-2}d_n, \\ f_{2n} &= p_2 c_{n-1} + p_0 c_n + p_{-2}c_{n-1} + q_3 d_{n-1} + q_1 d_n + q_{-1}d_n + q_{-3}d_{n-1}. \end{cases}$

References

[AW03] Adams M. D., Ward R. K.: Symmetric-extension-compatible reversible integer-to-integer wavelet transforms. *IEEE Transactions on Signal Processing 51*, 10 (Oct 2003), 2624–2636.

[BGS06] Bartels R. H., Golub G. H., Samavati F. F.: Some observations on local least squares. *BIT Numerical Mathematics 46*, 3 (Sep 2006), 455–477.

[BS00] Bartels R. H., Samavati F. F.: Reversing subdivision rules: local linear conditions and observations on inner products. *Journal of Computational and Applied Mathematics 119* (Jul 2000), 29–67.

[BS11] Bartels R., Samavati F.: Multiresolutions numerically from subdivisions. *Computers & Graphics 35*, 2 (2011), 185–197.

[CDF92] Cohen A., Daubechies I., Feauveau J.-C.: Biorthogonal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics 45*, 5 (1992), 485–560.

[Cha74] Chaikin G. M.: An algorithm for high-speed curve generation. *Computer Graphics and Image Processing 3*, 4 (1974), 346–349.

[Dau88] Daubechies I.: Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics 41*, 7 (1988), 909–996.

[Dau92] Daubechies I.: *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, USA, 1992.

[Fab09] Faber G.: Über stetige Funktionen. *Mathematische Annalen 66* (1909), 81–94.

[FB88] Forsey D. R., Bartels R. H.: Hierarchical B-spline refinement. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1988), SIGGRAPH '88, ACM, pp. 205–212.

[FS10] Fraser-Smith S.: Frangipani rust on Plumeria rubra. `http://en.wikipedia.org/wiki/File:Frangipani_rust_(caused_by_Coleosporium_plumeriae)_on_Plumeria_rubra.jpg`, Mar 16 2010.

[GC95] Gortler S. J., Cohen M. F.: Hierarchical and variational geometric modeling with wavelets. In *Proceedings of the 1995 Symposium on Interactive 3D graphics* (Monterey, CA, Apr 9 1995), I3D '95, ACM, pp. 35–42.

[Haa10] Haar A.: Zur Theorie der orthogonalen Funktionensysteme. *Mathematische Annalen 69* (1910), 331–371.

[Han02] Hanson G.: Digital Elevation Model (DEM) of Long Island shaded to emphasize topographic highs. `http://people.hofstra.edu/J_B_Bennington/research/long_island/LONG_ISLAND_GH_lg.jpg`, Nov 18 2002.

[HMC11] Hsu W.-H., Ma K.-L., Correa C.: A rendering framework for multiscale views of 3D models. In *Proceedings of the 2011 SIGGRAPH Asia Conference* (New York, NY, USA, 2011), SA '11, ACM, pp. 131:1–131:10.

[Hod03] Hodges E. R. S.: *The Guild handbook of scientific illustration*. John Wiley and Sons, Hoboken, NJ, USA, 2003.

[KNI94] Kiya H., Nishikawa K., Iwahashi M.: A development of symmetric extension method for subband image coding. *IEEE Transactions on Image Processing 3*, 1 (Jan 1994), 78–81.

[KZT02] Kharitonenko I., Zhang X., Twelves S.: A wavelet transform with point-symmetric extension at tile boundaries. *IEEE Transactions on Image Processing 11*, 12 (Dec 2002), 1357–1364.

[LHJ99] LaMar E., Hamann B., Joy K. I.: Multiresolution techniques for interactive texture-based volume visualization. In *Proceedings of the 1999 conference on Visualization* (Los Alamitos, CA, USA, 1999), VIS '99, IEEE Computer Society Press, pp. 355–361.

[LL00] Li S., Li W.: Shape-adaptive discrete wavelet transforms for arbitrarily shaped visual object coding. *IEEE Transactions on Circuits and Systems for Video Technology 10*, 5 (Aug 2000), 725–743.

[LS08] Lin J., Smith M. J. T.: New perspectives and improvements on the symmetric extension filter bank for subband/wavelet image compression. *IEEE Transactions on Image Processing 17*, 2 (Feb 2008), 177–189.

[MAT14] MATLAB and Wavelet Toolbox Release 2014a. The MathWorks, Inc., Natick, MA, USA, 2014.

[Mey90] Meyer Y.: *Ondelettes et Opérateurs*. Hermann, 1990.

[MRR09] MODIS Rapid Response Team, NASA: Typhoon Melor and tropical storm Parma. `http://eoimages.gsfc.nasa.gov/images/imagerecords/40000/40615/ParmaMelor_AMO_TMO_2009279_lrg.jpg`, Oct 7 2009.

[NLM12] U.S. National Library of Medicine: The Visible Human Project. `http://www.nlm.nih.gov/research/visible/visible_human.html`, Feb 2012.

[OSB07] Olsen L., Samavati F., Bartels R.: Multiresolution for curves and surfaces based on constraining wavelets. *Computers & Graphics 31*, 3 (2007), 449–462.

[PHF07] Plate J., Holtkaemper T., Froehlich B.: A flexible multi-volume shader framework for arbitrarily intersecting multi-resolution datasets. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (Nov-Dec 2007), 1584–1591.

[PTCF02] Plate J., Tirtasana M., Carmona R., Fröhlich B.: Octreemizer: a hierarchical approach for interactive roaming through very large volumes. In *Proceedings of the symposium on Data Visualisation 2002* (Aire-la-Ville, Switzerland, 2002), VISSYM '02, Eurographics Association, pp. 53–60.

[Sad11] Sadeghi J.: Reversing Faber subdivision. Personal communication, 2011.

[SB99] Samavati F. F., Bartels R. H.: Multiresolution curve and surface representation: reversing subdivision rules by least-squares data fitting. *Computer Graphics Forum 18*, 2 (1999), 97–119.

[SB04] Samavati F. F., Bartels R. H.: Local filters of B-spline wavelets. In *Proceedings of the 2004 Intlernational Workshop on Biometric Technologies* (2004), BT '04, pp. 105–110.

[SBO07] Samavati F. F., Bartels R. H., Olsen L.: Local B-spline multiresolution with examples in iris synthesis and volumetric rendering. In *Image Pattern Recognition: Synthesis and Analysis in Biometrics*, Yanushkevich S. N., Gavrilova M. L., Wan P. S. P., Srihari S. N., (Eds.), vol. 67 of *Series in Machine Perception and Artificial Intelligence*. World Scientific Publishing, 2007, pp. 65–102.

[SDS96] Stollnitz E. J., Derose T. D., Salesin D. H.: *Wavelets for computer graphics: theory and applications*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1996.

[SGM*11] Suter S., Guitian J. I., Marton F., Agus M., Elsener A., Zollikofer C., Gopi M., Gobbetti E., Pajarola R.: Interactive multiscale tensor reconstruction for multiresolution volume visualization. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (Dec 2011), 2135–2143.

[SSB08] Sandwell D. T., Smith W. H. F., Becker J. J.: Topographic and bathymetric shading of tile w140n90. http://topex.ucsd.edu/srtm30_plus/images/w140n90.Bathymetry.srtm.jpg, 2008.

[Sto04] Stöckli R.: Monthly global images, NASA. http://earthobservatory.nasa.gov/Features/BlueMarble/BlueMarble_monthlies.php, 2004.

[TSS*06] Taerum T., Sousa M. C., Samavati F. F., Chan S., Mitchell J. R.: Real-time super resolution contextual close-up of clinical volumetric data. In *Proceedings of the Joint Eurographics - IEEE VGTC Symposium on Visualization (EuroVis06)* (May 8-10 2006), Eurographics Association, pp. 347–354.

[VEN02] Visible Earth, NASA: Blue Marble: land surface, shallow water, and shaded topography. http://en.m.wikipedia.org/wiki/File:Whole_world_-_land_and_oceans.jpg, Feb 11 2002.

[WS05] Wang C., Shen H.-W.: Hierarchical navigation interface: leveraging multiple coordinated views for level-of-detail multiresolution volume rendering of large scientific data sets. In *Proceedings of the Ninth International Conference on Information Visualisation* (Jul 2005), pp. 259–267.

[WWLM11] Wang Y.-S., Wang C., Lee T.-Y., Ma K.-L.: Feature-preserving volume data reduction and focus+context visualization. *IEEE Transactions on Visualization and Computer Graphics 17*, 2 (Feb 2011), 171–181.

[ZSS97] Zorin D., Schröder P., Sweldens W.: Interactive multiresolution mesh editing. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., pp. 259–268.