Design and Implementation of Optimal Feature-Preserving (OFP) Wavelet Transforms for Computational Datasets

Gheorghe Craciun, Raghu Machiraju, *Member, IEEE,* and David Thompson, *Member, IEEE*

***Abstract*— In this interdisciplinary paper we develop a wavelet design framework emphasizing feature preservation. This work draws from results in signal processing, computational mathematics, and wavelet theory. We are particularly interested in linear wavelet transforms for large datasets generated by computational fluid dynamics simulations. High-fidelity wavelet transforms can facilitate the visualization and analysis of large scientific datasets. However, it is important that salient characteristics of the original features be preserved under the transformation. Our effort is different from classical filter design approaches which focus solely on performance in the frequency domain. In particular, we present a set of filter design axioms that ensure certain feature characteristics are preserved from scale to scale, and that the resulting filters correspond to wavelet transforms admitting in-place implementation. We also demonstrate how the axioms can be used to design optimal feature-preserving (OFP) filters, i.e., linear feature-preserving filters that are optimal in the sense that they are closest in L^2 to the ideal low pass filter. Results are included that demonstrate the feature-preservation characteristics of OFP filters.**

***Index Terms*— filter bank, wavelet design, lifting scheme, TVD transforms, feature preservation**

1. Introduction

Large-scale computational fluid dynamics simulations of physical phenomena produce data of unprecedented size (terabyte and petabyte range). Unfortunately, development of appropriate data management, analysis and visualization techniques has not kept pace with the growth in size and complexity of such datasets. One paradigm of large-scale visualization and analysis is to browse regions containing significant features of the dataset while accessing only the data needed to reconstruct these regions. The cornerstone of an approach of this type is a representational scheme that facilitates ranked access to macroscopic features in the dataset [12], [13], [16]. In this approach, a feature-detection algorithm is used to identify and rank contextually significant features directly in the compressed domain.

In [12], [13], [16], the linear lifting scheme [19] was used for compressing components of a vector field. The work reported here grew out of our efforts to analyze the implementation of the lifting scheme and design new transforms that more ardently preserve features in discrete flow fields. The rate-distortion characteristics of many wavelet transforms do not bode well for feature preservation [16]. However, it was unclear as to what distortions the wavelet transform wrought on the data. It is therefore useful to evaluate the effect of the wavelet transform in terms of processes that alter certain characteristics of the data, i.e. features. Additionally, for very large datasets it is necessary that the feature detection be performed in the compressed domain. In this context, it is essential that the wavelet transform preserve significant features in the data set.

In this paper we define a framework for the analysis and design of multiscale feature-centric filters through a variational characterization. Given the need for efficient compression and processing, we consider only linear transforms at this time. What is unique about our approach is that we design the behavior of the filter in the spatial domain. We suggest that the methods proposed here can be used in conjunction with frequency-based methods to design multiscale linear wavelet filters. A result of our characterization is a set of axioms that can be used to analyze and design filters. Filters that satisfy our axioms will be more likely to preserve features in a linear wavelet space and enable high-fidelity feature detection in large scientific datasets. Additionally, we seek to design filters corresponding to wavelets that can be implemented as a sequence of lifting steps [19].

A. Motivation

We now present a simple one-dimensional example from fluid dynamics to provide motivation for this effort. Shown in Figure 1 is a schematic of the shock tube problem. A shock tube can be idealized as a cylinder, closed at both ends, with a diaphragm that separates a region of gas on the left with pressure and density given by p_4 and ρ_4 respectively, from a region of gas on the right with pressure and density given by p_1 and ρ_1. Note that $p_4 > p_1$ and ρ_1 and ρ_4 must be specified. Initially, the gas is a rest in both regions. The diaphragm is then ruptured instantaneously and an unsteady motion ensues. In a typical situation, four uniform regions and one transitional region emerge. A normal shock wave propagating to the right defines the boundary between uniform Regions 1 and 2. The flow field properties exhibit a nonisentropic, discontinuous change across the moving normal shock. The boundary between Regions 2 and 3 is a contact discontinuity. Density and temperature change discontinuously across the contact discontinuity while pressure and velocity are unchanged. Region 3 is also a uniform region. Region 3 and

This work was partially funded by the National Science Foundation under the Large Data and Scientific Software Visualization Program (ACI-9982344), an NSF Early Career Award (ACI-9734483), and the Information Technology Research Program (ACS-00859669)

the uniform Region 4 are separated by an expansion fan in which the flow field properties vary isentropically. Analytical expressions can be used to define the properties in each region under the assumption of one-dimensional inviscid flow. A more complete description of this problem can be found in most compressible fluid dynamics textbooks (e.g., see [3]).

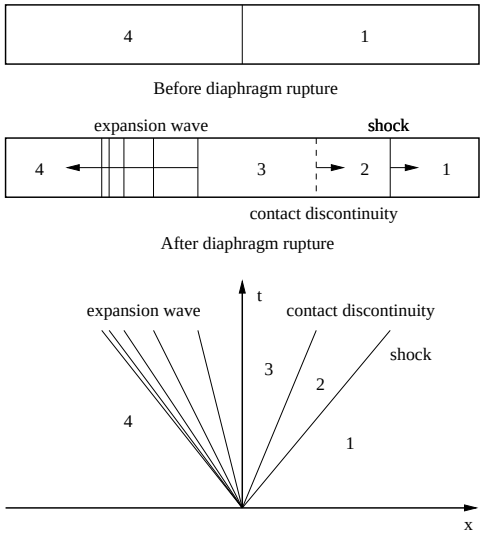


Fig. 1. One-Dimensional Shock Tube

Assuming we have a solution to the shock tube problem described above, we now want to visualize the data. In this case, it is not particularly challenging to locate the features in the flow field at a given time. However, for the sake of illustration, we assume that the data set is large and that we want to use a representation scheme that facilitates ranked access to features in the data set as discussed previously. As noted above, a significant component of the process is a feature detection performed using the compressed data. We choose as our wavelet transform the linear lifting scheme [19]. Figure 2 shows a sequence of figures illustrating the effects of applying the linear lifting scheme to the density field of a shock tube solution at a given time. The figure on the top left shows the original data defined using 65 points. Each remaining figure corresponds to an application of the wavelet transform resulting in 33 points, 17 points, and then 9 points. Of particular importance in these figures is the fact that application of the wavelet transform introduces oscillatory behavior in the data. Clearly, these oscillations are unacceptable if a feature detection algorithm based on gradients is to be used. Further, partial reconstruction of the data may be in significant error due to these oscillations and the compression of the otherwise relatively smooth data may not be as efficient.

B. Related Work

It is well known that wavelets can efficiently approximate smooth data [6] and produce efficient compression schemes. To suitably preserve edges in scalar image fields, several linear and non-linear or data-dependent schemes have been proposed [5], [7], [8], [14], [23]. In particular, Zhou [23] utilizes Essentially Non-Oscillatory (ENO) reconstructions [10]

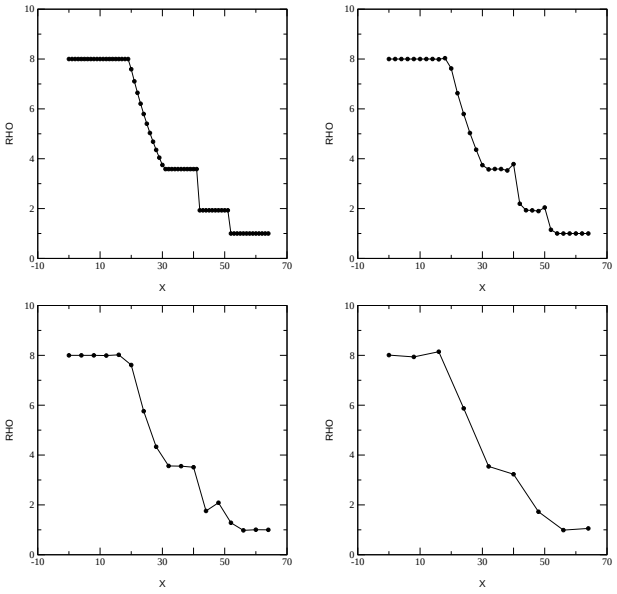


Fig. 2. Three Levels of Linear Lifting for Shock Tube Data

of the data so that fewer high frequency coefficients are created. Arandiga, Chiavassa, and Donat [4] use nonlinear, data dependent, ENO techniques for image compression, and show them to outperform classical algorithms for piecewise smooth images.

It is well known that lifting implementations of linear wavelets [19] allow for in-place computation and, in general, reduction by half of the computation time for the wavelet decomposition. In [20] it is shown that any biorthogonal wavelet transform can be factored into lifting steps.

Techniques employed in the study of partial differential equations (PDEs) have been extensively utilized to define the multiscale behavior of feature detection algorithms [1], [11], [17], [21] for images. Typically, the time variable in an evolutionary PDE is taken to represent a scale parameter. These techniques are used to enhance interregion boundaries and smooth intra-region variations. In vision and image processing applications, features of interest are usually defined by edges, i.e. localized regions of pixel gradients that can be thought of as discontinuities. It should be noted that linear PDEs are not completely successful in enhancing boundaries while eliminating noise. Discrete models of the diffusion equation with a nonlinear conductance based on gradient information have proven to be particularly useful for these applications [17], [21].

Our axiomatic filter design resembles the work of Weickert et al. [21] as well as that of Alvarez et al. [2]. Their frameworks are different since the domain of interest is limited to images populated with strong discontinuities such as edges. In our application, however, not all regions of strong gradients correspond to discontinuities. In fact, features with strong gradients such as expansions and boundary layers should not be treated as discontinuities.

C. Outline of the Paper

Our paper is structured as follows. In Section 2 we describe the general linear filter. In Section 3 we formalize our ideas regarding feature preservation including an analysis that defines constraints to be placed on the filter coefficients to ensure that new extrema are not created. In Section 4 we describe the lifting scheme, which allows for an efficient, in-place implementation of a biorthogonal wavelet transform. In Section 5 we present a set of filter design axioms based on the results in the previous two Sections. In Section 6, using these axioms, we design optimal feature preserving (OFP) filters: linear feature-centric filters that are optimal in the sense that they are closest in L^2 to the ideal low pass filter. Also, we describe how these filters may be employed in lifting implementations of feature-centric wavelets. Results are included in Section 7 to demonstrate the feature-preservation characteristics of our filters.

2. General Linear Filter

We now consider a general linear filter and characterize its behavior. We begin by defining a discrete, scalar quantity $s_{j,l}$ on an equally-spaced mesh $x_{j,l} = l\Delta x_j$ for $l = 0, ..., 2N$ with N being a positive integer. We seek a multiscale approximation to $s_{j,l}$ on a second equally-spaced mesh, $x_{j-1,l} = l\Delta x_{j-1}$ for $l = 0, ..., N$ with $\Delta x_{j-1} = 2\Delta x_j$, that preserves certain characteristics of the original scalar field. We denote this approximation as $s_{j-1,l}$.

We now consider a general linear filter of the form

$$s_{j-1,l} = \sum_{k=-m}^{+n} a_k s_{j,2l+k}, \tag{1}$$

where m and n are positive integers and the a_k are constants that are independent of the data. The a_k are composite coefficients that represent the combined effects of a wavelet transform implemented as a filter. The discrete moments of the filter are given by

$$\alpha_q = \sum_{k=-m}^{+n} k^q a_k \ . \tag{2}$$

The frequency response or amplification factor of the filter is given by

$$G(\beta) = \sum_{k=-m}^{+n} a_k e^{ik\beta} \tag{3}$$

where the amplification factor represents the response for the frequency β (here $i = \sqrt{-1}$). The magnitude of $G(\beta)$ measures the amplitude of a unit Fourier coefficient upon application of the filter and the phase of $G(\beta)$ measures the phase shift that occurs upon application of the filter.

3. Feature Preservation

It is now appropriate to define what we mean by feature preservation. In this context, feature preservation implies that the "location", "strength", and "shape" of features are unchanged after the application of the general filter (1). Of course, differences naturally occur due to the change in resolution between x_j and x_{j-1}. It should be noted that the behavior of the data in the spatial domain upon application of the general filter (1) is equivalent to the behavior of the solution of an evolutionary PDE in a scale space. However, we do not appeal to this approach in the current paper and rely, instead on more traditional analysis techniques. In a related vein, an analysis of spatial filters using Taylor series expansions was described in [15] where the filter performance was described in terms of spatial criteria by examining the non-zero discrete moments, i.e., the α_q in (2).

In the sections that follow, we formalize what we mean by feature preservation and develop conditions that the filter coefficients a_k must satisfy in order to preserve certain feature characteristics.

A. Feature Position

The "location" of a feature is simply its position within the domain. Clearly, it is undesirable to have a stationary feature that moves upon application of a filter. Similarly, a moving feature that is improperly translated is equally undesirable. This problem is simple to remedy, however. It is well known that unsymmetric filters, when applied to data, produce "shifts" in the data. On the other hand, if the filter is symmetric, no translation of the data occurs. For some unsymmetric filters, it is relatively straightforward to determine what shift occurs and to perform a translation of the entire dataset upon application of the filter. For others, this approach is not straightforward. The simplest approach is to consider only symmetric filters. The first condition for feature preservation we specify is that the general linear filter in equation (1) must be symmetric:

$$a_k = a_{-k} \quad \text{for all} \quad k \tag{$F1$}$$

B. Feature Strength

The "strength" of a feature can be described in terms of the changes in the data. For the strength to be preserved, the linear filter should not accentuate or diminish local extrema. This condition can be related to the frequency response (3) of the filter. To be effective, the filter must eliminate high frequency components of the data before sampling. Therefore, we expect the amplification factor to be zero at π, i.e., $G(\pi) = 0$. Further, we may want to specify the number of derivatives of the amplification factor that we want to be zero at π. This leads to the second constraint:

$$\sum_k (-1)^k k^j a_k = 0, \ \text{for } j = 0, 1, ..., p-1, \quad \text{and some } p \geq 1, \tag{$F2$}$$

where p is the number of zeros of $G(\beta)$ at $\beta = \pi$. To ensure that a constant field is not modified by application of the filter, we want the amplification factor at the lowest frequencies to be unity, i.e., $G(0) = 1$. Using (3), we can easily show that the amplification factor is one provided the coefficients satisfy the partition of unity. Therefore, the third condition for feature preservation we specify is that the coefficients of the general linear filter (1) must partition unity:

$$\sum_k a_k = 1 \tag{$F3$}$$

While the desired behaviors at zero and π are well understood, the behavior away from these extremes is not. The approach we have taken is to assume that the $sinc$ filter is the optimal filter for the intermediate ranges. Therefore, the fourth condition for feature preservation we specify is:

$(..., a_{-2}, a_{-1}, a_0, a_1, a_2, ...)$ minimizes the L^2 distance to the $sinc$ filter. $(F4)$

C. Feature Shape

On approach for describing the "shape" of a feature is in terms of regions of monotone variation in the data. In this context, "shape" preservation implies that the application of the linear filter should not introduce new extrema. This condition is expressed in image processing [21] as the "causality condition." This condition can be imposed by ensuring that the linear transform is Total Variation Diminishing (TVD) after the data is subsampled.

We now appeal to a concept from computational fluid dynamics to help define constraints on the a_k values. In modern CFD simulations, nonlinear techniques typically called "limiting" are used with some success to achieve higher-order temporal and spatial accuracy without spurious oscillations. Harten [9] and Yee [22] have made significant contributions to this field with their work on Total Variation Diminishing (TVD) algorithms. In our efforts, we focus purely on linear transforms to preserve efficient invertibility.

In a TVD algorithm, the total variation of the solution does not increase with time. In out context, we do not want the total variation to increase as we change scales from j to $j-1$. This can be formalized as

$$TV\left(s_{j-1}\right) \leq TV\left(s_j\right) \tag{4}$$

where s_j is the solution at the current time level, s_{j-1} is the solution at the next time level, and the total variation of the solution is given by

$$TV\left(s_j\right) = \sum_l \left|s_{j,l+1} - s_{j,l}\right| \tag{5}$$

where $s_{j,l+1}$ and $s_{j,l}$ are spatially consecutive values of the solution at time level j. By limiting the total variation of the solution to be less than or equal to the value in the initial data, spurious oscillations do not develop in the data.

We now want to impose the condition that the total variation of the data does not increase as we proceed from finer to coarser scales. Further, let us notice that we actually want to impose the TVD constraint on the subsampled data s_{j-1}. We will now describe a necessary and sufficient condition for this to hold for any values of s_j:

Theorem 1. *The following inequalities form a necessary and sufficient condition for a filter with coefficients a_k to have the TVD property (4) for any input s_j:*

$$\begin{aligned} \sum \left|a_{2k} + a_{2k+1}\right| &\leq 1 \\ \sum \left|a_{2k-1} + a_{2k}\right| &\leq 1 \end{aligned} \tag{6}$$

Proofs of Theorem 1 and Corollary 1 appear in the Appendix.

Corollary 1. *Suppose the filter coefficients a_k have the partition of unity property, i.e. $\sum_k a_k = 1$. Then a necessary and sufficient condition for the filter with coefficients a_k to have the TVD property (4) for any input s_j is that $a_k + a_{k+1} \geq 0$ for all k.*

The fifth and final condition for feature preservation we specify is that the general linear filter in equation (1) must be total variation diminishing after sampling:

$$a_k + a_{k+1} \geq 0 \quad \text{for all} \quad k \tag{$F5$}$$

4. Implementation of Biorthogonal Filter Bank

The general filter bank corresponding to a biorthogonal wavelet transform appears in Figure 3. The first half of the filter bank is called *analysis*, and the second is called *synthesis*. The filter h is the analysis low-pass filter, the filter g is the analysis high-pass filter, $\tilde{h}$ is the synthesis low-pass filter, and $\tilde{g}$ is the synthesis high-pass filter. Although the size of data in LP is just half of the size of data in the input we want it to be an accurate representation of the input data. Also, we want the *perfect reconstruction property:* the output exactly equals the input.

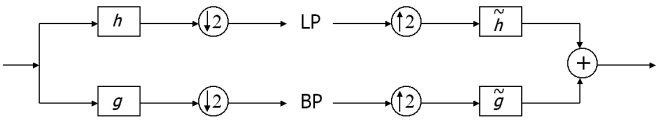


Fig. 3. Biorthogonal wavelet filter bank. LP stands for low pass and BP stands for band pass. The sign $\downarrow 2$ denotes downsampling and the sign $\uparrow 2$ denotes upsampling

A. The z-transform and Laurent Polynomials

Recall that a Laurent polynomial is a finite sum $L(z) = \sum_{j=n}^{m} a_j z^j$, where n, m are some (possibly negative) integers. The z-transform of a filter h with coefficients a_k is the Laurent polynomial $h(z) = \sum_k a_k z^k$. In other words, we think of the filter coefficients as coefficients of a Laurent polynomial. Note that we can think of the input sequence s as a Laurent polynomial as well. The main advantage of the z-transform is that the convolution of s and h corresponds to multiplication of Laurent polynomials. This makes it possible to have more algebraic structure in the convolution operation, as we will see below.

If we denote by h_e and h_o the even and odd parts of h regarded as Laurent polynomials we have:

$$h(z) = h_e(z^2) + z^{-1} h_o(z^2) \tag{7}$$

Then the Quadrature Mirror Filter (QMF) property implies (see [18]):

$$\begin{aligned} h_e(z) = \tilde{g}_o(z^{-1}), \quad h_o(z) = -\tilde{g}_e(z^{-1}), \\ g_e(z) = -\tilde{h}_o(z^{-1}), \quad g_o(z) = \tilde{h}_e(z^{-1}) \end{aligned} \tag{8}$$

$$g(z) = z^{-1}\tilde{h}(-z^{-1}), \quad h(z) = -z^{-1}\tilde{g}(-z^{-1}) \tag{9}$$

where $\tilde{h}$ is the synthesis low-pass filter, and $\tilde{g}$ is the synthesis high-pass filter.

Additionally, the perfect reconstruction property is equivalent to the following *complementarity condition*:

$$\det \begin{pmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{pmatrix} = 1, \tag{10}$$

which, in turn, is equivalent to (see [20]):

$$h_e(z) \text{ and } h_o(z) \text{ are relatively prime} \tag{$W1$}$$

i.e. that $h_e(z)$ and $h_o(z)$ have no common zeros.

A second condition related to the biorthogonal wavelet representation of our filter bank is that the *scaling function* (see [18]) of the wavelet transform is a finite energy function. This is guaranteed by a condition on the *restricted transfer operator* T. We define the restricted transfer operator T as follows: if the length of the convolution product of the sequence of coefficients a_k with itself is N, then T is the $(N-2)\times(N-2)$ matrix obtained from double shifts of this convolution product, times 2 (see [18]). Then the second condition derived from the biorthogonal wavelet representation is:

$$\begin{aligned} &T \text{ has one eigenvalue } \lambda = 1 \\ &\text{and all others have } |\lambda| < 1 \end{aligned} \tag{$W2$}$$

B. The Lifting Scheme

The lifting scheme is a method of factoring wavelet filters into basic building blocks called lifting steps, which also allows for spatial domain wavelet design. Daubechies and Sweldens ([20]) showed that any biorthogonal wavelet filter bank can be decomposed into lifting steps. In Figure 4 we

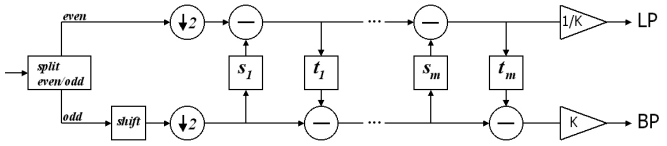


Fig. 4. Lifting steps decomposition: we think of the input and the filters as Laurent polynomials (see [20])

see the lifting steps decomposition of the analysis half of the biorthogonal filter bank in Figure 3: we first split the input sequence into even and odd entries, then we alternately filter each channel and use the output to modify the other channel, and in the end we multiply by scalars $1/K$, resp. K. In Figure 5 we see that the synthesis is exactly a *reversed* analysis. The implementation of the filter bank using lifting allows for an in-place computation of the wavelet transform and leads to an improvement in efficiency when compared to the standard implementation.

5. Axioms

Having defined feature preservation, we now enumerate a list axioms for the design of optimal feature-preserving (OFP) filters. According to the discussion in the previous two sections, we impose the following feature-preservation and

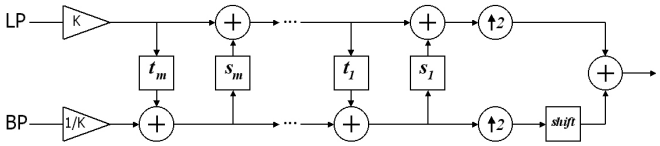


Fig. 5. Inverse lifting steps decomposition (see [20])

wavelet-representation requirements on the filter coefficients a_k:

$(F1)$ $a_k = a_{-k}$ for all k

$(F2)$ $\sum_k (-1)^k k^j a_k = 0,$ for $j = 0, 1, ..., p-1,$ and some $p \geq 1$

$(F3)$ $\sum_k a_k = 1$

$(F4)$ between all filters with the desired properties, the filter given by the coefficients a_k minimizes the L^2 distance to the $sinc$ filter

$(F5)$ $a_k + a_{k+1} \geq 0$ for all k

$(W1)$ if a_{-n} is the first nonzero coefficient, then the polynomials $a_{-n} + a_{-n+2}z + a_{-n+4}z^2 + ...$ and $a_{-n+1} + a_{-n+3}z + a_{-n+5}z^2 + ...$ are relatively prime.

$(W2)$ the restricted transfer operator T has one eigenvalue $\lambda = 1$, and all other eigenvalues have $|\lambda| < 1$

Axioms $(F1) - (F5)$ are related to the feature preservation properties of the OFP filters (see section 3 for details). It should be noted that the coefficients for the $sinc$ filter do not satisfy the TVD constraint $(F5)$. Thus, our OFP filter design strategy will seek a compromise between the ideal frequency behavior and the feature preservation properties of TVD filters. Finally, axioms $(W1)$ and $(W2)$ ensure that the proposed filter is the low pass filter of a biorthogonal wavelet transform which can be implemented as a series of lifting steps and has a finite energy scaling function (see Section 4 for details).

The following Theorem shows that the spatial domain axioms above are equivalent to a comprehensive list of feature-preserving, approximation, implementation, and optimality properties :

Theorem 2. *The requirements $(F1) - (W2)$ above are necessary and sufficient conditions for the following properties to hold:*

(a) *(Convergence of the cascade algorithm, see [18])*
The iteration $\phi^{(i+1)}(t) = \sum_k 2a_k\phi^{(i)}(2t-k)$, where $\phi^{(0)}$ is a box function, converges in L^2.

(b) *(Accuracy of approximation of order p)*
The error estimate for a function $f(t)$ of class C^p at scale $\Delta t = 2^{-j}$ is of the form $C(\Delta t)^p |f^{(p)}(t)|$.

(c) *(Total variation diminishes from fine to coarse scale)*

$TV(s_{j-1}) \leq TV(s_j)$.

(d) *(No moving features)*
There is zero phase shift from fine to coarse scale

(e) *(Lifting scheme implementation, see [20])*
There exists complementary high-pass filter, and the associated wavelet transform admits in-place implementation using the lifting scheme.

(f) *(Average grey level invariance)*
The average of the data is unchanged when passing from fine to coarse scale.

(g) *(Preservation of low frequencies)*
The moment of order 0 is 1, and the moment of order 1 is 0.

(h) *(Optimality)*
Between all filters with the desired properties, the filter given by the coefficients a_k minimizes the L^2 distance in the frequency domain to the ideal brick wall filter.

See the Appendix for a proof of Theorem 2.

6. Filter Design

We now use the framework developed in the previous Sections to design OFP filters and filter banks. For a symmetric filter with coefficients a_k the frequency response has the form:

$$G(\beta) = a_0 + 2\sum_{k\geq 1} a_k cos(k\beta) \qquad (11)$$

and all derivatives of G of odd order vanish at zero and at π. Therefore it is enough for $G(\beta)$ to have, for example, the second derivative at π equal to zero, and then its first, second and third derivatives at π are zero.

Remark. If we try to increase *both* the number of zeros at 0 *and* the number of zeros at π while imposing the TVD inequalities, we will notice that *there are no nontrivial solutions.* For example, if the second derivative of $G(\beta)$ has the property $G''(0) = G''(\pi) = 0$, then we get:

$$\begin{aligned} & a_1 + 4a_2 + 9a_3 + 16a_4 + 25a_5 + ... \\ = \; & a_1 - 4a_2 + 9a_3 - 16a_4 + 25a_5 - ... \\ = \; & 0. \end{aligned}$$

From this we conclude that:

$$a_1 + 9a_3 + 25a_5 + ... \; = \; 4a_2 + 16a_4 + 36a_6 + ... \; = \; 0,$$

and then

$$(a_1 + a_2) + 3(a_2 + a_3) + 6(a_3 + a_4) + 10(a_4 + a_5) + ... \; = \; 0$$

which together with the TVD inequalities implies that

$$a_1 + a_2 = a_2 + a_3 = a_3 + a_4 = a_4 + a_5 = ... \; = \; 0$$

and this together with the filter being FIR implies that $a_k = 0$ for all $k \geq 1$. The same holds if we replace the second derivative above with any other even order derivative of $G(\beta)$. Hence, we only impose conditions on the number of zeros at π.

A. OFP Low Pass Filter Design

Given two positive numbers N and p, we can find the filter coefficients of a symmetric TVD filter which is a partition of unity, has length N, has p zeros at π, and is closest in L^2 to the ideal low pass filter. We use the following two-step procedure:

Step 1. Make linear substitutions of a_i with some b_i such that the TVD condition on a_i translates to the condition that the new variables b_i are the coefficients of a point in the N-dimensional cube $[0,1]^N$.

Step 2. Use a quadratic optimization procedure to solve for the new variables b_i, under the p linear restrictions given by the existence of p zeros at π.

a_i	value
a_0	1/2
a_1	1/4
a_0	1/2
a_1	1/4
a_2	0
a_0	1/2
a_1	1/4
a_2	0
a_3	0
a_0	$(21\pi - 16)/36\pi$
a_1	$(33\pi + 16)/144\pi$
a_2	$(16 - 3\pi)/144\pi$
a_3	$-(16 - 3\pi)/144\pi$
a_4	$(16 - 3\pi)/144\pi$
a_0	$(375\pi - 224)/690\pi$
a_1	$(165\pi + 608)/1380\pi$
a_2	$(112 - 15\pi)/1380\pi$
a_3	$-(112 - 15\pi)/1380\pi$
a_4	$(112 - 15\pi)/1380\pi$
a_5	$(165\pi - 496)/1380\pi$

a_i	value
a_0	3/8
a_1	1/4
a_2	1/16
a_0	3/8
a_1	1/4
a_2	1/16
a_3	0
a_0	13/28
a_1	29/112
a_2	1/112
a_3	$-1/112$
a_4	1/112
a_0	$(-288 + 1075\pi)/2230\pi$
a_1	$(-192 + 9265\pi)/35680\pi$
a_2	$(5\pi + 36)/1115\pi$
a_3	$-(5\pi + 36)/1115\pi$
a_4	$(5\pi + 36)/1115\pi$
a_5	$(1344 - 185\pi)/35680\pi$

p	a_i	value
2	a_0	1/2
	a_1	1/4
4	a_0	3/8
	a_1	1/4
	a_2	1/16
6	a_0	5/16
	a_1	15/64
	a_2	3/32
	a_3	1/64
8	a_0	35/128
	a_1	7/32
	a_2	7/64
	a_3	1/32
	a_4	1/256
10	a_0	63/256
	a_1	105/512
	a_2	15/128
	a_3	45/1024
	a_4	5/512
	a_5	1/1024

Fig. 6. 1^{st} table: optimal feature preserving (OFP) filters with 2 zeros at π; longer filters are closer to the ideal low pass filter; 2^{nd} table: OFP filters with 4 zeros at π; again, longer filters are closer to the ideal low pass filter; 3^{rd} table: shortest OFP filters with p zeros at π (spline filters)

Examples. Here we explain how we obtained the tables in Figure 6. Let us examine the optimal feature preserving (OFP) filters with two zeros at π that, for a given length

N, are closest in L^2 to the ideal low pass filter. To compute them, we minimize the square of the norm of the difference between our filter and the ideal low pass filter, subject to the linear identities and inequalities given by $(F2)$, $(F3)$, and $(F5)$. Note that we supposed the filter to be symmetric from the very beginning, so $(F1)$ is also satisfied. Therefore we have to minimize a quadratic function of $a_0, ..., a_n$ subject to some linear inequalities and identities. We can replace the TVD inequalities with positivity requirements by using the following linear substitution:

$$\begin{aligned} a_n &= b_n \\ a_{n-1} &= b_{n-1} - b_n \\ a_{n-2} &= b_{n-2} - b_{n-1} + b_n \\ &\cdots \\ a_0 &= b_0 - b_1 + b_2 - ...(-1)^n b_n \end{aligned} \tag{12}$$

Then, imposing the TVD inequalities $(F5)$ on $a_0, ..., a_n$ is equivalent to imposing the positivity conditions $b_i \geq 0$ on $b_0, ..., b_n$. This allows us to reduce the problem to minimizing a quadratic function subject to some linear identities on the positive domain. Moreover, we can restrict the domain further to the cube $[0,1] \times [0,1] \times ... \times [0,1]$ when we notice that the TVD conditions imply that $b_k \leq 1$ for any k. This allows us to use a quadratic optimization procedure to compute the *exact* values of the solutions.

In Figure 6 we list the resulting a_k's just for $k \geq 0$, because for $k < 0$ they are determined by symmetry. In the first table in Figure 6 we impose accuracy of approximation of order at least $p = 2$. *Note that the spline filter* $(1/4, 1/2, 1/4)$ *is not optimal if we allow for a large enough number of filter taps.*

Let us look now at symmetric TVD filters with four zeros at π that are also closest to the ideal low pass filter. Again, we minimize the quadratic function given by the square of the norm of the difference between our filter and the ideal low pass filter, subject to the TVD inequalities, and to three linear identities, one identity given by $(F3)$ and, since $p = 4$, two identities given by $(F2)$. The result is the filters in the second table in Figure 6. *Again, note that the spline filter* $(1/16, 1/4, 3/8, 1/4, 1/16)$ *is not optimal if we allow for a large enough number of filter taps.*

We can also look at OFP filters which have the maximum number p of zeros at π. For any N we solve a system of linear inequalities given by the TVD conditions, and identities given by the equations $(F2)$ for the maximum p for which a solution still exists. We get the third table in Figure 6. *Note that these are exactly the spline filters.* On the other hand, they are the symmetric TVD filters that (for a given number of taps) have the smallest order error estimate for smooth data, while also preserving low frequencies, and annihilating high frequencies.

In conclusion, the *shortest filter* that satisfies all our axioms for a given p is a spline filter of length $p+1$. On the other hand, spline filters are *not optimal* if we allow a large enough number of filter taps.

B. High Pass Filter Design

We are now interested in finding short, symmetric, smooth high pass filters associated with the OFP low pass filters that we designed.

Recall that, since $g(z) = z^{-1}\tilde{h}(-z^{-1})$, the design of the analysis high pass filter g is equivalent to the design of the synthesis low-pass filter $\tilde{h}$. This simplifies the design, because symmetry and regularity conditions are easier to describe in terms of the synthesis low-pass filter $\tilde{h}$. Recall that the "Euclid algorithm" that generates a high-pass filter associated with a given low-pass filter in [20] does not give a unique solution. We would like to use the connection between g and $\tilde{h}$ to help us decide in the choice of g.

Suppose now that the synthesis low-pass filter $\tilde{h}$ has coefficients $..., a_{-1}, a_0, a_1, a_2, a_3, ...$, and is symmetric, i.e. $a_{-k} = a_k$ for all k. Then $\tilde{h}(z)$ is a symmetric Laurent polynomial of the form

$$\tilde{h}(z) = \ ... \ + a_2 z^{-2} + a_1 z^{-1} + a_0 + a_1 z + a_2 z^2 + \ ... \tag{13}$$

We have:

$$\begin{aligned} g_e(z) &= -\tilde{h}_o(z^{-1}) \\ &= \ ... \ - a_3 z^{-2} - a_1 z^{-1} - a_1 - a_3 z - \ ... \\ g_o(z) &= -\tilde{h}_e(z^{-1}) \\ &= \ ... \ - a_4 z^{-2} - a_2 z^{-1} - a_0 - a_2 z - a_4 z^2 - \ ... \end{aligned} \tag{14}$$

We are now going to describe an algorithm for finding coefficients a_k such that the filter g defined in terms of $\tilde{h}$ as in (14) is a high pass filter associated to a given low pass filter h.

The input of the algorithm consists of the coefficients of the filter h, and a positive integer l that is the desired number of zeros at π of the filter $\tilde{h}$. The s^{th} step in the algorithm looks for filter coefficients a_k such that $a_k = 0$ for all $k > s$, and such that all the linear conditions given by (10) and the linear constrains given by $\tilde{h}$'s zeros at π are satisfied. The algorithm terminates first time when such a solution is found. The algorithm *does* terminate since the number of degrees of freedom increases twice as fast as the number of constrains.

Examples. If for $h_1 = (1/4, 1/2, 1/4)$ we look for g such that $\tilde{h}$ has one zero at π, this algorithm finds the solution $a_0 = 3/2, a_1 = 1/2, a_2 = -1/4$, which gives

$$\tilde{h} = (-1/4, 1/2, 3/2, 1/2, -1/4).$$

Using the Quadrature Mirror Filter identities (QMF, see [18]), this implies that

$$g = (-1/4, -1/2, 3/2, -1/2, -1/4, 0, 0).$$

In the first table in Figure 7 we list the resulting a_k's for the filter h_1 and a few values of l (for $k < 0$ the coefficients are determined by symmetry).

One can of course apply the same method to any low pass filter h. For the filter

$$h_2 = (1/16, 1/4, 3/8, 1/4, 1/16)$$

we get the filters $\tilde{h}$ in the second table in Figure 7.

Similarly, for the filter

$$h_3 = (\frac{1}{112}, \frac{-1}{112}, \frac{1}{112}, \frac{29}{112}, \frac{13}{112}, \frac{29}{112}, \frac{1}{112}, \frac{-1}{112}, \frac{1}{112})$$

we get the filters $\tilde{h}$ in the third table in Figure 7.

a_i	value
a_0	3/2
a_1	1/2
a_2	-1/4
a_0	5/4
a_1	3/4
a_2	-1/4
a_3	-1/4
a_4	1/8
a_0	$15/14$
a_1	$13/14$
a_2	$-1/7$
a_3	$-9/14$
a_4	$5/28$
a_5	$2/7$
a_6	$-1/7$
a_0	$63/64$
a_1	$65/64$
a_2	$-1/16$
a_3	$-57/64$
a_4	$13/64$
a_5	$31/64$
a_6	$-3/16$
a_7	$-7/64$
a_8	$7/128$

a_i	value
a_0	5/2
a_1	5/16
a_2	-3/4
a_3	3/16
a_0	$28/15$
a_1	$63/80$
a_2	$-3/4$
a_3	$-5/24$
a_4	$19/60$
a_5	$-19/240$
a_0	$3/2$
a_1	$261/256$
a_2	$-37/64$
a_3	$-167/256$
a_4	$1/2$
a_5	$23/256$
a_6	$-11/64$
a_7	$11/256$

a_i	value
a_0	$39771/25480$
a_1	$27501/50960$
a_2	$-15/52$
a_3	$-243/5096$
a_4	$409/50960$
a_5	$409/50960$
a_0	$1476541/1136590$
a_1	$133247/174860$
a_2	$-32161/113659$
a_3	$-304387/1136590$
a_4	$63655/454636$
a_5	$3631/454636$
a_6	$-2501/568295$
a_7	$-2501/568295$

Fig. 7. The three tables above contain filters $\tilde{h}$ computed for the OFP filters $h_1 = (1/4, 2/4, 1/4)$, $h_2 = (1/16, 4/16, 6/16, 4/16, 1/16)$, $h_3 = (1/112, -1/112, 1/112, 29/112, 13/112, 29/112, 1/112, -1/112, 1/112)$, respectively; in each table the number of zeros at π increases with the number of filter taps

C. Lifting Decomposition

Once we decided on an associated analysis high pass filter g for a given OFP analysis low pass filter h, we can use the factoring algorithm from [20] to determine a lifting scheme decomposition corresponding to that particular choice of pair of filters (h, g). We will again have to make a choice between many possible lifting scheme decompositions. Our filters being symmetric, we decide on a symmetric factorization, i.e., one in which every quotient is a multiple of $z+1$ (see [20], Section 7.7).

Examples. We will now give the resulting lifting steps for some examples of filters (h, g). In the case that the OFP analysis low pass filter h is $(1/4, 1/2, 1/4)$, the lifting steps are given by filters s_1, the update filter t_1, and finally the scaling with factors K and 1/K. The filter t_1 that specifies the last lifting step depends on the number p of zeros at π that $\tilde{h}$ has. We denote it by s_i if $\tilde{h}$ has i zeros at π. The filter s_1 does not depend on $\tilde{h}$. We have:

$$\begin{aligned}
s_1 &= 1/2 + z/2 \\
t_1 &= -1/4 - 1/4z, \text{ if } p = 1 \\
t_1 &= 1/8z^2 - 3/8z - 3/8 + z/8, \text{ if } p = 2 \\
t_1 &= -1/7z^3 + 9/28z^2 - 13/28z - 13/28 + 9z/28 \\
&\quad -z^2/7, \text{ if } p = 3 \\
t_1 &= 7/128z^4 - 31/128z^3 + 57/128z^2 - 65/128z - \\
&\quad 65/128 + 57z/128 - 31z^2/128 + 7z^3/128, \text{ if } p = 4 \\
K &= 1/2 \quad \text{(scaling)}
\end{aligned}$$

For example, to implement the wavelet transform corresponding to $h = (1/4, 1/2, 1/4)$ and some $\tilde{h}$ that has 2 zeros at π we would go through the following lifting steps:

step 1: perform lifting using s_1
step 2: perform lifting using $t_1 = 1/8z^2 - 3/8z - 3/8 + z/8$
step 3: perform scaling using $K = 1/2$

In the case of the analysis low pass filter $h = (1/16, 1/4, 3/8, 1/4, 1/16)$ we find that the first lifting step will have to be t_1, and not s_1.We have:

$$\begin{aligned}
t_1 &= 1/4z + 1/4 \\
s_2 &= z + 1 \\
t_2 &= -3/16z - 3/16, \text{ if } p = 1 \\
t_2 &= 19z/240 - 4/15 - 4/15z + 19/240z^2, \text{ if } p = 2 \\
K &= 1/4 \quad \text{(scaling)}
\end{aligned}$$

In the case of the filter $h = (\frac{1}{112}, \frac{-1}{112}, \frac{1}{112}, \frac{29}{112}, \frac{13}{112}, \frac{29}{112}, \frac{1}{112}, \frac{-1}{112}, \frac{1}{112})$ we again find that the first lifting step will have to be t_1, and not s_1. We get:

$$\begin{aligned}
t_1 &= -1/z + 1 \\
s_2 &= -z/29 - 1/29 \\
t_2 &= 841/980z + 841/980 \\
s_3 &= 245z/377 + 245/377 \\
t_3 &= -5317/27440 - 5317/27440z, \text{ if } p = 1 \\
t_3 &= 19z/240 - 4/15 - 4/15z + 19/240z^2, \text{ if } p = 2 \\
K &= 13/28 \quad \text{(scaling)}
\end{aligned}$$

Let us describe a complete filter bank associated to the OFP filter $h = (\frac{1}{4}, \frac{1}{2}, \frac{1}{4})$. According to [20] there exist many complementary high pass filters associated to a given low pass filter. Let us look for a symmetric one. It is easy to check that the best we can do with a two-step lifting is

$$g = (-1/8, -1/4, 3/4, -1/4, -1/8, 0, 0).$$

It is not possible for g to be symmetric, but the one above has linear phase, and is the shortest complementary filter with these properties. The filters h and g form the analysis half of the filter bank. To get a biorthogonal wavelet transform the associated synthesis filters will have to be $\tilde{h} = (-1/8, 1/4, 3/8, 1/4, -1/8)$, and $\tilde{g} = (0, 0, -1/4, 1/2, -1/4)$. This is the dual of the biorthogonal Cohen-Daubechies-Feauveau (5,3) wavelet system.

7. Results

In this Section we will present some results obtained using the optimal feature preserving (OFP) filters we designed, and compare them with results obtained using filters that are not optimal and not feature preserving according to our design procedure, e.g. the linear and cubic lifting filters.

A. 1D Results

Figure 8 shows the same sequence of images shown in Figure 2, but using an OFP filter. Unlike Figure 2, the initially monotone data remains monotone. Although the dissipative nature of the filter is evident from some smearing (blurring) of the features, the dissipation of the filter is not too severe.

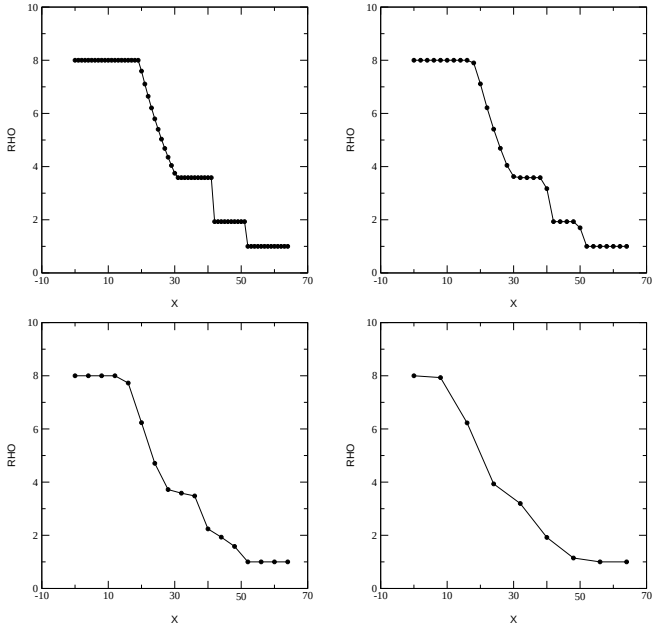


Fig. 8. Three Levels of OFP Lifting for Shock Tube Data

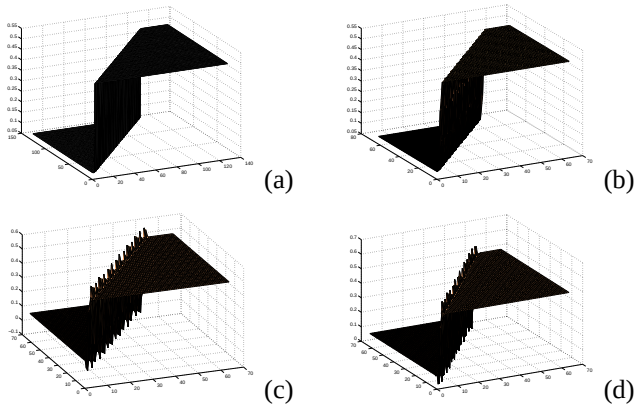


Fig. 9. OFP versus non-OFP lossy wavelet transforms for the pressure field: (a) original pressure values; (b) OFP transformation of pressure field; (c) linear lifting transformation of pressure field; (d) cubic lifting transformation of pressure field.

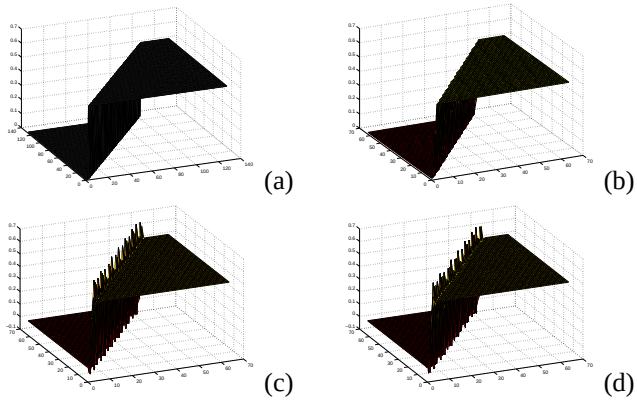


Fig. 10. OFP versus non-OFP wavelet transforms for the direction of the vector fields: (a) original vector directions; (b) vector directions after OFP transformation; (c) vector directions after transformation using the linear lifting scheme; (d) vector directions after transformation using the cubic lifting scheme.

B. 2D Results

Let us look at what happens when we use OFP and non-OFP wavelet transforms for lossy transformations of two dimensional data (where some percentage of the wavelet coefficients are discarded).

Let us first consider a fluid dynamics data set containing a stationary oblique shock. The oblique shock is a discontinuity across which the velocity magnitude and direction, pressure, and density change in a prescribed manner. The original scalar pressure field and velocity direction field are in Figure 9(a) and Figure 10(a), respectively. The streamlines of the velocity field are shown in Figure 11. In Figure 9(b-d) we see the results of OFP versus non-OFP wavelet transforms results for the pressure field. Note the significant Gibbs-like oscillations. Similarly, the direction of the vector field obtained using an OFP transformation has a smooth transition across the shock, as shown in Figure 10(a), while the direction of the vector fields obtained by using non-OFP filters varies wildly near the shock, as shown in Figures 10(b) and 10(c). In Figure 11 the streamlines of transformed vector fields obtained by using filters with the OFP properties are very similar to the streamlines of the initial vector field, while the streamlines of transformed vector fields obtained by using filters that do not have the OFP properties have some random artifacts. This can be explained by referring to Figure 10. The transformed vector fields obtained by using filters that do not have the OFP properties have directions that are not convex combinations of neighboring vectors and, therefore, create the "clustering" evident in the streamline plots.

Since our new filter banks were designed mainly for feature preservation, we do not expect them to be better than linear or cubic lifting wavelet transforms at lossy data transformation. Nevertheless, in general they provide error rates that are almost as good as for linear or cubic transformation, and, for some input datasets, they provide smaller error rates than linear and cubic wavelets. On the right side of Figures 12, 13, 14 the horizontal axis of the error rates graphs measures the percentage of wavelet coefficients which are discarded, and the vertical axis measures the total L^2 error.

For example, if the input is the periodic, high-oscillation texture dataset that we see on the left side in Figure 12, we get the error rates results on the right side of Figure 12. Note that the OFP error rates in Figure 12 are smaller than the linear and cubic transformation error rates in Figure 12. Also, note that the smallest error rates are obtained by using $\tilde{h}$ with two zeros at π, and not one zero at π as before.

If the input is the scalar density field on the left side of Figure 13 we get very similar error rate results for the OFP

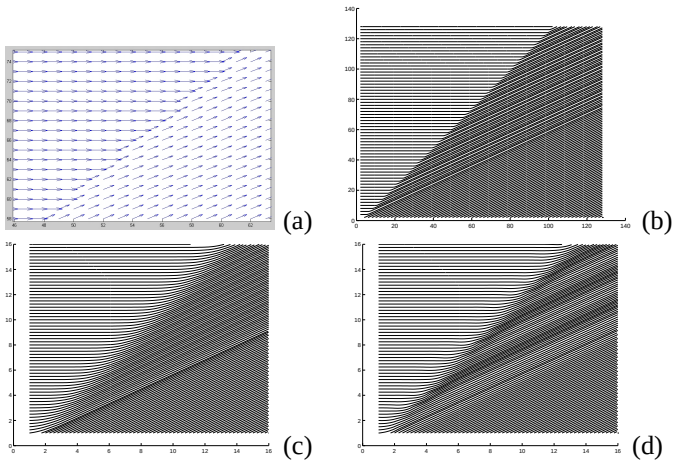


Fig. 11. OFP versus non-OFP wavelet transforms for vector fields: (a) Zoom in: the original vector field; (b) streamlines of the original vector field; (c) streamlines of vector field transformed using an OFP wavelet; (d) streamlines of vector field transformed using the linear wavelet

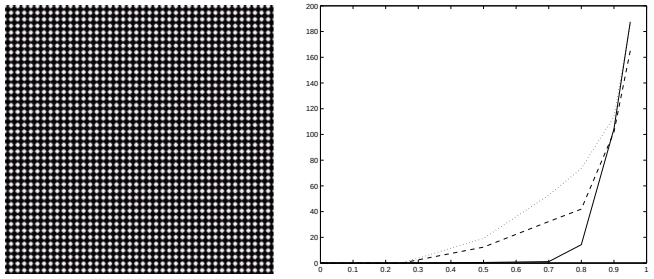


Fig. 12. Lossy OFP versus non-OFP wavelet transforms for a periodic, high-oscillation texture dataset: Left: texture dataset; Right: error levels using OFP (solid line), linear (dashed line) and cubic (dotted line) wavelet transforms

and non-OFP wavelet transforms, as we see on the right side of Figure 13.

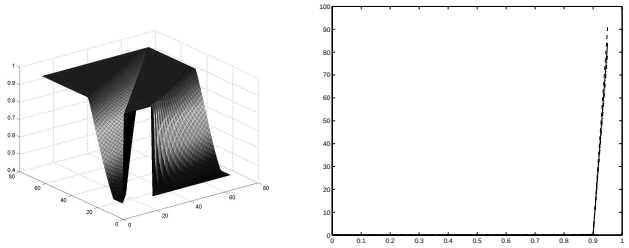


Fig. 13. Lossy OFP versus non-OFP wavelet transforms for a density dataset. Left: density dataset; Right: error levels using OFP (solid line), linear (dotted line) and cubic (dashed line) wavelet transforms

If the input is a scalar field of vector magnitudes that we see in Figure 14, which originated from a Pacific Ocean currents dataset, we get the error rates on the right in Figure 14.

In Figure 15 we see the result of applying OFP and non-OFP filters to a 2D shock image. To get the middle image we used a two level wavelet transform associated to the OFP filter (1/4,1/2,1/4). To get the right-side image we used a two level cubic lifting wavelet transform. We notice that the image transformed using the cubic wavelet has strong artifacts, due to the sharp variations near discontinuities.

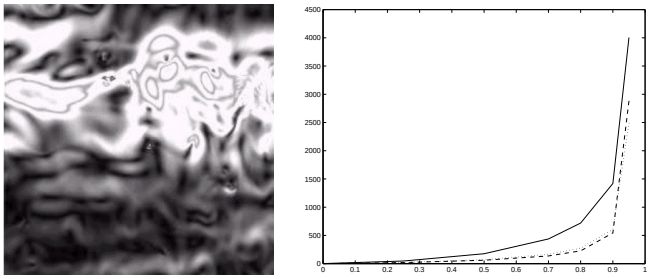


Fig. 14. OFP versus non-OFP wavelet transforms for a vector magnitude dataset. Left: vector magnitude dataset; Right: error levels using OFP (solid line), linear (dotted line) and cubic (dashed line) wavelet transforms

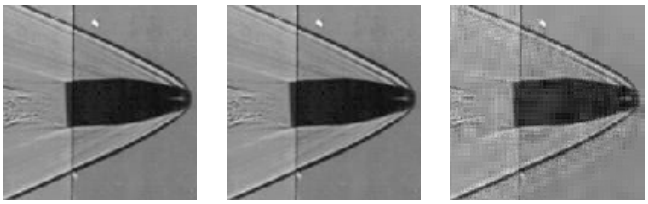


Fig. 15. Left: original 2d shock image; Middle: OFP lifting result, 80% wavelet coefficients discarded; Right: Cubic lifting, 80% wavelet coefficients discarded

C. 3D results

Let us also consider a three dimensional dataset. In Figure 16(a) we are looking at an explosion dataset, where an iso-surface was extracted and rendered. We apply OFP and cubic wavelet transforms to the original 3D dataset, and again extract and render an iso-surface from each transformed dataset. When we used the OFP wavelet, the octant corresponding to the smoothed version of the original data looks similar to the original iso-surface. When we used the non-OFP wavelet, the octant corresponding to the smoothed version of the original data does not look similar to the original iso-surface: brightness is significantly reduced due to Gibbs-like variation of the data in the transformed dataset, especially near the region where there is a sharp transition to very high temperature.

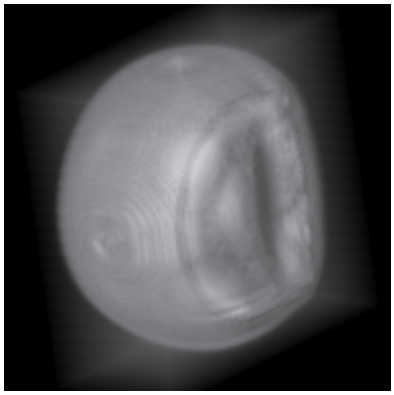
8. Conclusions

In this paper we defined a spatial domain framework for the analysis and design of multiscale filters. Included in this framework are a set of axioms that can be used to design filters that preserve certain characteristics of the data—namely the position, strength, and shape of features. The OFP filters and wavelet transforms designed using our axioms have consistently better feature preservation properties than usual filters (such as the linear and cubic lifting filters), without significant sacrifice of the approximation performance for lossy transformations.

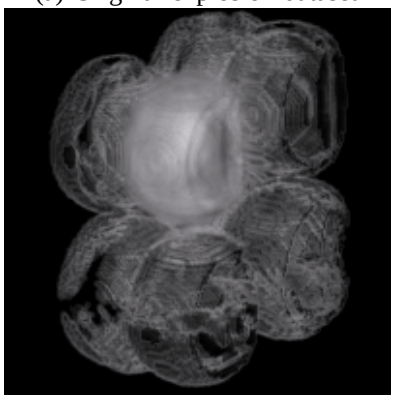
We suggest that the methods proposed here can also be used in conjunction with frequency domain methods to design multiscale linear wavelet filters. We plan to utilize these techniques to develop vector-valued wavelets with feature preserving qualities.

Appendix I

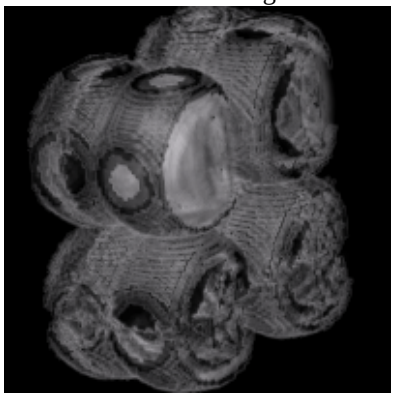
Proof of Theorem 1. Let us first look at the special



(a) Original explosion dataset



(b) 3D wavelet transform using an OFP wavelet



(c) 3D wavelet transform using cubic wavelet

Fig. 16. OFP versus non-OFP wavelet transforms in 3D case

case where the $2N$ values of s_j on level j are of the form $0, 0, ..., 0, 1, 1, ..., 1$. Without loss of generality we can suppose that $a_k = 0$ for all $k \leq 0$ and for all $k \geq n+1$. For large enough values of N the intermediate values $\bar{s}_j$ will have the form $0, 0, ...0, a_n, a_n + a_{n-1}, a_n + a_{n-1} + a_{n-2}, ..., a_n + ... + a_1, a_n + ... + a_1, ..., a_n + ... + a_1$. Here we suppose that we use an appropriate method of handling the boundary when computing the convolution product, by which the input signal s_j is padded at each boundary with values equal to the leftmost and the rightmost existing values at the boundary. For example, for our data, we pad the left boundary with zeros and the right boundary with ones. Therefore the subsampled data s_{j-1} will have either the form $0, 0, ..., 0, a_n, a_n + a_{n-1} + a_{n-2}, a_n + a_{n-1} + a_{n-2} + a_{n-3} + a_{n-4}, ..., a_n + ... + a_1, a_n + ... + a_1, ..., a_n + ... + a_1$, or the form $0, 0, ..., 0, a_n + a_{n-1}, a_n + a_{n-1} + a_{n-2} + a_{n-3}, ..., a_n + ... + a_1, a_n + ... + a_1, ..., a_n + ... + a_1$. From this it follows that either

$$TV\left(s_{j-1}\right) = \sum |a_{2k} + a_{2k+1}| \tag{15}$$

or

$$TV\left(s_{j-1}\right) = \sum |a_{2k-1} + a_{2k}| \tag{16}$$

according to the parity of the index l such that $s_{j,l} = 0$ and $s_{j,l+1} = 1$. Let us notice that $TV(s_j) = 1$. This implies that a necessary condition for $TV(s_{j-1}) \leq TV(s_j)$ is that the coefficients in (1) satisfy the two inequalities (6).

We will now prove that the two inequalities (6) are also a sufficient condition for $TV(s_{j-1}) \leq TV(s_j)$. Without loss of generality we can suppose $s_{j,k} = 0$ for all $k < 0$. For some fixed level j and values s_j let us define y by

$$y_k = s_{j,k+1} - s_{j,k} \tag{17}$$

Also, let us consider a matrix A with components $A_{m,i} = a_{i-2m-1} + a_{i-2m-2}$. Recall that $|A|_1 = \max_i \sum_m |A_{m,i}|$, where $|\ |_1$ is the usual L^1 operator norm. If (6) holds, then we have

$$\sum_m |A_{m,i}| \leq 1 \tag{18}$$

for any i, which implies that $|A|_1 \leq 1$. In particular, it follows that $|Ay|_1 \leq |y|_1$, so

$$\sum_m |\sum_i A_{m,i} y_i| \leq \sum_m |y_m| \tag{19}$$

We are now going to show that the desired inequality $TV(s_{j-1}) \leq TV(s_j)$ is implied by (19). We have

$$\begin{aligned} TV(s_{j-1}) &= \sum_m |\bar{s}_{j,2m+2} - \bar{s}_{j,2m}| \\ &= \sum_m \left| \sum_i a_i s_{j,2m+2+i} - \sum_i a_i s_{j,2m+i} \right| \\ &= \sum_m \left| \sum_i a_i (s_{j,2m+2+i} - s_{j,2m+i}) \right| \\ &= \sum_m \left| \sum_i a_i (y_{2m+i+2} + y_{2m+i+1}) \right| \end{aligned} \tag{20}$$

On the other hand according to (19) we have

$$\begin{aligned} TV(s_j) &= \sum_m |s_{j,m+1} - s_{j,m}| \\ &= \sum_m |y_m| \\ &\geq \sum_m \left| \sum_i A_{m,i} y_i \right| \\ &= \sum_m \left| \sum_i (a_{i-2m-1} + a_{i-2m-2}) y_i \right| \end{aligned} \tag{21}$$

Let us also notice that

$$\begin{aligned} & \sum_i a_i(y_{2m+i+2} + y_{2m+i+1}) \qquad (22) \\ = & \sum_i (a_{i-2m-1} + a_{i-2m-2})y_i \end{aligned}$$

since we can split the first sum into two sums and then regroup the terms with the same y_i factor to get the second sum.

Finally, by putting (20), (21) and (22) together, we get $TV(s_{j-1}) \leq TV(s_j)$. Therefore we have proved that (6) is a necessary and sufficient condition for the TVD property (4). Let us also notice that in the special case when $\sum a_i = 1$, i.e. the coefficients a_i are a partition of unity, we can use the inequalities

$$\begin{aligned} \sum |a_{2k} + a_{2k+1}| & \geq \sum a_k \qquad (23) \\ \sum |a_{2k-1} + a_{2k}| & \geq \sum a_k \end{aligned}$$

to conclude that the condition (6) is equivalent to having

$$a_k + a_{k+1} \geq 0 \qquad (24)$$

for all k. In other words, we obtain a proof of Corollary 1.

Proof of Theorem 2.

According to [18], (a) is equivalent to $(W2)$ and (b) is equivalent to $(F2)$ (see the Appendix more details). According to the Corollary in section 3.2, (c) is implied by $(F5)$ and $(F3)$. Also, according to our observations in section 3.1, (d) is implied by $(F1)$. According to [20] (e) is equivalent to $(W1)$. From their definitions, (f) is equivalent to $(F3)$ and (g) is implied by $(F1)$ and $(F3)$. Finally, (h) is equivalent to $(F4)$ since the Fourier transform preserves the energy from the spatial domain to the frequency domain.

For convenience, we describe some important results about wavelets and lifting from [18] and [20], used by us in the Theorem in Section 6 and in the design of the algorithm that generates an associated high pass filter of a given low pass filter.

Axiom $(W2)$ is also known as *Condition E*. We have (see [18]):

Theorem. Assume that the dilation equation $\phi(t) = \sum_k 2a_k\phi(2t-k)$ has a finite energy solution $\phi(t) \in L^2$. Then Condition E is a necessary and sufficient condition for the cascade sequence $\phi^{(i+1)}(t) = \sum_k 2a_k\phi^{(i)}(2t-k)$ to converge to $\phi(t)$. Moreover, if Condition E holds, then the cascade sequence converges to some $\phi(t) \in L^2$.

Another result from [18] is:

Theorem. If $\sum_k (-1)^k k^j a_k = 0$ for $j = 0, 1, ..., p-1$, then the error estimate for a function $f(t)$ of class C^p at scale $\Delta t = 2^{-j}$ is of the form $C(\Delta t)^p |f^{(p)}(t)|$.

The exact statement of the theorem about factoring wavelet transforms into lifting steps is (see [20]):

Theorem. Given a complementary filter pair (h, g), there always exist Laurent polynomials $s_i(z)$ and $t_i(z)$ for $1 \leq i \leq m$ and a nonzero constant K such that the analysis half of the filter bank can be decomposed as in Figure 4 and the synthesis half of the filter bank can be decomposed as in Figure 5.

ACKNOWLEDGMENTS

We thank Dr. James Fowler, Engineering Research Center, Mississippi State University, for pointing out the relationship between our filter banks and some Cohen-Daubechies-Feauveau filter banks. We also thank Ming Jiang and Yootai Kim, Department of Computer and Information Science, The Ohio State University, for help coding the wavelet transforms.

REFERENCES

[1] Special Issue on Partial Differential Equations and Geometry-Driven Diffusion. *IEEE Trans. Image Proc.*, 7, 1998.

[2] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel. Axioms and fundamental equations of image processing. *Arch. Rational Mech. Anal.*, 16(9):199–257, 1993.

[3] J. D. Anderson. *Modern Compressible Flow with Historical Perspective.* McGraw Hill, 1982.

[4] F. Arandiga, G. Chiavassa, R. Donat. Applications of Harten's framework for Multiresolution: From Conservation Laws to Image Compression. In *Multiscale and Multiresolution Methods*, Springer 2002.

[5] P. Claypoole, G. Davis, W. Sweldens, and R. Baraniuk. Nonlinear wavelet transforms for image coding. In *Asilomar Conference on Systems, Signals and Computers, (preprint)*, 1997.

[6] I. Daubechies. *Ten Lectures on Wavelets.* SIAM, 1992.

[7] D. Donoho. Denoising by soft thresholding. *IEEE Trans. Inf. Theory*, 41:613–627, 1995.

[8] D. Donoho, I. Daubechies, R. DeVore, and M. Vetterli. Adapting to unknown smoothness via wavelet shrinkage. *J. Amer. Stat. Assoc.*, 90:1200–1224, 1995.

[9] A. Harten. On a class of high resolution total-variation-stable finite-difference schemes. *SIAM J. Num. Anal.*, 21:1–23, 1984.

[10] A. Harten, E. Enquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory scheme, III. *J. Comp. Phys.*, 71:231–303, 1987.

[11] B. Kimia, A. Tannenbaum, and S. W. Zucker. Towards a computational theory of shape. *Lect. Notes Comp. Sci.*, 427, 1990.

[12] R. Machiraju, J. Fowler, D. Thompson, W. Schroeder, and B. Soni. EVITA - Efficient visualization and interrogation of tera-scale data. In *Data Mining for Scientific and Engineering Applications*, (Book Chapter) Kluwer, 2001, pp. 257-279.

[13] B. Nakshatrala, D. Thompson, and R. Machiraju. Ranked representation of vector fields. In *Proc. of Dagstuhl 2000 Seminar on Scientific Visualization (to appear)*.

[14] S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1998.

[15] T. Möller, R. Machiraju, K. Mueller, and R. Yagel. Evaluation and design of optimal filters using a Taylor series expansion. *IEEE Trans. Vis. Graphics*, 3:184–199, 1997.

[16] B. Nakshatrala. Feature-based embedded representation of vector fields. Master's thesis, Mississippi State University, December 1999.

[17] P. Perona and J. Malik. Scale-space diffusion and edge detection using anisotropic diffusion. *IEEE Trans. Pat. Anal. Int.*, 12:629–639, 1990.

[18] G. Strang and T. Nguyen. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 1996.

[19] W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29:511–526, 1997.

[20] W. Sweldens and I. Daubechies. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 29:511–546, 1997.

[21] J. Weickert, B. M. ter Haar Romeny, and M. A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Trans. Image Proc.*, 7(3):398–410, March 1998.

[22] H. C. Yee. A class of high-resolution explicit and implicit shock-capturing methods. NASA TM 101088, NASA Ames Research Center, February 1989.

[23] H.-M. Zhou. *Wavelet transforms and pde techniques in image compression.* PhD thesis, University of California Los Angeles, 2000.