NONLINEAR WAVELET TRANSFORMS FOR IMAGE CODING

ROGER CLAYPOOLE†, GEOFFREY DAVIS‡, WIM SWELDENS*, AND RICHARD BARANIUK†

November 1997

ABSTRACT. We examine the central issues of invertibility, stability, artifacts, and frequency-domain characteristics in the construction of non-linear analogs of the wavelet transform. The lifting framework for wavelet construction motivates our analysis and provides new insight into the problem. We describe a new type of non-linearity for use in constructing non-linear transforms: a set of linear predictors that are chosen adaptively using a non-linear selection function. We also describe how earlier families of non-linear filter banks can be extended through the use of prediction functions operating on a causal neighborhood. We present preliminary results for a synthetic test image.

1. INTRODUCTION

In his classic treatise on the workings of the human visual system, Marr focuses on the importance of the *representation* of information for various cognitive tasks [9]. The way in which information is represented brings out certain types of features while hiding others. Image compression applications also rely heavily on having an efficient representation of image data. Ideally we would like to approximate an image with a small number of parameters. The wavelet transform provides such an efficient representation. Most of the wavelet coefficients of a typical image are nearly zero, and the image thus is well-approximated with a small number of large wavelet coefficients.

The reason for the efficiency of the wavelet representation is that images often are well modeled as a set of locally smooth regions separated by edges. Within these smooth regions, fine-scale wavelet coefficients are small, and coefficients decay rapidly from coarse to fine scales. In the neighborhood of edges wavelet coefficients decay much more slowly, but because of the local support relatively few wavelet coefficients are affected by edges.

Still the large wavelet coefficients near edges are expensive to code. Much current research focuses on enabling wavelet coders to exploit the structure present in wavelet coefficients along edges. Successful coders perform some form of conditioning [11] or variance prediction [8].

Key words and phrases. image coding, nonlinear, wavelet, lifting.

†Electrical and Computer Engineering Department, Rice University, Houston, TX.

‡Mathematics Department, Dartmouth College, Dartmouth, NH.

*Lucent Technologies, Bell Laboratories, Murray Hill, NJ.

claypoorl@rice.edu, gdavis@cs.dartmouth.edu, wim@bell-labs.com, richb@rice.edu.

In this paper we focus on improving the properties of the wavelet transform rather than the encoder. More precisely, we try to build adaptive wavelet transforms that result in fewer large wavelet coefficients. Such non-linear wavelet transforms allow for more flexibility in image representations. Construction of non-linear filter banks is a straightforward process and has been discussed in [7] and [6]. Experiments with a nonlinear filter bank for image coding presented in [4] show results that are quite promising. The key open question in the use of these non-linear constructions is one of design: what is the most effective way to utilize the additional degrees of freedom obtained from relaxing the constraint of linearity?

We examine the central issues of invertibility, stability, artifacts, and frequency-domain characteristics (to the extent to which these are well-defined) in the construction of non-linear analogues of the wavelet transform. Our analysis is motivated by the new perspective provided by the lifting framework for the wavelet transform. We describe a useful new type of non-linearity for use in constructing our non-linear transforms: a set of linear predictors that are chosen adaptively using a non-linear selection function. We also describe how the family of non-linear filter banks of [7] and [6] can be extended through the use of prediction functions operating on a causal neighborhood.

2. The Lifting Scheme

Our non-linear transform is most easily described in terms of the lifting framework [13, 12]. The main feature of lifting is that it provides an entirely spatial interpretation of the wavelet transform, as opposed to the more traditional Fourier based constructions. This spatial interpretation will enable us to construct spatially varying and adaptive or non-linear wavelet transforms.

Let $X[n]$ be a signal. Our goal is to obtain a more efficient representation of $X[n]$, i.e., a representation in which most of the coefficients are near zero. We first partition $X[n]$ into its even and odd polyphase components $X_e[n]$ and $X_o[n]$, where $X_e[n] = X[2n]$ and $X_o[n] = X[2n+1]$.

If the $X[n]$ correspond to the samples of an underlying smooth function, then the even and odd polyphase components are highly correlated. This correlation structure is typically local and thus we should be able to accurately predict a coefficient from the odd polyphase component based on nearby coefficients from the even polyphase component.

Prediction: In the interpolating formulation of lifting, we first predict the odd polyphase coefficients $X_o[n]$ from the neighboring even coefficients $X_e[n]$. The predictor for each $X_o[n]$ is a linear combination of neighboring even coefficients:

$$P(X_e)[n] = \sum_l p_l X_e[n+l].$$

We obtain a new representation of the $X[n]$ by replacing $X_o[n]$ with the prediction residual. This leads to the first lifting step:

$$H[n] = X_o[n] - P(X_e)[n]. \tag{1}$$

If the underlying signal is locally smooth, the prediction residuals $H[n]$ will be small. Furthermore, the new representation contains the same information as the original signal $X[n]$: given the even polyphase $X_e[n]$ and the prediction residuals $H[n]$, we can recover the odd polyphase coefficients $X_o[n]$ by noting that

$$X_o[n] = H[n] + P(X_e)[n].$$

This prediction procedure is equivalent to applying a high-pass filter to $X[n]$ (hence the notation $H[n]$). The prediction filter is typically designed so that it is perfect for polynomials up to and including degree $N-1$. In wavelet terminology, the synthesis scaling function to which the prediction filter gives rise can reproduce polynomials of degree up to $N-1$ and the dual wavelet has N zero moments.

Update: The second lifting step transforms the even polyphase coefficients $X_e[n]$ into a low-passed and subsampled version of $X[n]$. We obtain this low-pass filtering version by *updating* $X_e[n]$ with a linear combination of the prediction residuals $H[n]$. We replace $X_e[n]$ with

$$L[n] = X_e[n] + U(H)[n], \tag{2}$$

where $U(H)$ is a linear combination of neighboring H values:

$$U(H)[n] = \sum_l u_l H[n+l].$$

Because each lifting step is always invertible, again no information is lost. Given $L[n]$ and $P[n]$, we have

$$X_e[n] = L[n] - U(H)[n]$$

and again

$$X_o[n] = H[n] + P(X_e)[n].$$

Note that H and L are at half rate, and thus this transform corresponds to a critically sampled perfect reconstruction filter bank. One can show that the update function determines the properties of the primal wavelet and dual scaling function. In particular, if the update filter is one half the adjoint of the predict filter then the primal wavelet has N zero moments as well [12].

A simple example of lifting is the construction of the Deslauriers-Dubuc family of wavelets from a single prediction step followed by a single update step. The following prediction and update steps are equivalent to performing a single stage of the (4,4) Deslauriers-Dubuc wavelet transform:

$$\begin{aligned} H[n] &= X_o[n] - (X_e[n-1] + 9X_e[n] + 9X_e[n+1] + X_e[n+2])/16 && (3) \\ L[n] &= X_e[n] + (H[n-2] + 9H[n-1] + 9H[n] + H[n+1])/32 && (4) \end{aligned}$$

The predict step cancels cubic polynomials and leaves the residual in the high pass signal $H[n]$. The update step results in a low-passed and subsampled version of X being placed in L. It should be emphasized that lifting is a general construction that is not limited to the

Deslauriers-Dubuc family. Using the Euclidean algorithm, one can decompose any FIR wavelet transform into a sequence of prediction and update steps [3].

3. Nonlinear Lifting

The analysis filters used for wavelet compression applications typically correspond to fourth order polynomial predictors like the one described above. Such predictions work well if the underlying signal is locally smooth. However, these predictions break down when the signal is not locally well-modeled as a low-order polynomial. In particular, the predictions work poorly near edges and other singularities.

Our goal is to switch between different predictors based on the local properties of the image. This makes the P operator data dependent and thus non-linear; lifting guarantees that the transform remains reversible. Where the image is locally smooth we use higher order predictors. Near edges we reduce the order and thus the length of the predictor as to avoid using coefficients across edges. Ideally we would like to use predictors that take into account the fact that discontinuities in images tend to occur along continuous curves. Such an adaptation would allow us to exploit the additional spatial structure that we know exists in edges.

3.1. **Problems.** Adapting the predictor makes the transform non-linear; thus the transform of the sum of two images is no longer the sum of each of the transforms. Consequently, it no longer makes sense to talk about the concept of basis functions, which relies fundamentally on linear superposition. We thus focus on the properties of the transform.

There are two problems with making the above predict-update lifting non-linear:

1. We need a coherent interpretation of the updated coefficients. After the first iteration of our transform, we are basing all of our predictions on updated coefficients. If we are to make effective predictions beyond the first iteration of the transform, we need some kind of structure in the update.

   Furthermore, the reason why wavelets provide efficient representations lies in their *space-frequency localization properties.* The spatial localization comes from the fact that the filter operations are local; this immediately carries over to the non-linear case. The frequency localization comes from the interpretation of the H as a band-pass-filtered and downsampled version of the signal and L as a low-pass-filtered and downsampled version of the signal. Maintaining this in the non-linear case is highly non-trivial.

   Consider again the example (4). While it is easy to see that the prediction filter P leads to a high pass filter, it is not immediately clear that the update U leads to a low pass filter. The reason is that the lifting structure mandates that the high pass coefficients H must be reused in the computation of L, and thus L depends both on P and on U. By carefully adjusting the update U to the prediction P, one can ensure that L is a low-pass-filtered and subsampled version of the original signal. In the example $U(H)$ had to be chosen as $(H[n-2] + 9H[n-1] + 9H[n] + H[n+1])/32$. While it is known how to adjust U for a spatially varying, but linear P [13], it is not immediately clear how to build a non-linear U adjusted to a non-linear P.

2. We need to ensure that the transform is stable. Lossy coding schemes introduce errors into the transform coefficients, so it is crucial that the non-linearities do not unduly amplify these errors. Our goal is to use a high order predictor in smooth regions and a lower order prediction near edges. In order to avoid sending side information on which predictor was chosen we need to base the choice only on the $X_e[n]$. However, in lossy compression the decoder only has quantized even coefficients $\widehat{X}_e[n]$ rather than the original coefficients $X_e[n]$. If we use locally adapted filters, quantization errors in coarse scales could cascade across scales and cause a series of incorrect filter choices leading to serious reconstruction errors. The problem cannot be solved by synchronization, i.e., having the encoder make its choice of predictor based on quantized even values. The reason is that the reconstructed values $\widehat{X}_e[n]$ are obtained from quantized low-pass values $\widehat{L}[n]$. The low pass signal $L[n]$ is a function of the prediction residual signal $H[n]$, which in turn depends on what filters are chosen for prediction. Hence the encoder cannot obtain the quantized values $\widehat{X}_e[n]$ until it selects a predictor, and it cannot select a predictor without obtaining $\widehat{X}_e[n]$. If we are to employ a non-linear lifting procedure for lossy coding, it is essential that we avoid this stability problem.

3.2. **Solution.** We propose a simple modification which solves both problems: switch the predict and update lifting steps in the wavelet transform. We thus first update the even samples based on the odd samples yielding the low pass coefficients. We then reuse these low pass coefficients to predict the odd samples, which gives the high pass coefficients. We use a linear update filter and let only the choice of predictor depend on the data. Because we update first and the transform is only iterated on the low pass, all low pass coefficients throughout the entire pyramid linearly depend on the data and are not affected by the non-linear predictor. Thus the prediction is only based on low pass coefficients which are computed as in the classical wavelet transform. Furthermore, if we perform the transform *backwards*, i.e., starting the prediction process at the lowest frequency (coarsest) subband and working from coarse to fine scales, we can keep the encoder and decoder perfectly synchronized.

The question remains on how to find the P and U filters even in the linear case. One idea is to simply use the same P and U filters from the Deslauriers-Dubuc family but in reversed order; this results in switching the analysis and synthesis functions. This, however, is problematic for coding applications, because the analysis wavelets in the Deslauriers-Dubuc family are much less smooth than the synthesis wavelets. Since reconstructed images are built up from synthesis wavelets, these non-smooth building blocks would lead to highly visible artifacts in the reconstructed image. Furthermore, trying to boost the smoothness of the new building blocks leads to needlessly long filters which cause ringing.

Instead we propose a solution based on Donoho's average-interpolation which fits into the update-predict form of lifting [5, 14]. It leads to the $(1, N)$ branch of the Cohen-Daubechies-Feauveau family which is biorthogonal to the box function [2]. Let us consider a simple example. The low pass coefficients are computed using a Haar filter:

$$L[n] = (x[2n] + x[2n+1])/2.$$

The high pass coefficients are the residuals of a prediction of the odd samples based on the $L[n]$. A first order Haar prediction is

$$H[n] = x[2n+1] - L[n],$$

while a third order predictor, i.e., one that is exact for quadratics, is given by

$$H[n] = x[2n+1] - (-L[n-1]/8 + L[n] + L[n+1]/8).$$

Higher order predictors can be build in a straightforward way. The smoothness of the resulting scaling functions increases with the order. A lower bound for the Hölder regularity $R(N)$ as a function of N is given by $R(3) = .678$, $R(5) = 1.272$, $R(7) = 1.826$, $R(9) = 2.354$, and asymptotically $R(k) \approx .2075N$ [5]. In numerical experiments the order $(1, 7)$ filter set has been found to yield performance approaching that of the 7-9 filter set that is more commonly used in coding applications.

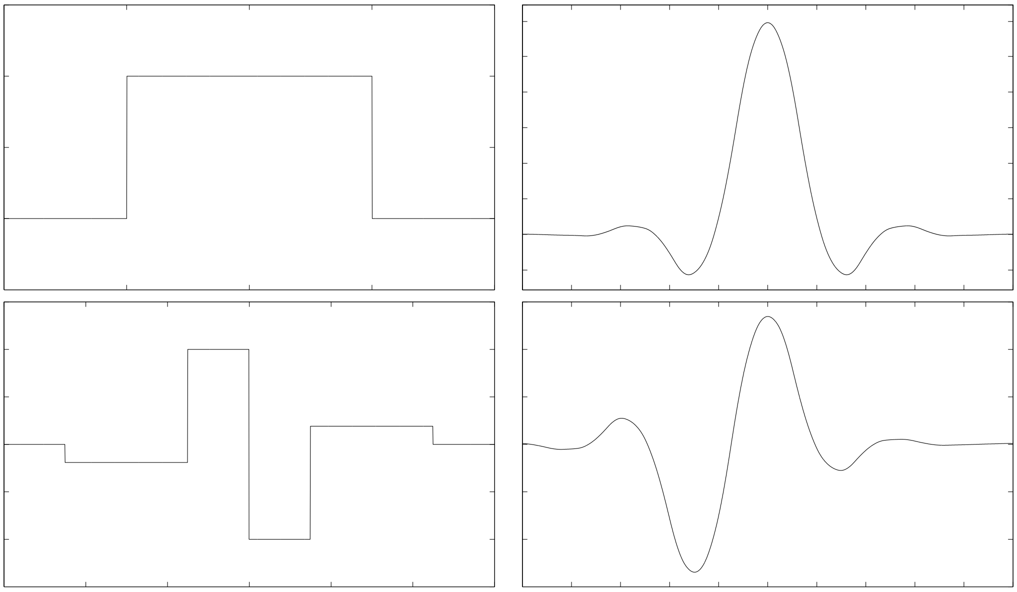

FIGURE 1. *Top row: analysis (left) and synthesis (right) scaling functions for the order (1,7) Cohen-Daubechies-Feauveau filter used in our experiments. Bottom row: analysis and synthesis wavelets. These basis functions correspond to the update-predict form of lifting.*

Our goal is to use non-linear predictor operators $P(L)$ so that

$$H[n] = x[2n+1] - P(L)[n],$$

obtained by switching between different linear predictors. We classify coefficients at a given scale into "edge" coefficients and "non-edge" coefficients based on the gradient at the next coarser scale. For a region containing non-edge coefficients, we use the 7th order predictor. Near edges we reduce the order of the predictor so that the neighborhood we use for prediction never overlaps the edge. This way we maintain high accuracy away from edges, and avoid large errors in the presence of edges. Figure 2 illustrates the process of selecting these predictors near an ideal step edge.

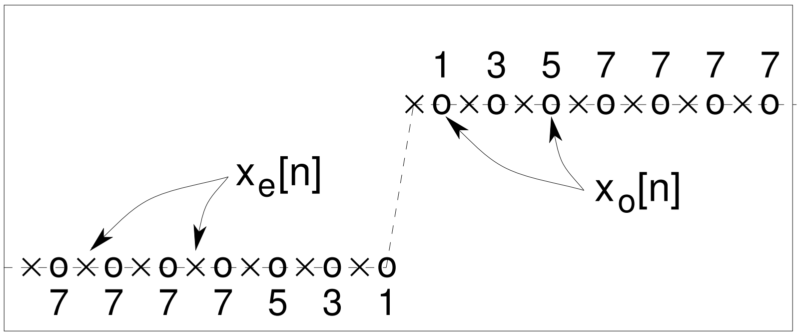

FIGURE 2. *Predictor selection at an ideal step edge. Numbers indicate the order of the predictors used. The closer to the edge, the lower order predictor is used.*

As we stressed in the previous section, maintaining synchronization between the adaptations of the encoder and the decoder is essential for a stable inversion. Encoding of a p-level transform proceeds as follows: we first compute the coarsest scale coefficients of the transform $L^p[n]$ by iterating the linear update procedure p times. We then quantize $L^p[n]$ to $\widehat{L}^p[n]$ and send them. Then we compute the high pass coefficients $H[n]$ as

$$H[n] = L^{p-1}[2n+1] - P(\widehat{L}^p)[n],$$

quantize them to $\widehat{H}[n]$ and send them. Both encoder and decoder now need the quantized values of the next finer scale $\widehat{L}^{p-1}$; the even and odd components are respectively computed by undoing the prediction and updating step, but now based on the *quantized* values $\widehat{L}^p[n]$:

$$\begin{aligned} \widehat{L}_o^{p-1}[n] &= \widehat{L}^{p-1}[2n+1] \;=\; \widehat{H}^p[n] + P(\widehat{L}^p)[n] \\ \widehat{L}_e^{p-1}[n] &= \widehat{L}^{p-1}[2n] \;=\; 2\widehat{L}^p[n] - \widehat{L}^{p-1}[2n+1] \end{aligned} \tag{5}$$

We now can compute the high pass coefficients on the next finer level. By basing our choice of predictor at each stage on the quantized values $\widehat{L}$, we maintain synchronization between encoder and decoder.

3.3. **Further Adaptivity.** Non-linear lifting also allows us to use not only the low-passed coefficients for prediction of $X_o[n]$, but also other odd coefficients in a causal neighborhood of $X_o[n]$. Suppose our signal $X[n]$ is a row in an image. We would predict $X_o[n]$ from low pass coefficients on its left and right. Suppose there is a vertical step edge near $X_o[n]$. The precise location of the edge cannot be determined from the low pass coefficients $X_o[n]$. However, if we know the value of the coefficients from the row immediately *above* $X_o[n]$, we can determine the orientation and strength of the edge, and we can use this information in the prediction of $X_o[n]$.

The use of coefficients from a causal neighborhood offers significant potential reductions in prediction errors. This increased flexibility comes at the price of decreased stability. Consider the example above in which we resolve difficulties in predicting the location of a vertical edge in a row of coefficients by using already inverted coefficients in the row above. Such a scheme permits a quantization error in one row to propagate along a vertical edge to all other rows. We can prevent such propagation by again using a DPCM-like strategy of using *quantized* data from the causal neighborhood for making predictions in the encoder as well as in the decoder.

3.4. **Related Work.** The update-then-predict lifting scheme we describe is related to the Laplacian pyramid of Burt and Adelson [1] in which images are represented as a series of prediction residuals, and the predictors are not constrained to being linear. The Laplacian pyramid has the disadvantage that it expands the number of coefficients in the image being transformed by a factor of 4/3. Lifting, on the other hand, guarantees a critically sampled decomposition.

Our non-linear lifting framework generalizes the ideas of de Quieroz *et al.* [4], and makes clear the relationship between the non-linear filter banks described by these authors and the wavelet transform. The filter bank described in [4] uses the update function $U \equiv 0$ and a non separable prediction function P that returns the median of its arguments. This filter bank performs particularly well for test images containing sharp edges, such as the cameraman image and images of text. Perceptually this non-linear transform has significant advantages near edges, and it minimizes problems with ringing around edges. However, the transform suffers from speckling artifacts due to aliasing of high frequency noise into the low pass subbands. The use of a better update function has the potential to eliminate this speckling while maintaining high quality reconstruction around edges.

4. Preliminary Results

Figure 3 shows the result of a simple feasibility test of an edge-avoiding adaptive transform. We demonstrate the ability of the adaptive transform to reduce the number of non-zero coefficients by computing the transform, thresholding the transform coefficients with a large threshold, and inverting the result. We use a black disk on a grey background as our test image, as this image contains edges in all orientations. The figure on the left is obtained by performing the wavelet transform using the (1,7) Cohen-Daubechies-Feauveau functions shown in Figure 1. The figure on the right was obtained using an adaptive transform.

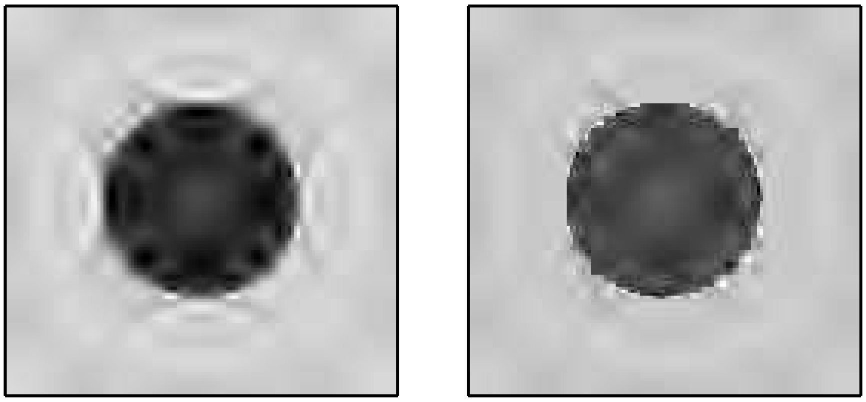

FIGURE 3. *Reconstructed unit disks with threshold $T = 1$. The left image was transformed with the $(1, 7)$ update-predict linear lifting. The right was transformed with an edge-avoiding non-linear lifting procedure. We assume that the forward and inverse transforms make the same edge decisions.*

For our adaptive transform we choose a filter from the $(1, N)$ branch of the CDF family, where $N \in \{1, 3, 5, 7\}$. The choice of predictor is based on an edge analysis of each 7-point prediction window. We locate edges in the data by finding local maxima in the gradient of the coefficients. We estimate the size of an edge by comparing an average of coefficients on either side of the edge, and we only adapt if the step size is large compared to the local variance on either side of the edge. The prediction filter is chosen as illustrated in Figure 2: we choose the width of our prediction window to be the largest window (up to width 7) not containing an edge.

We see that in the adaptive transform we have significantly reduced ringing around the edges of the disk and we have preserved edge sharpness. The reason is that edges in our new transform are represented in a more compact fashion, and as a result there is less degradation of the image when we zero out small, non-zero coefficients. Note that while ringing has been greatly reduced in the horizontal and vertical edges, there are still some ringing artifacts in the diagonal direction. The reason for these remaining artifacts is that we are using a separable transform in which we seek to avoid horizontal and vertical edges.

One future research avenue that will allow us to avoid privileged edge directions in the transform is to make use of a non-separable prediction function. We can further improve things by predicting values along edges rather than across edges by using data in a causal neighborhood.

Acknowledgement. This work was supported by an NSF Mathematical Sciences Postdoctoral Research Fellowship, NSF grant number MIP–9457438, and ONR grant noumber N00014–95–1–0849. G. Davis would like to acknowledge the hospitality of Rice University and Bell Laboratories where part of this work was done.

References

[1] Peter J. Burt and Edward H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, COM-31(4), April 1983.

[2] A. Cohen, I. Daubechies, and J. Feauveau. Bi-orthogonal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 45:485–560, 1992.

[3] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. Technical report, Bell Laboratories, Lucent Technologies, 1996.

[4] R. de Quieroz, D. A. F. Florêncio, and R. W. Schafer. Non-expansive pyramid for image coding using a non-linear filter bank. *IEEE Trans. Image Processing*, 1996. Preprint.

[5] D. L. Donoho. Smooth wavelet decompositions with blocky coefficient kernels. In [10], pages 259–308.

[6] Dinei A. F. Florêncio and Ronald W. Schafer. Perfect reconstructing nonlinear filter banks. In *Proc. ICASSP*, 1996.

[7] F. J. Hampson and J.-C. Pesquet. A nonlinear subband decomposition with perfect reconstruction. In *Proc. ICASSP*, 1996.

[8] Scott M. LoPresto, Kannan Ramchandran, and Michael T. Orchard. Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework. In James A. Storer and Martin Cohn, editors, *Proc. Data Compression Conference*, Snowbird, Utah, March 1997. IEEE Computer Society.

[9] David Marr. *Vision*. W. H. Freeman, New York, 1982.

[10] L. L. Schumaker and G. Webb, editors. *Recent Advances in Wavelet Analysis*. Academic Press, New York, 1993.

[11] Eero Simoncelli. Statistical models for images: compression, enhancement, and synthesis. In *Asilomar Conference on Signals, Systems, and Computers*. IEEE Signal Processing Society, November 1997.

[12] W. Sweldens. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Appl. Comput. Harmon. Anal.*, 3(2):186–200, 1996.

[13] W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, to appear, 1997.

[14] W. Sweldens and P. Schröder. Building your own wavelets at home. In *Wavelets in Computer Graphics*, pages 15–87. ACM SIGGRAPH Course notes, 1996.