Adaptive Spatial Prediction For Lifting Scheme in Image Compression

R.Satyabama[†] **and Dr.S.Annadurai**[††]

[†] Research Scholar, Govt. College of Technology, Coimbatore, INDIA
[††] Additional Director of Technical Education, Chennai, INDIA

Summary
Wavelet transform has demonstrated excellent compression performance with Natural images. However the performance suffers in the neighborhood of oriented edges. Under these conditions the Lifting –style implementation of wavelets are widely used in image coders. The framework of lifting offers the flexibility for developing adaptive wavelet transforms. In this paper adaptive prediction techniques based on the local properties of the image are proposed. This adaptive lifting transform appears promising for image compression.

Key words:
Adaptive lifting, Edge detection, Nonlinear wavelet transform ,Relative local variance , CDF wavelets.

1. Introduction

Due to its many advantages, such as multiresolution representation, good energy compaction, and decorrelation, the discrete wavelet transform (DWT) has become one of the most important techniques for image and video compression in the last decade and been adopted by JPEG2000 standard [1]. The wavelet-based JPEG2000 not only presents superior coding performance over the DCT-based JPEG [2], but also provides scalabilities in rate, quality and resolution.

Conventionally, 2-D DWT is carried out as a separable transform by cascading two 1-D transforms in the horizontal and vertical direction. The wavelet transform can be efficiently implemented by the lifting scheme [3], [4] where the FIR wavelet filter can be factored into several lifting stages. A lifting stage is comprised of the four steps: Split, Predict, Update and Normalize. The lifting scheme proposed by Sweldens in [3] and [4] is an efficient tool for constructing second generation wavelets and has advantages such as faster implementation, fully in-place calculation, perfect reconstruction with low memory and computational complexity [3].

It can also be considered as an alternate implementation of the first generation classical wavelet transform. In many applications it is desirable to have a filter bank that somehow determines how to shape itself according to the data that it analyzes. This can be achieved by allowing lifting scheme to adapt its update and prediction filters to the local properties of the signal. In this paper adaptivity is introduced by choosing the prediction operator based on the local properties of the image.

The paper is organized as follows. In Section 2 lifting scheme is briefly reviewed. In section 3 we discuss the issues of adaptivity and the selection of the prediction operator. Experimental results are reported in section 4 followed by conclusion.

2. Lifting Scheme

Each 1-D wavelet transform can be factored in to one or more lifting stages. A typical lifting stage is comprised of four steps: Split, Predict, Update and Normalize

Split: The signal x[n] is first split into even subset $\mathbf{x_e[n]}$ and the odd subset $\mathbf{x_0[n]}$,where $\mathbf{x_e[n]}$ = x[2n] and $\mathbf{x_0[n]}$ = x[2n+1].

Predict: Then the odd subset $\mathbf{x_0[n]}$ is predicted from the neighboring even subset $\mathbf{x_e[n]}$. The Predictor P(.) is a linear combination of the neighboring even subset

$$P(x_e)[n] = \sum_i p_i x_e[n+i]$$

where $\mathbf{p_i}$ is the prediction filter coefficient which is a highpass filter. This leads to the detail coefficient

$$\mathbf{d[n]} = \mathbf{x_0[n]} - P(x_e)[n].$$

If the signal is locally smooth, the prediction residual **d[n]** will be small. Given the even subset $\mathbf{x_e[n]}$ and the prediction residual **d[n]**, the odd subset $\mathbf{x_0[n]}$ can be recovered by noting that

$$\mathbf{x_0[n]} = \mathbf{d[n]} + P(x_e)[n].$$

Update: The Update step transforms the even subset $x_e[n]$ into a low-pass filtered version of x[n]. This coarse approximation is obtained by updating with a linear combination of the prediction residual **d[n]**.Then the approximation coefficients

$$\mathbf{c[n]} = x_e[n] + \mathbf{U(d)}[n]$$

where U(.) is a linear combination of neighboring d values given by

$$\mathbf{U(d)}[n] = \sum_j u_j\, d[n+j]$$

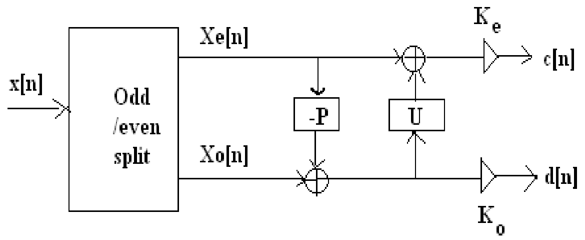
Manuscript received February 5, 2010
Manuscript revised February 20, 2010

where u_j is the low pass filter coefficient. The lifting construction guarantees perfect reconstruction for any Predict and Update filters.. Given **d[n]** and **c[n],** we have

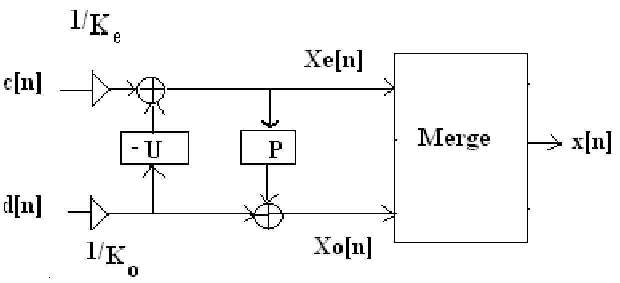
$$x_e[n] = \mathbf{c[n] - U(d)[n]}$$

Normalize: The outputs of the lifting are weighted by k_e and k_o. The values serve to normalize the energy of the underlying scaling and wavelet functions, respectively.
For 2D signals upon the completion of the 1-D lifting based horizontal transform, the 1-D lifting based vertical transform is performed in the same way.The forward and Inverse lifting is carried out as shown in Fig.1. The four steps in Inverse lifting are:
(i) Undo Normalize
(ii) Undo Update:
(iii) Undo Predict
(iv) Merge

The lifting framework allows us to incorporate non-linearities while retaining control over the properties of the wavelet transform. The non-linearity comes from adaptively choosing from a set of linear predictors. Such non-linear wavelet transforms provide added flexibility for image representations.



(a)



(b)

Fig.1 The Lifting Scheme. (a) Forward Transform

(b) Inverse Transform

3. Adaptivity in wavelet transforms

Wavelet bases typically employed for image compression utilize smooth scaling and wavelet functions. Such bases can be easily constructed with the predict-then-update form of lifting described above. Larger predictors that can exactly predict polynomials of higher degree correspond to smoother basis functions; these lifting predictors work well when the underlying signal is smooth. However, most of the images consist of regions of smoothness and texture separated by discontinuities (edges).These discontinuities cannot be well-represented by smooth basis functions. Since smooth basis functions correspond to lifting predictors with wide support, these predictors work poorly near edges, when the discontinuity is within the data we are using for the prediction.

We introduce a mechanism that allows us to choose the prediction operator based on the local properties of the image. This makes the P operator data-dependent and thus the transform is non-linear. However, lifting guarantees that the transform remains reversible. In regions where the image is locally smooth, we use higher order predictors. Near edges we reduce the order and thus the length of the predictor. Such an adaptation would allows us to exploit the spatial structure that exists in edges.

3.1 Adaptive prediction based on edge detection (Method1)

An edge detection algorithm analyzes the data in the 2-D prediction window to determine the location and the orientation of the edge. When an edge pixel is detected then we use a lower order predictor.

In this paper an edge detection algorithm using Sobel operator is considered [6]. The Sobel operator performs a 2-D spatial gradient measurement on an image. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. The Sobel edge detector uses a pair of 3x3 convolution masks, one estimating the gradient in the x-direction and the other estimating the gradient in the y-direction A convolution mask is usually much smaller than the actual image. As a result, the mask is slid over the image, manipulating a square of pixels at a time. The actual Sobel masks are shown below:

$$G_X = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_Y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

At each point in the image the resulting gradient approximations can be combined to give the gradient magnitude using G = $\sqrt{G_X^2 + G_Y^2}$ and using this information we can also calculate the gradient direction

θ = arc tan $\left(\frac{G_Y}{G_X}\right)$

Thus we define a point in an image as being edge point if its two dimensional first order derivative is greater than the specified threshold.

In lossy compression the decoder has only the quantized even coefficients rather than the original coefficients. If we use locally adapted filters, then quantization errors in coarse scales could cascade across scale and cause a series of incorrect filter choices leading to serious reconstruction errors.

The simple modification that solves this problem is to reverse the order of the predict and update lifting steps in the wavelet transform [5] as shown in Fig 2. We first update the even samples based on the odd samples yielding the low-pass coefficients c[n]. We then reuse these low-pass coefficients to predict the odd samples, which gives the high-pass coefficients d(n). When updating first the prediction operator is outside the loop. The coarse coefficients can be iterated to the lowest scale , quantized and reconstructed prior to the predictions. We use a linear update filter and let only the choice of predictor depend on the data.

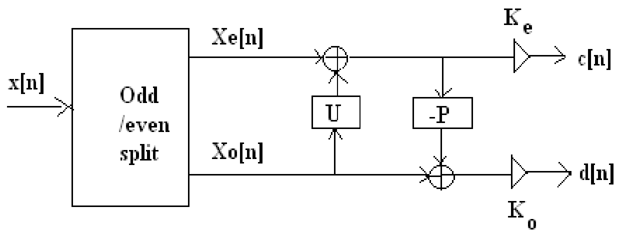


Fig. 2 Update First Lifting Scheme

When we do update first, the transform is only iterated on low pass coefficients and all c(n) depend on data and are not affected by nonlinear prediction. Here we considered CDF(1,N) wavelets (Cohen-Daubechies-Feauveau) for adaptive lifting[7]. The low pass coefficients are first computed using a Haar filter (one point update filter).We choose higher order predictors where the image is locally smooth, resulting in many negligible detail coefficients, and near edges, lower order predictors are activated, resulting in large lifting detail coefficients for better image representation. Thus based on the gradient the prediction filters are chosen.

we choose N={1,3,5} point prediction. The prediction filters are represented as

P(1) = [0, 0, 1, 0, 0] for order N = 1

P(2) = [0, -1, 8, 1, 0]/8 for order N = 3

P(3) = [-3, 22, 128, -22, 3]/128 for N= 5

and the update filter is

U =[1,1]/2

The low pass coefficients are first computed using a Haar filter (one point update filter), where

$$c(n) = \frac{x(n) + x(2n+1)}{2}$$

First order Haar prediction leading to(1,1) wavelet gives

$$d(n) = x(2n+1) - c(n)$$

The third order predictor leading to(1,3) wavelet gives

$$d(n) = x(2n+1) - \left(\frac{-c(n-1)}{8} + c(n) + \frac{c(n+1)}{8}\right)$$

The Fifth order predictor leading to(1,5) wavelet gives

$$d(n) = x(2n+1) - \left(\frac{-3c(n-2)}{128} + \frac{22c(n-1)}{128} + c(n) - \frac{22(n+1)}{128} + \frac{3c(n+2)}{128}\right)$$

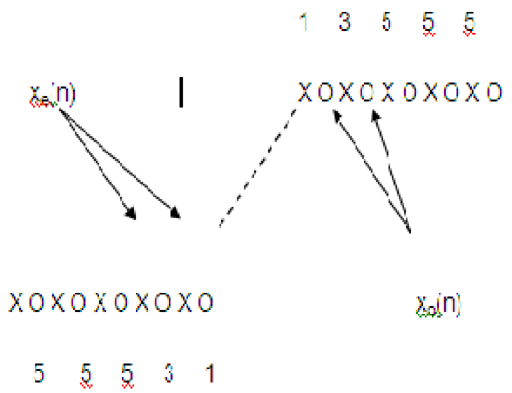


Fig. 3 Predictor Selection Near Edges. Number indicates the order of the predictor used

3.2 Adaptive prediction based on relative local variance (Method 2)

The smoothness of the image can determined by measuring the relative local variance (rlv).

$$rlv[I](i,j) = \frac{\sum_{K=i-T}^{i+T}\sum_{l=j-T}^{j+T}\left(I(K,l) - \overline{\mu_{i,j}}\right)^2}{var(I)}$$

With

$$\overline{\mu_{i,j}} = \sum_{K=i-T}^{i+T} \sum_{l=j-T}^{j+T} \frac{I(K,l)}{(2T+1)^2}$$

T is the size of sliding window, we take T =5. σ(I) is the standard variance of the image I. For all pixels (i, j) to be predicted, we first compute rlv[I] (i, j). Then quantizing the values of the rlv yields a decision map indicating which prediction filters should be used at which positions. The rlv for all subsequent pixels (i, j) to be predicted is computed, and suitable predictors are chosen. Two thresholds are chosen preliminarily according to practical situations. Quantization levels can be taken as multiples of the mean of the rlv. Test results have shown that [μ(rlv) 1.5μ(rlv) 2μ(rlv)] are the quantization levels that yield a good performance. The rlv value above the bigger threshold indicates that a lower order predictor namely P_1 should be selected. When rlv value is below the smaller threshold, a higher order predictor, namely P_3 should be activated. Otherwise P_2 is activated. Here also, we first update the even samples based on the odd samples. Then we reuse the low pass coefficients to predict odd samples which gives the high pass coefficients d(n). c(n) does not get affected and the choice of the predictor depends only on the data.

4. Experimental results

We have applied the adaptive prediction algorithms to an edge dominated circle image shown in Fig 4. The original image is transformed and it is compressed using SPHIT algorithm [8] and the performance is compared with the popular 9/7 transform. Table 1 gives the PSNR at different bitrates for the circle image. We observe that the image transformed with the adaptive lifting has sharp edges. The performance metric PSNR is computed as

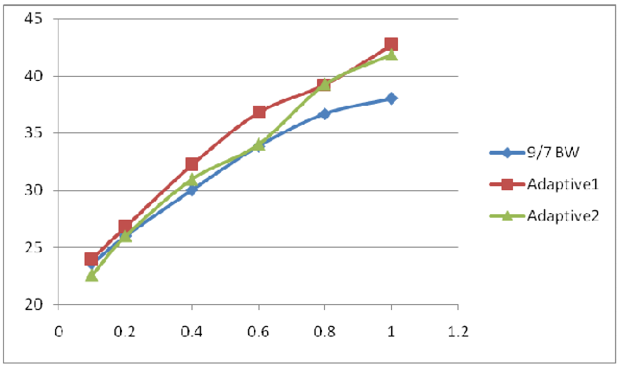
$$\text{PSNR} = 10\log_{10}\left[\frac{255^2}{e_{ms}^2}\right]\text{db}$$

$$e_{ms}^2 = 1/N^2 \quad \sum_i^N \sum_j^N (X(i,j) - Y(i,j)^2)$$

taking X as the original image and Y as the reconstructed image.

Table 1: PSNR for Circle Image

Bit rate	*9/7 wavelet*	*Adaptive prediction Method 1*	*Adaptive prediction Method 2*
0.1	23.59	24.01	22.56
0.2	26.01	26.8	26.02
0.4	30.01	32.21	30.96
0.6	33.83	36.79	34.02
0.8	36.69	39.23	39.26
1	38.02	42.69	41.86

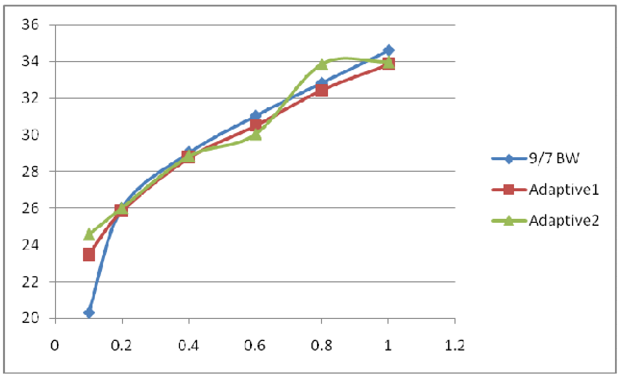


Graphical plot of PSNR vs Bitrate (Circle Image)

Table 2 gives the PSNR values of the proposed algorithms for the cameraman image. The visual quality of the proposed adaptive algorithm is comparable with 9/7 wavelet transform as shown in Fig.5.

Table 2: PSNR for Cameraman Image

Bitrate	*9/7 wavelet*	*Adaptive prediction Method 1*	*Adaptive prediction Method 2*
0.1	20.32	23.47	24.56
0.2	26.01	25.83	26.02
0.4	29.02	28.75	28.85
0.6	31.02	30.5	30.04
0.8	32.83	32.43	33.86
1	34.6	33.83	33.96

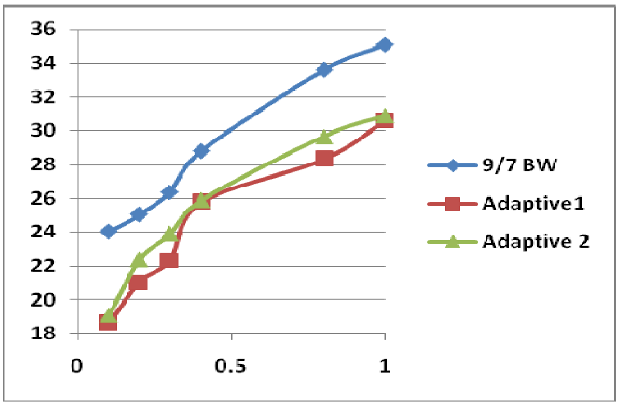


Graphical plot of PSNR vs Bitrate(Cameraman Image)

Table 3 gives the performance of the proposed algorithms for the Lena image. Since Lena image is a smooth image, 9/7 wavelet transform gives better PSNR and visual quality than the adaptive prediction methods as shown in Fig.6

Table 3: PSNR for Lena Image

Bitrate	*9/7 wavelet*	*Adaptive prediction Method 1*	*Adaptive prediction Method 2*
0.1	24.02	18.60	19.05
0.2	25.02	21.02	22.36
0.3	26.38	22.34	23.89
0.4	28.78	25.74	25.87
0.8	33.58	28.29	29.65
1	35.07	30.56	30.88



Graphical plot of PSNR vs Bitrate(Lena Image)

4.1 Conclusion

Lifting allows us to incorporate adaptivity and nonlinear operators into the transform. The proposed methods efficiently represent the edges. This adaptive lifting transform appears promising for image compression. It reduces edge artifacts and ringing and gives improved PSNR for edge dominated images. For cameraman image the performance is comparable with 9/7 wavelet transform. For smooth images like Lena 9/7 transform gives much better performance.

References

[1] "Information technology—JPEG 2000 image coding system: Core coding system," ISO/IEC 15444-1, 2004.

[2] "Information technology—Digital compression and coding of continuous-tone still images: Requirements and guidelines," ITU-T T.81,ISO/IEC IS 10918-1, 1993.

[3] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," Appl. Comput. Harmon. Anal., vol. 3, no. 2, pp. 186–200, 1996.

[4] I. Daubechies and W. Sweldens, "Factoring wavelet transform into lifting steps," J. Fourier Anal. Appl., vol. 4, no. 3, pp. 245–267, 1998.

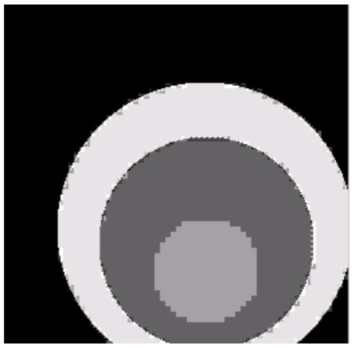
[5] R. L. Claypoole, G. M. Davis, W. Sweldens, and R. G. Baraniuk, "Nonlinear wavelet transforms for image coding via lifting," IEEE Trans.Image Process., vol. 12, no. 12, pp. 1449–1459, Dec. 2003.

[6] Rafael C. Gonzalez, Richard E.Woods " Digital Image Processing" Prentice Hall of India, NewDelhi, Second Edition.

[7] G.Uytterhoeven, D.Roose and A.Bultheel, "Wavelet transform using the lifting scheme", Report ITA-Wavelet-WP1.1, Department of Computer Science, Katholieke University Leuven, 1997.

[8] A. Said and William A. Pearlman.,1996, "A New, Fast, and Efficient Image Codec Based on Set PartitioninHierarchical Trees", IEEE Trans. On Circuitsand Systems for Video Technology, vol.6, No.3, pp. 243 -250.

ORIGINAL IMAGE



9/7 LIFTING

(1,N) ADAPTIVE LIFTING (METHOD1)

(1,N) ADAPTIVE LIFTING (METHOD2)

Fig.4 Circle image compressed at the rate of 0.5 bpp

ORIGINAL IMAGE

9/7 LIFTING

(1,N) ADAPTIVE LIFTING (METHOD1)

(1,N) ADAPTIVE LIFTING (METHOD2)

Fig.5 Cameraman image compressed at the rate of 0.5 bpp

ORIGINAL IMAGE

9/7 LIFTING

(1,N) ADAPTIVE LIFTING (METHOD1)

(1,N) ADAPTIVE LIFTING (METHOD2)

Fig.6 Lena image compressed at the rate of 0.5 bpp

R.Satyabama received B.E Degree in Electronics and communication Engineering from Thiyagaraja College of Engineering, Madurai, Tamilnadu, India and M.E. Degree in Applied Electronics from Govt. College of Technology, Coimbatore. Currently she is a research scholar in Anna University, Chennai. She is a life member of ISTE. Her areas of interest are signal processing, Image processing and communication.

Dr.S.Annadurai is currently the Additional Director of Technical Education, Chennai,.He did his B.E(ECE) and M.E (PS) at University of Madras, M.E (CSE) at Bharathiyar University and PhD at Anna University. He has contributed more than hundred research papers in different areas of Computer science and Engineering such as Automatic Speech Recognition, Neural and Fuzzy Neural Networks, Image Processing and Computer Networks. He is a life member of Professional bodies such as CSI and ISTE.