WAVELET VIDEO CODING VIA A SPATIALLY ADAPTIVE LIFTING STRUCTURE

Zhen Li[*1], *Feng Wu*[2], *Shipeng Li*[2], *and Edward Delp*[1]

[1]Video and Image Processing Laboratory (VIPER)
School of Electrical and Computer Engineering, Purdue Univ., West Lafayette, Indiana, USA
[2]Microsoft Research Asia, Beijing, China

ABSTRACT

In this paper we present a spatially adaptive wavelet video coding technique with an update-first lifting structure. A common problem in many adaptive-transform frameworks is the introduction of large overhead to address side information. In this paper we demonstrate that our structure does not need to transmit any side information to synchronize the encoder and decoder. We incorporate this technique in a motion compensated wavelet video codec. The experimental results confirmed the performance improvement.

1. INTRODUCTION

A typical hybrid wavelet video encoder generally consists of the following three components as shown in Figure 1. First the video sequences are sent to motion prediction to de-correlate the temporal dependence, the residue frames are generated here. Afterwards, a wavelet transform is used to de-correlate the spatial dependence inside a residue frame and obtain transform coefficients. Finally these coefficients are quantized and sent to an entropy coder to form the compressed stream.

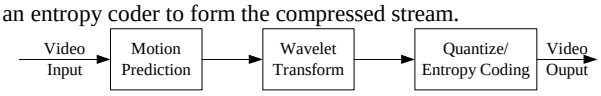

Figure 1. Hybrid wavelet video encoder

However, despite a great deal of effort in designing motion prediction, quantizers and entropy coders to adapt to various characteristics of video sequences, there is relatively little work reported on adaptive transform for video coding in the literature. One reason is that it is both conceptually and computationally difficult to design efficient transforms with respect to the spatial context. Moreover, generally a large overhead is inevitable needed to address the context information in such a design and can easily overwhelm the performance gain of adaptive transforms.

Fortunately, the first challenge is alleviated with the introduction of the lifting structure [1]. The lifting structure provides a spatial domain interpretation of wavelet transforms. In this paper we base our work on an update-first lifting structure proposed by Claypool et al [2]. This structure introduces adaptivity in the predict lifting step and has been proved useful for edge-dominated still images. We incorporate this technique into our hybrid video codec. Our experiments confirmed the performance improvement.

The rest of this paper is organized as follows. Section 2 describes the problem formulation of adaptive transforms for video coding. We give a brief introduction of the lifting structure in Section 3 and further explore in details the update-first lifting structure. We present our video coding with adaptive lifting in Section 4 and evaluate the experimental results in Section 5. Section 6 concludes the paper with remarks on future work.

2. PROBLEM FORMULATION

It is widely recognized that the Daubechies (9,7) wavelet transform achieves the best compression performance for still natural images compared to other wavelet transforms due to its ability to efficiently approximate smooth signals. However, this may not be true for the residual frames generated by motion compensation in a video codec. Residual frames may have a lot of edges and discontinuities and cannot be efficiently represented by long tapped filter banks such as the (9,7) transform.

To make this clearer, we present an example. Figure 2 is the Y component of the 1st frame in the Foreman QCIF sequence which is coded as an I frame. Figure 3 is the Y component of the 210th residual frame generated after the motion prediction in a hybrid wavelet transform encoder. Note that since the original Y in a residue frame where the pixels range from –255 to 255, we clip it to (0, 255) to ensure proper display. We then compressed each of these two frames with the (9,7) transform and the (1,7) transform under various data rates. The PSNR is evaluated by taking these two images as original images and comparing them with the reconstructed images respectively. Figure 4 and Figure 5 show that the (9,7) transform clearly outperforms the (1,7) transform for the I frame, while in the 210th P frame the two transforms outperform each other alternatively. Intuitively, if we can find a more suitable transform for each spatial area, the overall coding efficiency could to be increased.

Generally it is difficult to implement such adaptive transforms since the encoder needs to transmit side information so that the decoder is able to employ the exactly same transform in each spatial area as the encoder. And as traditional transforms are constructed based on the frequency domain analysis, it is also hard to construct transforms according to the spatial context. In Section 3 we will see the adaptive lifting structure can address

[*] This work was performed while the author was with Microsoft Research Asia

these two problems and provide a solution for the adaptive transform in video coding.

Figure 2. Y component of the 1st frame of Foreman QCIF

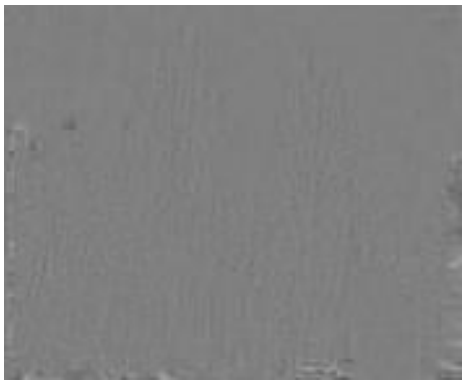

Figure 3. Y component of the 210th residue frame of Foreman QCIF

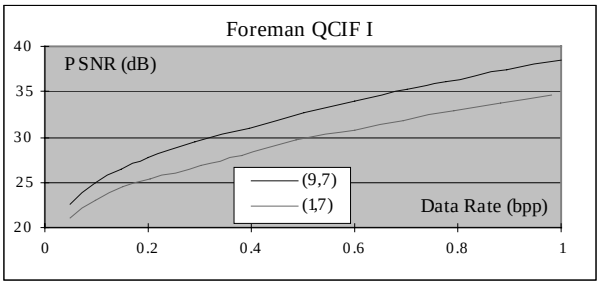

Figure 4. Coding efficiency for 1st frame

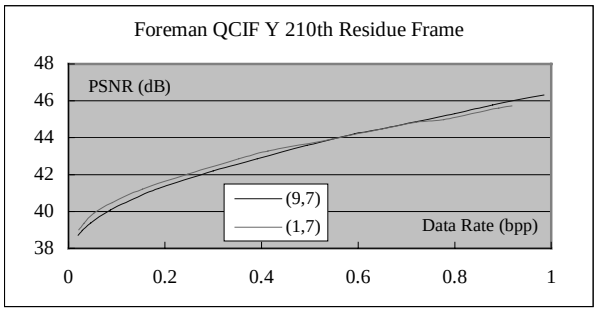

Figure 5. Coding efficiency for 210th frame

3. ADAPTIVE LIFTING SCHEME

In this section we first present a short introduction to the lifting scheme. Readers who are interested in more detail are referred to [1][2]. We then discuss the adaptive lifting structure in detail.

The lifting structure emerged as a new approach to construct wavelets. It was originally developed to adjust wavelet transforms to complex geometries and irregular characteristics of sampling data. It provides an entirely spatial domain interpretation of wavelet transforms. It was shown in [3] that all 1-D FIR filter banks can be implemented by the lifting structure. Due to its advantage of custom design, in-place computation, integer-to-integer transforms, and speed, the lifting scheme has been widely investigated recently.

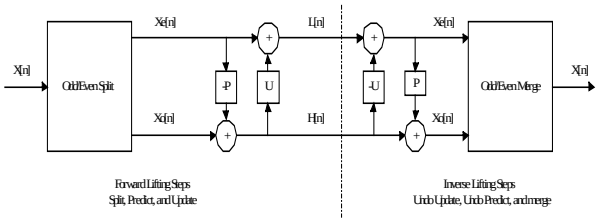

Figure 6. Standard predict-first lifting structure

A typical lifting scheme, shown in Figure 6, comprises three steps in the analysis part: split, predict and update. On the synthesis part, correspondingly, there are also three steps, i.e., undo update, undo predict, and merge. We describe the three analysis steps in the following.

Split: This step takes in the signal $x[n]$ and splits into even and odd components, $x_o[n]$ and $x_e[n]$, respectively, where

$x_e[n] = x[2n]$ (1)

$x_o[n] = x[2n + 1]$ (2)

Predict: In this step $x_o[n]$ is predicted from its neighboring even coefficients $x_e[n]$'s with the predictor

$$P(x_e)[n] = \sum_l p_l x_e[n+l] \quad (3)$$

Here p_l is the prediction coefficients and $x_o[n]$ is replaced by the prediction residual

$d[n] = x_o[n] - P(x_e)[n]$ (4)

At the decoder given the even components $x_e[n]$'s and the prediction residual $d[n]$, we can recover the odd components by

$x_o[n] = d[n] + P(x_e)[n]$ (5)

which ensures the perfect reconstruction (PR) property in this lifting step. It is noted that from the point of view of signal processing, the predict step is actually a high pass filter, which extracts the high frequency component of the original signal.

Update: In this step the even coefficient $x_e[n]$ is updated with

$c[n] = x_e[n] + U(d)[n]$ (6)

where $U(d)$ is a linear combination of prediction residuals

$$U(d)[n] = \sum_l u_l d[n+l] \quad (7)$$

where u_l is the weighting factor.

Here the update step is a low pass filter. This step also reserves the PR property since given $c[n]$ and $d[n]$'s, the $x_e[n]$ can be recovered by

$x_e[n] = c[n] - U(d)[n]$ (8)

The inverse lifting steps basically reverse the three steps mentioned above.

In light of the spatial domain interpretation of wavelet transforms by the lifting structure, there has been work trying to introduce adaptivity into the spatially domain. In [4], an adaptive update approach is presented based on maximum likelihood decoding, where no bookkeeping is required. To our understanding, it demonstrates entropy reduction in synthesis signals and images, whereas no application with respect to residue images has been found in the literature.

Our work is based on [3], where a simple structure is used to introduce adaptivity in the predict step. The basic idea is to reverse the order of predict and update step as shown in Figure 7. The even coefficients are first updated based on the odd samples and yield low pass approximation coefficients $c[n]$, i.e.,

$$c[n] = x_e[n] + U(x_o)[n] \quad (9)$$

Here $U(x_o)[n]$ is a linear combination of $x_o[n]$'s with

$$U(x_o)[n] = \sum_l u_l x_o[n+l] \quad (10)$$

where u_l is the weighting factor.

then these low pass coefficients are used to predict the odd samples and gives the high pass coefficients $d[n]$ by

$$d[n] = x_o[n] - P(d)[n]. \quad (11)$$

Here $P(d)$ is the predictor with

$$P(d)[n] = \sum_l p_l d[n+l] \quad (12)$$

where p_l is the prediction coefficient.

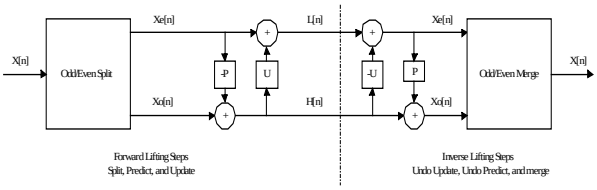

Figure 7. Update-first lifting scheme

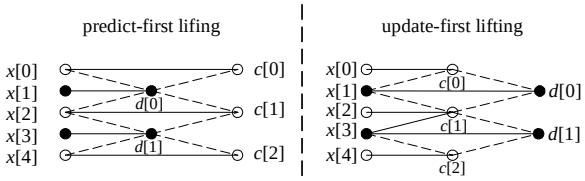

Figure 8. An example of the lifting structure

A simple example is presented. The left part of Figure 8 is a standard predict-first lifting structure and the right part is a corresponding update-first lifting structure. We note that if in the predict-first scheme we also want to introduce spatial adaptivity based on the local spatial property in the predict lifting step, i.e., the prediction coefficients p_l's are not fixed but dependent on the sampling data. A straightforward approach is to minimize the difference between the predicted sample $x_o[n]$ and the predictor. Then at the decoder, since the predict function is also dependent on $x_o[n]$'s, which are not available when we perform undo-predict step, we cannot reconstruct the same transform and drift error is introduced due to the mismatch.

On the other hand, with the update-first lifting scheme, the predict function is based on the updated coefficients $c[n]$'s, all of which are available at the decoder before the undo-predict step, hence we are able to perfectly reconstruct the predict step.

Two comments on the update-first adaptive lifting structure are in order here. First is that in the original lifting scheme we can implement the adaptive update but the fixed predict with the same mechanism here. However, it turns out that the prediction residuals $d[n]$'s, which are the high pass residuals, are not accurate enough to reflect the local spatial property. Furthermore, the introduction of the adaptivity in the predict step, or high pass step, is more critical than that in the update step, or low pass step,. A more accurate prediction can directly result in smaller coefficients, while a better update only results in different low pass residues, which are generally still large.

Second, another approach to implement adaptive predict in the predict-first structure is to have the predict function dependent only on the even samples $x_e[n]$'s, which are available prior to the undo-predict step. However, $x_e[n]$'s, unlike $c[n]$'s, contain no information about the predicted samples $x_o[n]$. Hence the spatial property $x_e[n]$'s expressed can be totally different from that of $x_o[n]$'s, esp. in those areas where a lot of edges and discontinuities exist.

4. SPATIALLY ADAPTIVE WAVELET VIDEO CODING

In this section we incorporate the idea of spatially adaptive lifting structure to the hybrid video coding.

The video coding framework is basically a typical motion compensated (MC) 2D wavelet structure, as shown in Figure 9. The residue frames obtained after the motion prediction are sent to the wavelet codec where the adaptive lifting structure is used.

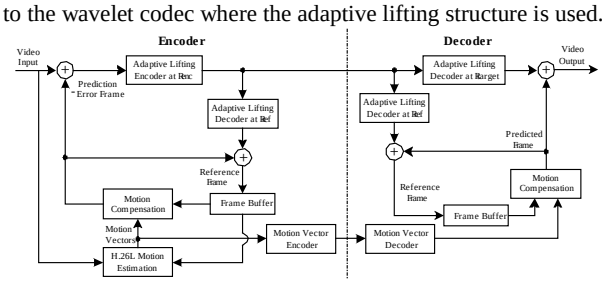

Figure 9. Video coding structure

The lifting structures are chosen from a fixed set of wavelet transforms, which have different tap length, to adapt to various spatial properties. For each data we first examine its spatial smoothness characteristics. The smoothness is determined by fitting the predicted sample and its nearby data samples with order-n polynomials, where higher order fittings indicate smoother areas. The adaptive lifting is then used. The basic idea is to use higher order transforms to smooth areas while using lower ones for edge areas and discontinuities. We are also considering generating the lifting structures online according to the local image property rather than making a choice over a fixed set. This approach may problems due to complexity issues and is note reported here.

For the sake of simplicity, we employ only the first four wavelet transforms in the (1,N) branch of CDF family [5] here, as also used in [2]. We give the coefficients for reference. The low-pass update coefficients are obtained using a Haar filter

$c[n] = (x[2n] + x[2n+1])/2$ (13)

The high-pass predict coefficients are obtained as the residues of a prediction of the odd samples, where

For the (1,1) transform

$d[n] = x[2n+1] - c[n]$ (14)

For the (1,3) transform

$d[n] = x[2n+1] - (-c[n-1]/8 + c[n] + c[n+1]/8)$ (15)

For the (1,5) transform

$d[n] = x[2n+1] - (3*c[n-2]/128 - 11*c[n-1]/64 + c[n] + 11*c[n-1]/64 - 3*c[n-2]/128)$ (16)

For the (1,7) transform

$d[n] = x[2n+1] - (-5*c[n-3]/1024 + 44*c[n-2]/1024 - 201*c[n-1]/1024 + c[n] + 201*c[n-1]/1024 - 44*c[n-2]/1024 + 5*c[n-3]/1024)$ (17)

5. EXPERIMENTAL RESULTS

This section verifies the performance of the adaptive lifting structure. We first use the adaptive lifting to the 210th residue frame as mentioned in Section 2. Figure 10 shows that the adaptive structure achieves approximately a 0.3 - 0.5dB gain.

The performance comparisons of the original video codec and the codec with the adaptive lifting structure are shown in Figure 11 and Figure 12. In each figure we list the results of the adaptive scheme, best and worst individual transform in terms of coding efficiency. We see that even though with only four choices of lifting transforms, the adaptive lifting still yields performance gain over the best transform and significant gain over the worst transform. It should be noted that the best choice of an individual transform might vary for different sequences due to different characteristics. For example, the (1,3) transform is better than the other three transforms in the Stefan test sequence, while it is the worst in the Foreman scene change sequence. Hence it is remarkable to have an adaptive lifting scheme that achieves consistently better performance than all those individual transforms without prior knowledge of sequence characteristics.

With respect to computational complexity, we only need to evaluate the smoothness of the local area with a simple criterion. So there is no significant additional complexity in this case.

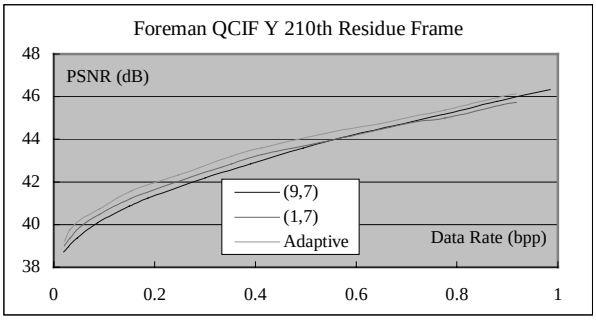

Figure 10. Coding efficiency for 210th frame

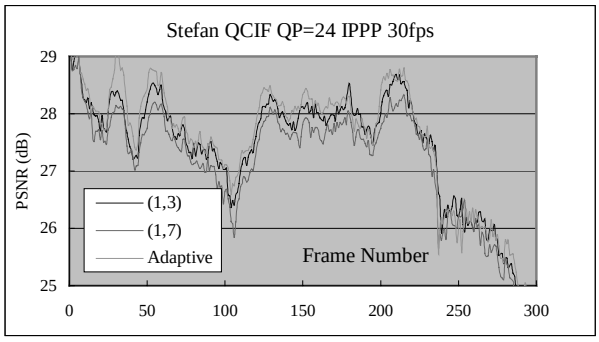

Figure 11. Performance comparison of adaptive lifting

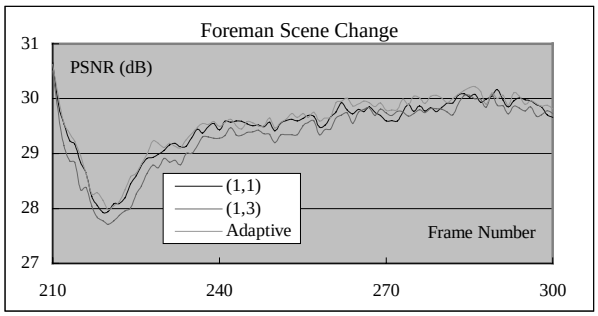

Figure 12. Performance of adaptive lifting at scene change

6. CONCLUSIONS AND FUTURE WORK

In this paper we present a novel spatially adaptive lifting scheme for video coding based on an update-first lifting structure. We have shown that no additional overhead information is needed in the scheme and performance improvements have been achieved.

In the future more accurate characteristics classification and transform are needed to further explore the advantage of adaptive lifting structures.

REFERENCES

[1]. W. Sweldens and P. Schroder, "Building your own wavelets at home," *Wavelets in Computer Graphics*, ACM SIGGRAPH Course Notes, pp.15-87, 1996.

[2]. R. Claypoole, G. Davis, W. Sweldens and R. Baraniuk, "Nonlinear wavelet transforms for image coding via lifting,," *Proc. 31st Asilomar Conf. Signals, Syst., Compu.* vol. 1, pp. 662-667, 1997

[3]. I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Applicat.*, vol. 4, no.3, pp. 247-269, 1998.

[4]. G. Piella and H. Heijmans, "Adaptive lifting schemes with perfect reconstruction," *IEEE Trans. on Signal Processin,*, vol. 50, no. 7, pp. 1620-1630, July, 2002.

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