Scalable Video Coding Using Adaptive Directional Lifting-Based Wavelet Transform

Chao-Hsiung Hung
Department of Electronics Engineering
National Chiao-Tung University
Hsinchu
morning.ee94g@nctu.edu.tw

Hsueh-Ming Hang
Department of Electronics Engineering
National Chiao-Tung University
Hsinchu
hmhang@mail.nctu.edu.tw

***Abstract*—3-D wavelet video coding technique provides an elegant solution to the multimedia transmission scalability requirement. It uses motion-compensated temporal filtering (MCTF) for temporal decomposition and 2-D discrete wavelet transform (2-D DWT) for spatial decomposition. MCTF decomposes video frames into temporal high-pass residuals and temporal low-pass residuals. The temporal high-pass residuals have similar statistical properties as the motion compensation residuals but the temporal low-pass ones are similar to the original images. The adaptive directional lifting-based wavelet transform (ADLWT) finds the best filter direction in an image local area and provides better energy compaction than DWT. In this paper, we use ADLWT to replace 2-D DWT to improve the performance of 3-D wavelet video coding.**

I. Introduction

Multimedia applications become very popular in recent years. Because different users have different bandwidth constraints, such as mobile video or digital TV, scalable Video Coding (SVC) techniques are developed to solve this problem. The 3-D wavelet video coding technique achieves spatial, temporal, and SNR scalabilities in an elegant way. A typical 3-D wavelet video coding schemes contains a Motion-Compensated Temporal Filter (MCTF) unit to reduce temporal redundancy and a 2-D DWT unit to reduce spatial redundancy. It provides good spatial and temporal scalability owing to the mutliresolution property of wavelet transform. The arithmetic coder compresses the acquired subbands and motion vectors well and achieves SNR scalability.

MCTF removes temporal redundancy efficiently by constructing a temporal decomposition along the motion trajectories. Ohm proposed a block matching motion compensation classifying pixels into connected/unconnected and cover/uncovered ones [1]. Pesquet-Popescu and Bottreau proposed half-pixel accuracy motion estimation combining lifting structure with Haar filters [2]. Luo *et al.* also proposed similar structure with biorthogonal 5/3 filters [3]. Secker and Taubman showed that the 5/3 filters lead to better coding results [4]. Xiong *et al.* proposed the barbell lifting scheme to solve the mismatch of motion vectors between the prediction and the update steps. MCTF decomposes video sequences into temporal low-pass and high-pass residuals. The temporal high-pass residuals have similar statistical characteristics as the motion-compensated residuals and the low-pass ones are similar to the original images. The low-pass residuals contain most energy and have strong impact on the compression performance.

The 2-D DWT applies two 1-D DWTs along horizontal and vertical directions, separately. However, the texture directions of images and residual signals often do not align with the exact horizontal or vertical directions, and thus it does not present these signals well. Often, the 2-D DWT generates many large magnitude high-frequency coefficients. If we quantize these coefficients to zero at low bit transmission rates, the reconstructed image shows Gibbs artifacts along edges. Ding *et al.* proposed the directional-adaptive 1-D DWT with interpolation called ADLWT to match the image edges. It tries to find the most appropriate direction (of a block) and the block size by minimizing the high-pass energy (prediction error) under the given constraint bits [6]. Chang and Girod proposed another type of directional-adaptive 1-D DWT without interpolation [7]. The directional-adaptive 1-D DWT also provides multiresolution property but achieves better compression performance than 2-D DWT.

The rest of the paper is organized as follows. Section II introduces the structure of MCTF and section III introduces the structure ADLWT. We show experimental results in Section IV and conclude this paper by Section V.

II. Motion-Compensated Temporal Filtering

There are three steps in MCTF, polyphase decomposition, prediction step, and update step. The polyphase decomposition splits the input frames F_k into odd frames F_{2i} and even frames F_{2i+1}. The prediction step generates the high-pass residuals H_i by predicting F_{2i+1} from F_{2i} and F_{2i+2}

$$H_i = F_{2i+1} - \frac{1}{2}(MC(F_{2i}, MV_{2i+1\to 2i}) + MC(F_{2i+2}, MV_{2i+1\to 2i+2})) \quad (1)$$

This work was supported in part by the NSC, Taiwan under Grants 98-2221-E-009 -076 and NSC 96-2221-E-009-063.

where $MV_{2i+1\to 2i}$ is the motion vector from frame F_{2i+1} to F_{2i}. MC(F_{2i}, $MV_{2i+1\to 2i}$) is the motion compensation process using motion vector $MV_{2i+1\to 2i}$ to generate the predicted pixels of F_{2i+1} from F_{2i}. Then the update step generates the low-pass residuals by updating F_{2i} from H_{i-1} and H_i

$$L_i = F_{2i} + \frac{1}{4}(MC(H_{i-1}, MV_{2i\to 2i-1}) + MC(H_i, MV_{2i\to 2i+1})) \qquad (2)$$

Through one level of MCTF, video frames are decomposed into temporal low-pass and high-pass residuals. Another level of MCTF decompose low-pass residuals again and iteratively to achieve temporal scalability.

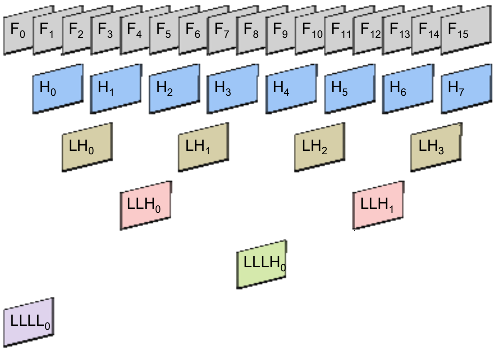

Figure 1. Temporal residuals after 4-level MCTF applied to 16 input frames, F_0~F_{15}.

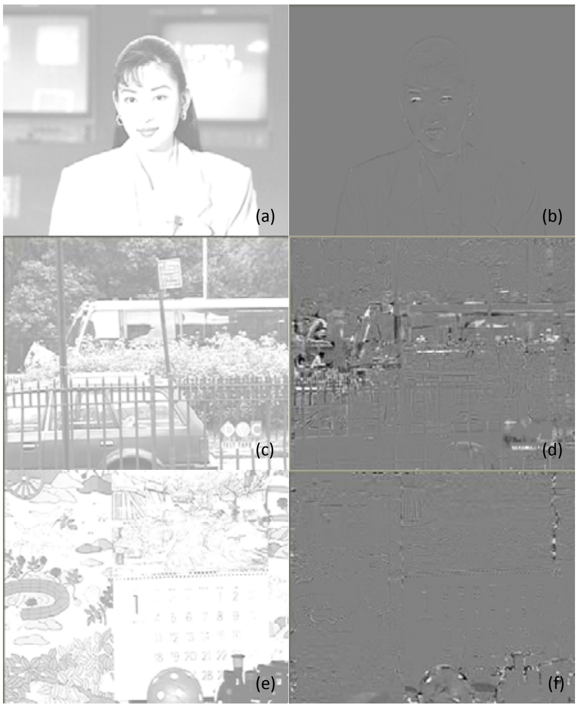

Figure 2. Temporal residuals of 3 test video sequences. (a) LLLL$_0$ of *Akiyo*, (b) LLLH$_0$ of *Akiyo*, (c) LLLL$_0$ of *Bus*, (d) LLLH$_0$ of *Bus*, (e) LLLL$_0$ of *Mobile*, (f) LLLH$_0$ of *Mobile*.

Figure 1 shows the temporal residuals after 4-level MCTF applied to 16 input frames, F_0~F_{15}. The results are 1 low-pass residuals LLLL$_0$, and 15 high-pass residuals, LLLH$_0$, LLH$_0$~ LLH$_1$, LH$_0$~ LH$_3$, H$_0$~ H$_7$. Figure 2 shows the low-pass and high-pass residuals of different test video sequence. The low-pass residuals are similar to the original images but the high-pass ones are similar to the motion-compensated residuals. The motion-compensated residuals have different spatial characteristics from the original images [8].

III. Motion-Compensated Temporal Filterin

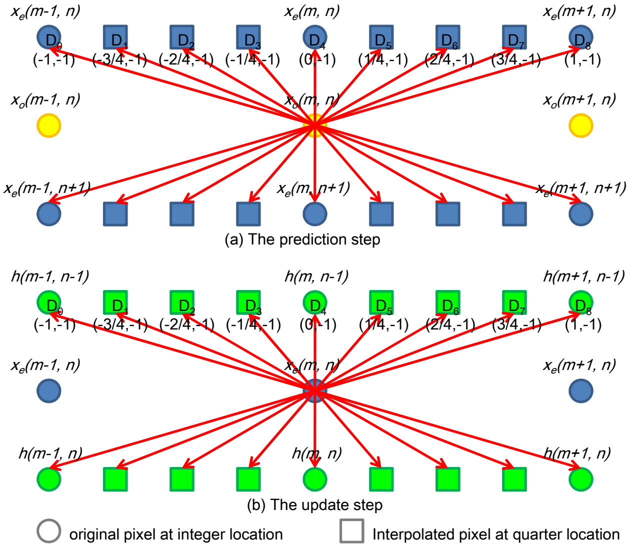

Figure 3. The lifting directions in ADLWT [6].

Lifting scheme is an efficient method to implement DWT It also consists of polyphase decomposition, prediction steps and update steps. Since 2-D DWT applies DWT horizontally and vertically, the prediction and the update steps are also performed in the horizontal or vertical directions. ADLWT tries to find the best direction for lifting scheme by minimizing the prediction error in a local area.

We denote an image as 2-D signal $x(m, n)$, where m, n ∈ Z. In the polyphase decomposition, we first decompose the rows of x into x_e and x_o

$$\begin{cases} x_e(m,n) = x(m,2n) \\ x_o(m,n) = x(m,2n+1) \end{cases} \qquad (3)$$

Then, we interpolate 3 quarter pixels between integer pixels in x_o in Figure 3(a). We calculate the high-pass coefficients h in the prediction step as follows

$$h(m,n) = x_o(m,n) - P_i(x_e) \qquad (4)$$

$$P_i(x_e) = \alpha_j(x(m+d_{xi}, 2n+1+d_{yi}) + x(m-d_{xi}, 2n+1-d_{yi})) \qquad (5)$$

where $P_i(x_e)$ is a linear combination of pixels in x_e, including the original and the interpolated ones, along direction D_i=(d_{xi}, d_{yi}), i=0~8 in Figure 3(a). α_j is the predictor weighting coefficient depends on the used spatial wavelet filter.

In the update step, we first interpolate h similar as above in Figure 3(b). Then, we compute the low-pass coefficients l as follows

$$l(m,n) = x_e(m,n) - U_i(h) \qquad (6)$$

$$U_i(x_o) = \beta_j(h(m+d_{xi}, 2n+d_{yi}) + h(m-d_{xi}, 2n-d_{yi})) \qquad (7)$$

where $U_i(x_o)$ is a linear combination of pixels in h, including the original and the interpolated ones, along direction D_i=(d_{xi},

d_{yi}), i=0~8 in Figure 3(b).β_j is the update weighting coefficient depends on the spatial wavelet filter.

After finishing the ADLWT along each row we separate l and h apart. Then, we apply ADLWT along each column inside l and h similarly. This is one level of ADLWT. We apply another level of ADLWT to the LL subband again and iteratively to achieve spatial multiresolution decomposition.

In order to decide the best direction in each local region of an image, we partition an image into small blocks and find the best direction for each block. The best direction is the direction with the minimum prediction error. ADLWT constructs an optimal quad-tree block partition. We next consider merging small blocks into a larger block. By minimizing the Lagrange cost function including the prediction error and the required bits, we decide the cost of merging blocks. There are two types of required bits here. One considers only the bits for coding the selected directions [7] and the other considers the total coded bits for both the acquired coefficients and selected directions [6]. These two cost functions produce almost the same coding performance except for very low bit rate (≤0.1 bpp) [9]. Therefore, for simplicity, we consider the bits of selecting direction in the cost function.

IV. EXPERIMENTAL RESULTS

We first apply the 4-level MCTF to the test video sequence and results in 5 temporal layers. The first approach is to apply 2-D DWT to each residual, including both the low-pass and the high-pass ones, and we call this scheme "2-D DWT". The second approach is applying ADLWT to the temporal low-pass residuals and 2-D DWT to the temporal high-pass ones and we call this scheme "ADLWT". In this scheme, we apply 2-level 2-D DWT or 2-level ADLWT to each residual. We use only the bit cost function in selecting direction and block partition when constructing the optimal tree .The minimum block size is 4×4 and the maximum block size is 64×64. The test video sequences are all CIF format and 30 fps. We take the well-known scalable wavelet codec, Vidwav [10], as the reference software and the platform of our proposed scheme.

Figure 4 shows that the proposed scheme "ADLWT" has better results than "2-D DWT", especially at low bit-rates. The gain of PSNR is about 0.38dB for *Akiyo*, 0.25dB for *Bus*, 0.34dB for *Flowergarden*, 0.13dB for *Foreman*, 0.12dB for *Hall_monitor*, 0.19dB for *Mobile*, 0.26dB for *Mother_daughter*, 0.20dB for *News*, 0.14dB for *Silent*, and 0.32dB for *Table*, at 128 kbps.

Figure 5 compares the PSNR of each frame at 128 kbps. From Figure 5, we see that "ADLWT" has comparable or better PSNR than "2D DWT" for nearly all frames.

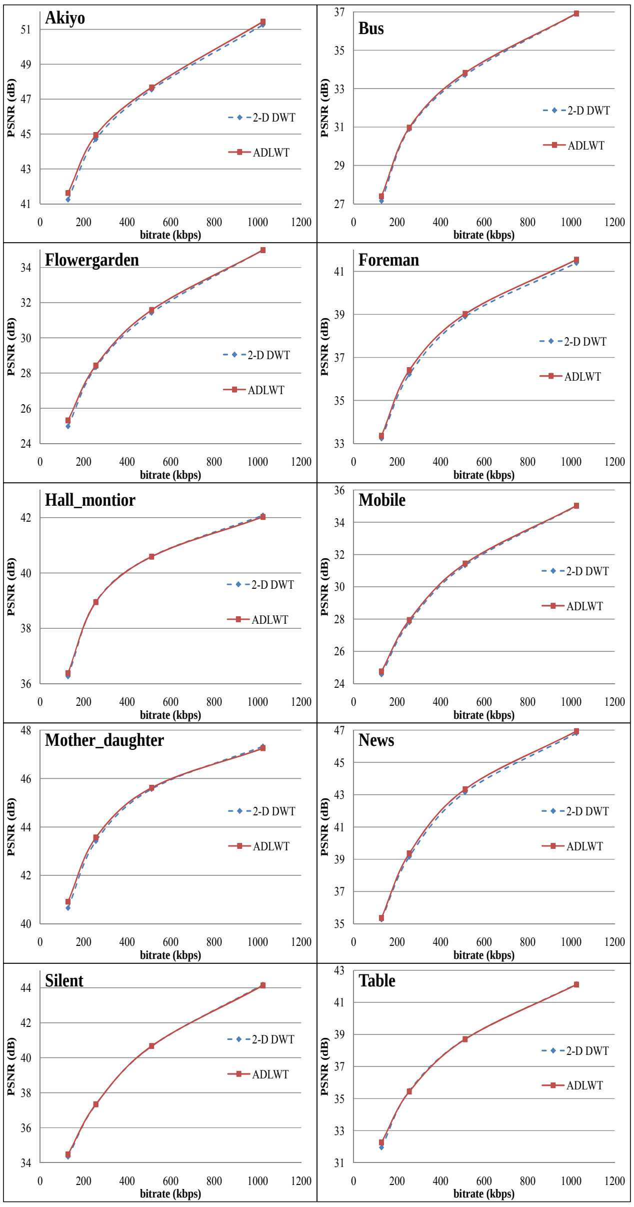

Figure 4. PSNR of two schemes.

TABLE I. BIT RATE OF SIDE INFORMATION FOR THE OPTIMAL TREE

Test Video	*Akiyo*	*Bus*	*Flowergarden*	*Foreman*	*Hall_monitor*
Bit Rate(kbps)	1.05	0.93	0.85	0.71	0.59
Test Video	*Mobile*	*Mother_daughter*	*News*	*Silent*	*Table*
Bit Rate(kbps)	1.76	0.86	1.11	0.55	1.01

TABLE I shows the bit rate for side information of block partition and selected direction for each test video sequence. We see that the bit rate required for side information is very small compared to the coding bit rate.

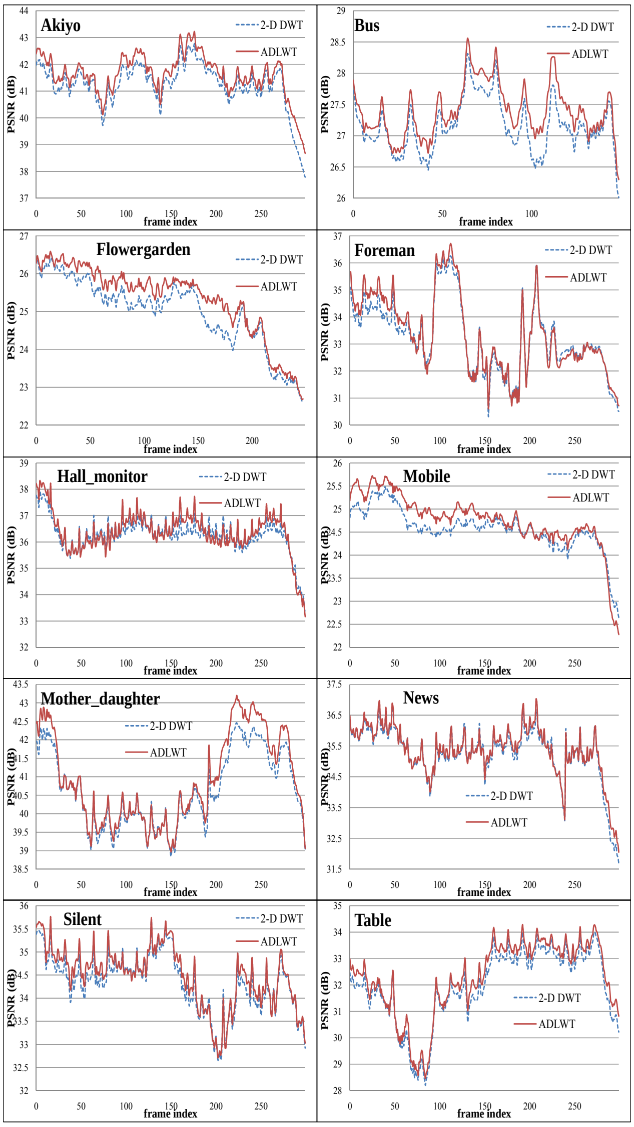

Figure 5. PSNR of each frame of two schemes at 128kbps.

TABLE II. BD-PSNR AND BD-RATE OF EACH TEST VIDEO SEQUENCE

Test Video	*Akiyo*	*Bus*	*Flowergarden*	*Foreman*	*Hall_monitor*
BD-PSNR(dB)	0.21	0.10	0.14	0.16	0.01
BD-BR(%)	-4.05	-2.01	-2.85	-3.99	0.55
Test Video	*Mobile*	*Mother_daughter*	*News*	*Silent*	*Table*
BD-PSNR(dB)	0.12	0.10	0.16	0.02	0.02
BD-BR(%)	-2.39	-3.04	-2.75	-0.55	-2.45

TABLE II shows the Bjontegaard delta bit rate (BD-BR) and Bjontegaard delta PSNR (BD-PSNR) [11]. ADLWT shows better performance than 2-D DWT with typically the bit rate advantage of 2% to 3%.

V. CONCLUSION

This paper adopts ADLWT for temporal low-pass residuals in 3-D wavelet video coding. The proposed scheme provides better performance than 2-D DWT, especially at low bit rates. ADLWT also provides comparable or better PSNR on most reconstructed frames than 2-D DWT. The bit rate of side information in ADLWT is very small comparing to the total transmission bit rate.

REFERENCES

[1] J. R. Ohm, “Three-dimensional subband coding with motion compensation,” *IEEE Trans. Image Process.*, vol. 3, no. 9, pp. 559–571, Sep. 1994.

[2] B. Pesquet-Popescu and V. Bottreau, “Three-dimensional lifting schemes for motion compensated video compression,” in *Proc. ICASSP*, May 2001, pp. 1793–1796.

[3] L. Luo, F. Wu, S. Li, Z. Xiong, and Z. Zhuang, “Advanced motion threading for 3-D wavelet video coding,” *Signal Process.: Image Commun.*, vol. 19, no. 7, pp. 601–616, Aug. 2004.

[4] A. Secker and D. Taubman, “Lifting based invertible motion adaptive transform, LIMAT, framework for highly scalable video compression,” *IEEE Trans. Image Process.*, vol. 12, pp. 1530–1542, Dec. 2003.

[5] R. Xiong, J. Xu, F. Wu, and S. Li, “Barbell-Lifting Based 3-D Wavelet Coding Scheme,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, Sep. 2007.

[6] W. Ding, F. Wu, X. Wu, S. Li, and H. Li, “Adaptive directional lifting-based wavelet transform for image coding,” *IEEE Trans. Image Processing.*, vol. 16, no. 2, pp. 416–427, Feb. 2007.

[7] C.-L. Chang and B. Girod, “Direction-adaptive discrete wavelet transform for image compression,” *IEEE Trans. Image Processing.*, vol. 16, no. 5, pp. 1289–1302, May 2007.

[8] F. Kamisli and J.S. Lim, “Transforms for the motion compensation residual,” *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 789–792, April 2009.

[9] T. Xu, C.-L. Chang, and B. Girod, “Scalable direction representation for image compression with direction-adaptive discrete wavelet transform,” in *Proc. Visual Communication and Image Processing*, 2007.

[10] R. Xiong, X. Ji, D. Zhang, and J. Xu, “Vidwav wavelet video coding specifications,”ISO/IEC JTC1/SC29/WG11 MPEG, M12339, 2005.

[11] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD Curves*, document VCEG-M33, ITU-T SG16 Q.6 VCEG, Apr. 2001.