IMPROVED TEMPORAL FILTERING SCHEME TO REDUCE DELAY AND DISTORTION FLUCTUATION IN 3D WAVELET BASED VIDEO CODING

Vidhya Seran, Lisimachos P. Kondi

332 Bonner Hall, Dept. of Electrical Engineering
State University of New York at Buffalo, Buffalo, NY 14260
Tel: (716) 645-2422 ext. 1147 Fax: (716) 645-3656
Email: { vseran,lkondi }@eng.buffalo.edu

ABSTRACT

In this work, we propose a new 3D temporal filter set that modifies the existing temporal filter structure, to achieve minimum delay and improve compression efficiency. The modified temporal filter set does not have any constraints on group of frame (GOF) size and it can be processed independently without using future or past GOFs. The proposed filter will not exhibit any boundary effects and has less distortion fluctuation.

1. INTRODUCTION

As an alternative approach to traditional predictive coding for video, wavelet based video coding has emerged as a viable option. A three-dimensional wavelet transform is employed to exploit the temporal redundancy in the video source. There are two main theoretical developments that promise efficient wavelet-based video codecs: Temporal filtering using lifting and motion compensation in the Overcomplete Discrete Wavelet Transform. The three-dimensional wavelet decomposition can be performed in two ways:

1. Two-dimensional spatial filtering followed by temporal filtering (2D+t) [1, 2, 3]

2. Temporal filtering followed by two-dimensional spatial filtering (t+2D). [4, 5, 6].

In addition to providing high coding efficiency, 3D wavelet coding has good spatial and temporal scalabilities. Though 3D schemes offer drift-free scalability, there is an increase in delay requirements. A group of frames (GOF) must be available at the encoder to start the coding process. The receiver can start decoding only after receiving all the frames in a GOF. Thus, 3D video coding schemes offer better performance but also relax the causality of the system.

We propose a new 3D temporal filter set, which offers minimum delay and reduced distortion fluctuation within a GOF [7]. The proposed filter will not have any boundary effects at the GOF. Unlike Haar or longer filters like 5/3, which require GOFs to be in some power of 2, the proposed filter does not have any constraints on GOF length. The length of the GOF can vary from five to any number of frames depending on delay requirements. The proposed filter set is perfectly invertible. This can be applied to both t+2D and 2D+t schemes. In this paper, we propose update steps for the new 3D temporal filter set to improve compression efficiency.

The rest of the paper is organized as follows: In Section 2, we explain the motion-compensated temporal filtering (MCTF) using lifting and related problems. In Section 3, we discuss our proposed method: a new filter set to minimize delay. Finally, in Section 4, we present the simulation results.

2. MOTION-COMPENSATED TEMPORAL WAVELET TRANSFORM USING LIFTING

In this work, the motion estimation/motion compensation is done using the low band shift method [8]. Hence our proposed coder 3D-coder is in 2D+t category. An input frame is decomposed in the critically sampled DWT domain and the reference frame is transformed using ODWT. The wavelet coefficients are rearranged to form wavelet blocks such that the related coefficients in all scales and orientations are included in each wavelet block. This can be pictorially represented as in Figure 1. The video frames of size $N \times M$ are subjected to a three level decomposition. So, for each cofficient at level L in DWT domain, there are 4^L coefficients in ODWT. Thus, there are 4^L phases in level L.Motion estimation is done using the block matching technique. Thus, the wavelet block of the reference frame is matched with the wavelet blocks of the current frame in a search window W, and the reference wavelet block is selected by minimizing the Mean Absolute Difference (MAD).

Lifting allows the incorporation of motion compensation in temporal wavelet transforms while still guaranteeing perfect reconstruction. Let us consider as an example biorthogonal 5/3 wavelet transform using lifting:

$$\begin{aligned} h_k &= f_{2k+1} - \frac{1}{2}[f_{2k} + f_{2k+2}] \quad (1) \\ l_k &= f_{2k} + \frac{1}{4}[h_k + h_{k-1}] \end{aligned}$$

where f_k denotes frame k and h_k and l_k represent the high-pass and low-pass subband frames. It is possible to modify the above equations in order to incorporate motion compensation. Let $W_{i \to j}(f_i)$ denote the motion-compensated mapping of frame f_i into frame f_j. Thus, the operator $W_{i \to j}(.)$ gives a per pixel mapping between two frames. No particular motion model is assumed. Thus, the above equations are modified as

$$h_k = f_{2k+1} - \frac{1}{2}[W_{2k \to 2k+1}(f_{2k}) + W_{2k+2 \to 2k+1}(f_{2k+2})]$$

$$l_k = f_{2k} + \frac{1}{4}[W_{2k-1 \to 2k}(h_{k-1}) + W_{2k+1 \to 2k}(h_k)]$$

These equations correspond to taking the 5/3 transform along the motion trajectories [4]. In the lifting operation, the prediction residues (temporal highpass subbands) are used to update the reference frame to obtain a temporal low subband. We will refer this as an update step in the following discussions.

With a increase in temporal decomposition level for longer filters like 5/3, the update step depends on more number of past and future frames. Hence the grouping of video frames of finite size becomes difficult for longer filters. If a video sequence is divided into a number of GOFs, that are processed independently, without using frames from other GOFs, high distortion will be introduced at the GOF boundaries. In order to reduce this variation at the boundaries, the encoder needs to use frames from past and future GOFs. Thus, it is clear that the introduced delay (in frames) is greater than the number of frames in the GOF. The encoding and decoding delay will be very high, as the encoder has to wait for future GOFs. In [6], the distortion at the boundaries is reduced to some extent by using a sliding window approach.

To minimize delay and decrease ghosting artifacts in the low pass temporal subbands, the update step for 5/3 filter can be entirely skipped [4, 9]. But at full frame rate resolution, 1/3 filter suffers in compression efficiency compared to the 5/3 filter.

So far, an overview of motion compensation ODWT and temporal filtering were discussed.

3. PROPOSED NEW 3D TEMPORAL FILTER SET

In 3D coding schemes, high level of compression is achieved by applying temporal filter for a group of frames. The number of frames in a buffer will increase with the length of the filter and the number of temporal decomposition levels. This introduces a delay both at the encoder and decoder.

We propose a new filter set that minimizes delay and performs at par with longer filters. In this filter design, low-pass temporal frames are created at the beginning and at the end of a group of frame during the first level temporal decomposition. We first describe the proposed filter design without including any update step. Thus, the lowpass temporal frames are unfiltered original video frames. Let N be the length of the GOF and L be the maximum number of temporal decomposition levels. Let l ($1 \leq l \leq L$) be the l^{th} level temporal decomposition. The steps to be performed for proposed filter set is given below:

1. Get No of frames in a Group, N and temporal level, L
2. At temporal level l ($1 \leq l \leq L$), n = $round$ ($\frac{N}{2}$)
3. Get HighPass temporal frames: If $N > 2$, then for k = $0, 1, \cdots, (n-1)$

   $$h_k = f_{2k+1} - \frac{1}{2}[W_{2k \to 2k+1}(f_{2k}) + W_{2k+2 \to 2k+1}(f_{2k+2})]$$

   If $N == 2$, then for k=0

   $$h_k = f_{2k+1} - [W_{2k \to 2k+1}(f_{2k})]$$

4. Get temporal Lowpass frames: If $N > 2$
   - if ($modulo(N, 2)$ == 0), k = $0, 1, \cdots, (n-1)$ and k = $\frac{N}{2}$
   - if ($modulo(N, 2)$!= 0), k = $0, 1, \cdots, (n-1)$

   If (N == 2), k = 0

   $$l_k = f_{2k}.$$

5. For next temporal level, l : Set N = no of Low Frames and f = lowpass frames and Goto Step 2

3.1. Update Step for proposed filter

The lifting update step is included to further increase the compression efficiency without increasing the delay. The update step can be performed using the following steps:

- Get temporal Lowpass frames: If $N > 2$
  - if ($modulo(N, 2)$ == 0), k = $0, 1, \cdots, (n-1)$ and k = $\frac{N}{2}$
  - if ($modulo(N, 2)$!= 0), then k = $0, 1, \cdots, (n-1)$

For $k = 0$

$$l_k = f_{2k} + w_1[W_{2k+1\rightarrow 2k}(h_k)]$$

For $k = 1$ to $((n-1)-1)$

$$l_k = f_{2k} + w_2[(W_{2k-1\rightarrow 2k}(h_{k-1})) \\ +(W_{2k+1\rightarrow 2k}(h_k)+)]$$

For $k = (s-1)$

$$l_k = f_{2k} + w_3[W_{2k-1\rightarrow 2k}(h_{k-1})]$$

If $(N == 2)$, $k = 0$

$$l_k = f_{2k} + w_1[W_{2k+1\rightarrow 2k}(h_k)]$$

The weights w_1, w_2 and w_3 are set according to the motion modeling. If we set w_1 and w_3 as zero, then the first and last low pass frame inside the GOF with not be updated. This can be used when we use frames from next GOF, so that the delay will be always GOF or GOF+1. The weights can be adaptively selected based on the energy content in the high pass frames [10].

Also, Figure 2 is used to explain our proposed scheme for GOF=8. One should note that four levels of temporal decomposition in the proposed filter set actually correspond to three levels in Haar or longer filters, in the sense that the same number of lowpass frames exists in both the cases.

4. EXPERIMENTAL RESULTS

We have considered four standard test sequences, two in SIF (352×240) resolution: "Football" and "Flower Garden" and two in QCIF (176×144) resolution: "Foreman" and "Susie". A Daubechies $(9, 7)$ filter with a three level decomposition is used to compute the wavelet coefficients. The motion estimation is performed in the overcomplete wavelet domain using the block matching technique for integer pixel accuracy. A 16×16 wavelet block is matched in a search window of $[-16, 16]$.

In the experimental results, we compared the performance of the proposed temporal filters with 5/3 filter. The temporal subbands are compressed using 2D-SPIHT coder [11] and there is no rate control. All temporal subbands are coded at the same rate. The proposed temporal filter with update with a GOF of eight frames and nine frames are compared with 5/3 with a GOF of 16 frames. All the GOFs are processed independently and no future GOF is used in any case. Table 1 gives the average PSNR values of the Y component for an encoded bit rate of 380 kbps for QCIF sequences and 1 Mbps for SIF sequences. The average PSNR will increase if we use optimal bit allocation [7]. It can be seen from the Table 1 that the proposed filter outperforms the 5/3 filters, while exhibiting lower delay requirements. The "Susie" sequence was decoded at 380 kbps and the PSNR of each frame is plotted in Figure 3. The forty eigth frame of susie sequence is shown in Figures 4 and 5 for proposed GOF=9 case and 5/3 GOF=16 case, respectively. As can be seen the picture quality is GOf=9.

5. CONCLUSION

We have proposed a novel temporal filter set with motion compensation for 3D wavelet-based video coding. The filter set described offers flexible features for compression efficiency and delay requirements. Our experimental results show, the effectiveness of the proposed scheme. An adaptive update step with optimal bit allocation schemes can be developed to enhance the overall system performance.

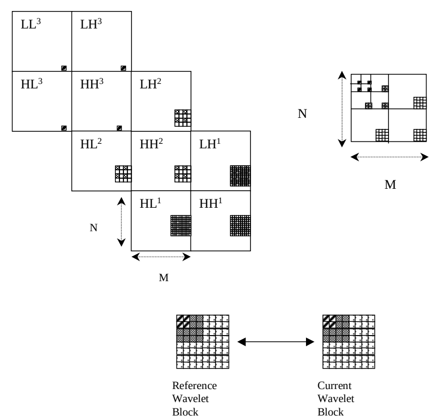

Fig. 1. Overcomplete Wavelet Transform (ODWT) vs Discrete Wavelet Transform (DWT) is shown for a frame of size $N \times M$. The wavelet block formation formation is explained using the patched lines.

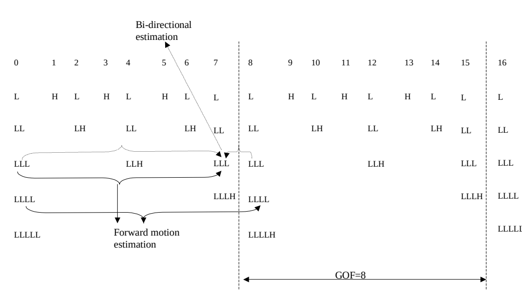

Fig. 2. Proposed Filter with GOF=8

Table 1. Average PSNR values of Y component

Sequence	5/3 Filter (GOF=16)	Proposed Filter (GOF=8)	Proposed Filter (GOF=9)
Football	27.63 dB	27.59 dB	28.28 dB
Foreman	31.03 dB	31.05 dB	31.43 dB
Susie	37.52 dB	37.72 dB	38.13 dB
Garden	22.34 dB	22.33 dB	22.68 dB

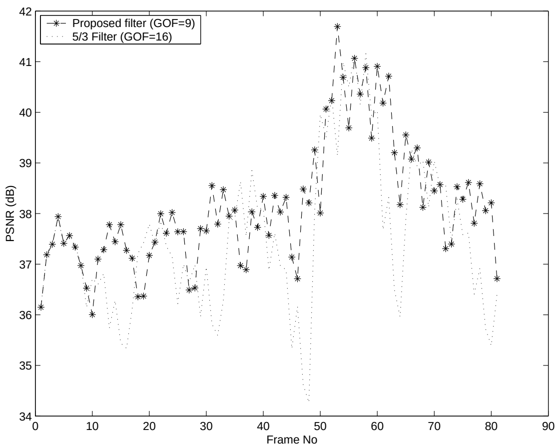

Fig. 3. Susie seqeunce at 380 kbps .

Fig. 4. Proposed GOF=9

Fig. 5. 5/3 GOF=16

6. REFERENCES

[1] Y. Andreopoulos, A. Munteanu, J. Barbarien, M. van der Schaar, J. Cornelis, and P. Schelkens, "In-band motion compensated temporal filtering," *Signal Processing: Image Communication (special issue on "Subband/Wavelet Interframe Video Coding")*, vol. 19, pp. 653–673, August 2004.

[2] Y. Wang, S. Cui, and J. E. Fowler, "3D video coding using redundant-wavelet multihypothesis and motion-compensated temporal filtering," in *Proceedings of the IEEE International Conference on Image Processing*, Barcelona, Spain, 2003, vol. 2, pp. 755–758.

[3] X. Li, "Scalable video compression via overcomplete motion compensated wavelet coding," *Signal Processing: Image Communication (special issue on "Subband/Wavelet Interframe Video Coding")*, vol. 19, pp. 637–651, August 2004.

[4] A. Secker and D. Taubman, "Lifting based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Transactions on Image Processing*, vol. 12, pp. 1530–1542, December 2003.

[5] S. T. Hsiang and J. W. Woods, "Embedded video coding using motion compensated 3-D subband/wavelet filter bank," in *Proceedings of the Packet Video Workshop*, Sardinia, Italy, May 2000.

[6] A. Golwelkar and J. Woods, "Scalable video compression using longer motion compensated temporal filters," in *Proc. SPIE VCIP*, 2003, vol. 5150, pp. 1406–1416.

[7] V. Seran and L.P. Kondi, "3D based video coding in the overcomplete discrete wavelet transform domain with reduced delay requirements," *Proceedings of the IEEE International Conference on Image Processing*, September 2005.

[8] H. W. Park and H. S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Transactions on Image processing*, vol. 9, pp. 577–587, April 2000.

[9] M. Van der Schaar and D. Turaga, "Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2003.

[10] N. Mehrseresht and D. Taubman, "An efficient content adaptive motion compensation 3D-DWT with enhanced spatial and temporal scalability," in *Proceedings of the IEEE International Conference on Image Processing*, 2004.

[11] A. Said and W. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 243–250, June 1996.