New Temporal Filtering Scheme to Reduce Delay in Wavelet-Based Video Coding

Vidhya Seran and Lisimachos P. Kondi, *Member, IEEE*

***Abstract*—Scalability is an important desirable property of video codecs. Wavelet-based motion-compensated temporal filtering provides the most powerful scheme for scalable video coding and provides high-compression efficiency that competes with the current state of art codecs. However, the delay introduced by the temporal filtering schemes is sometimes very high, which makes them unsuitable for many real-time applications. In this paper, we propose a new temporal filter set to minimize delay in 3-D wavelet-based video coding. The new filter set gives a performance at par with existing longer filters. The length of the filter can vary from two to any number of frames depending on delay requirements. If the frames are processed as separate groups of frames (GOFs), the proposed filter set will not have any boundary effects at the GOF. Experimental results are presented and conclusions are drawn.**

***Index Terms*—Motion-compensated temporal filtering, wavelet-based video coding.**

I. Introduction

THE popularity of multimedia applications demands support for different receivers that operate at different bit rates, resolution, and complexity. This mandates the need for a scalable video coder with high-compression efficiency. All current video compression standards are based on the motion-compensated discrete cosine transform (MC-DCT) paradigm and its variations. This paradigm has been in use for over two decades and is widely used in a range of applications. Traditional video coders use the previous frame to perform motion estimation and compensation. Though they are less complex and have minimum coding delays, these coders lose their efficiency when subjected to scalability requirements.

Wavelet-based image coding has the very best coding efficiency and provides SNR scalability, besides resolution scalability. Wavelet-based compression is known to outperform DCT-based compression for image coding. The popular JPEG-2000 image compression standard is also wavelet based. In order to have efficient video compression, the temporal redundancy in the video data has to be properly exploited. This has kindled the minds of many researchers to explore the possibilities of using wavelets in video coding.

Initial approaches to applying motion compensation to the discrete wavelet transform (DWT) were not very successful. If motion compensation is performed in the spatial domain, as in MC-DCT-based codecs, and the prediction error is encoded using DWT instead of DCT, compression efficiency will not be good since the DWT is not well suited to the statistics of the prediction error. Also, band-to-band motion compensation in the DWT domain is not efficient because the DWT is not shift-invariant and the wavelet coefficients of the current frame cannot be accurately predicted from the coefficients of the previous frame. This led to the use of the overcomplete wavelet decomposition to overcome the aliasing problem. Several works have been recently proposed for motion estimation and compensation in the overcomplete wavelet domain [1]–[4]. The drift introduced by the predictive coding or closed loop scheme can be overcome by drift control methods [5], [6].

In 3-D wavelet-based video coding schemes, the sub-band decomposition is extended to the temporal domain and it employs a 3-D wavelet transform. Thus, temporal redundancy in the video source is exploited using temporal filtering. Three-dimensional filtering avoids the predictive feedback loop, and, hence, 3-D schemes offer drift-free scalability. The multiresolutional nature of the wavelet coding provides spatial and temporal scalability. Though simple 3-D methods are a direct extension of 2-D wavelet coding, the temporal correlation moves away from the temporal axis with motion. That is, without any motion compensation, temporal transforms produce low-quality temporal sub-bands with ghosting artifacts and high-energy distribution in the high-pass sub-bands. This decreases the coding efficiency and is undesirable when temporal scalability is of interest.

The performance of the 3-D coder is improved by incorporating motion compensation in temporal filtering. The main theoretical development that promises efficient 3-D wavelet-based video codecs is motion-compensated temporal filtering (MCTF) using lifting. Among the early works on MCFT are [7] and [8]. Motion-compensated lifting was first introduced in [9] and [10]. The MCTF can be performed in two ways:

1) two-dimensional spatial filtering followed by temporal filtering (2-D +t) [11]–[13];
2) temporal filtering followed by 2-D spatial filtering (t+2-D) [14]–[16].

The resulting wavelet coefficients can be encoded using different algorithms like 3-D-SPIHT [17] or 3-D-ESCOT [18]. Several enhancements have been recently made to the MCTF schemes presented either by introducing longer filters or by

Manuscript received June 16, 2006; revised August 12, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Giovanni Poggi.

The authors are with the Department of Electrical Engineering, The State University of New York at Buffalo, Buffalo, NY 14260 USA (e-mail: vseran@eng.buffalo.edu; lkondi@eng.buffalo.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2007.909316

optimizing the operators involved in the temporal filter [11], [12], [16], [19]. Recently, a JPEG2000-compatible scalable vide compression scheme has been proposed that uses the 3/1 filter for MCTF [20].

Though 3-D schemes offer drift-free scalability with high-compression efficiency, they introduce considerable delay, which makes them unsuitable for some real time video applications like tele-conferencing. In contrast to 2-D methods, frames cannot be encoded one by one but processing is done in groups of frames. Thus, a certain number of frames must be available to the encoder to start encoding. The number of frames required depends on the filter length. Similarly, the group of frames must be available at the receiver before decoding can start. Thus, 3-D video coding schemes offer better performance but also relax the causality of the system.

In this paper, we propose a new temporal filter set that offers minimum delay while retaining good compression efficiency in 3-D coding. The length of the filter can vary from two to any number depending on the delay requirements. The proposed filter set is perfectly invertible and can be applied to both t+2-D and 2-D+t schemes. For this filter set, we propose a new rate allocation scheme to minimize the total distortion of the reconstruced frames given a fixed rate budget. Some preliminary results have been presented in [21] and [22].

The rest of the paper is organized as follows. In Section II, we discuss the motion-compensated temporal filtering using lifting and the delay characteristics of the temporal filter used. In Section III, we discuss our new filter set to minimize delay. In Section IV, the rate allocation scheme for the proposed filter is explained. Finally, in Section V, we present the simulation results for different delay cases.

II. Motion-Compensated Temporal Wavelet Transform Using Lifting

Lifting allows the incorporation of motion compensation in temporal wavelet transforms while still guaranteeing perfect reconstruction. Any wavelet filter can be implemented using lifting. Let us consider as an example the Haar wavelet transform

$$\begin{aligned} h_k(x,y) &= f_{2k+1}(x,y) - f_{2k}(x,y) \\ l_k(x,y) &= \frac{1}{2}\left[f_{2k}(x,y) + f_{2k+1}(x,y)\right] \end{aligned} \tag{1}$$

where $f_k(x,y)$ denotes frame k and $h_k(x,y)$ and $l_k(x,y)$ represent the high-pass and low-pass sub-band frames.

Using lifting, the Haar filter along the motion trajectories with motion compensation can be implemented as [9], [10], [14]

$$\begin{aligned} h_k(x,y) &= f_{2k+1}(x,y) - W_{2k\to 2k+1}\left(f_{2k}(x,y)\right) \\ l_k(x,y) &= f_{2k}(x,y) + \frac{1}{2}W_{2k+1\to 2k}\left(h_k(x,y)\right) \end{aligned} \tag{2}$$

where $W_{i\to j}(f_i)$ 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 and this is applicable to any motion model.

For the case of the biorthogonal 5/3 wavelet transform, the analysis equations using motion-compensated lifting are

$$\begin{aligned} h_k(x,y) &= f_{2k+1}(x,y) \\ &\quad - \frac{1}{2}\left[W_{2k\to 2k+1}\left(f_{2k}(x,y)\right)\right. \\ &\quad \left. + W_{2k+2\to 2k+1}\left(f_{2k+2}(x,y)\right)\right] \end{aligned} \tag{3}$$

$$\begin{aligned} l_k(x,y) &= f_{2k}(x,y) \\ &\quad + \frac{1}{4}\left[W_{2k-1\to 2k}\left(h_{k-1}(x,y)\right)\right. \\ &\quad \left. + W_{2k+1\to 2k}\left(h_k(x,y)\right)\right]. \end{aligned} \tag{4}$$

In the lifting operation, the prediction residues (temporal high-pass sub-bands) are used to update the reference frame to obtain a temporal low sub-band. We will refer to this as the update step (4) in the following discussions.

If the motion is modeled poorly, the update step will cause ghosting artifacts to the low-pass temporal sub-bands. The update step for longer filters depends on a larger number of future frames. If a video sequence is divided into a number of fixed sized GOFs that are processed independently, without using frames from other GOFs, high distortion will be introduced at the GOF boundaries for longer filters. When longer filters based on lifting are used with symmetric extension, the distortion will be in the range of 4–6 dB (PSNR) at the GOF boundaries irrespective of the motion content or model used [12], [16], [23]. Hence, to reduce this variation at the boundaries, we need to use frames from past and future GOFs. Thus, it is observed 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 [16], the distortion at the boundaries for the 5/3 filter is reduced to some extent by using a sliding window approach. However, this clearly introduces delay both at the encoder and at the decoder. We should note that, even when the delay is high, if the motion is not modeled properly, then the low-pass temporal sub-bands will not be free from ghosting artifacts.

By skipping entirely the update step for 5/3 filter [14], [24], the analysis equations can be modified as

$$\begin{aligned} h_k(x,y) &= f_{2k+1}(x,y) \\ &\quad - \frac{1}{2}\left[W_{2k\to 2k+1}(f_{2k})(x,y)\right. \\ &\quad \left. + W_{2k+2\to 2k+1}(f_{2k+2})(x,y)\right] \\ l_k(x,y) &= f_{2k}(x,y). \end{aligned} \tag{5}$$

We refer to this filter set as the 1/3 transform.

Filters without update step will minimize the dependency on future frames thereby reducing the delay. Also, the low-pass temporal sub-bands are free from ghosting artifacts introduced by the update step. Hence, by avoiding the update step, we get high-quality temporal scalability with reduced delay. However, at full frame rate resolution, the 1/3 filter suffers in compression efficiency compared to the 5/3 filter.

So far, an overview of motion-compensated temporal filtering was discussed.

TABLE I
DELAY IN NUMBER OF FRAMES FOR HAAR AND 5/3 TEMPORAL FILTER

Delay	Haar	5/3
N_{en}	$2^L - 1$	$2^{L+1} - 2$
N_{dec}	2^{L-1}	$3 * 2^{L-1} - 1$
N_{end}	$2^L - 2$	$3 * (2^L - 1)$

A. Delay Analysis for MCTF

Delay requirements are very important for applications like tele-conferencing, video streaming and video surveillance. There are many sources of delay in a video codec system. In this section, we analyze the delay associated with the temporal filtering structure for MCTF filters. Let L be the number of temporal decomposition levels used. Let the encoding delay N_{en} be the maximum number of future frames that the encoder must receive before it can encode the current frame at level L. Let the decoding delay N_{dec} be the maximum number of future frames that the decoder must receive before it can start decoding the current frame. Let the end-to-end delay N_{end} be the maximum number of frames that the encoder has to capture and the frames needed by the decoder to display a frame. N_{en}, N_{dec}, and N_{end} in number of frames for the Haar and the 5/3 filter are summarized in Table I. The Haar filter offers less delay compared to the 5/3 filter but using longer filters increases the coding gain by 1–2 dB compared to the Haar filter. The coding efficiency is improved in longer filters because of the bi-directional prediction step used which reduces the energy in the high-pass temporal sub-bands. From (4), we can see that the update step involved in the temporal filtering introduces additional delay for the 5/3 filter. As discussed earlier, when the update is totally ignored, the delay can be reduced but the compression efficiency suffers. If we consider the 5/3 filter with three levels of temporal decomposition, the end-to-end delay is 21 frames. For an additional level of temporal decomposition, the delay is more than doubled. In [16], the Haar filter is used to reduce the encoding delay at the last stage of the temporal decomposition. However, this method is not flexible when delay requirements are considered. In [25], to reduce the delay in the 5/3 filter, some operators involved in the temporal filtering are removed based on the delay requirements. The scheme offers flexible MCTF structures according to the delay requirements but it affects the coding efficiency. Thus, the coding efficiency is decreased when the delay is reduced.

III. PROPOSED TEMPORAL SET FOR FLEXIBLE DELAY REQUIREMENTS

In 3-D coding schemes, a high level of compression efficiency is achieved by applying a temporal filter to 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. Our goal is to propose a family of temporal filters with the following requirements.

- Any GOF length N can be used.
- Each GOF can be processed independently, without need for frames from neighboring GOFs.
- Motion-compensated temporal filtering is used.
- Compression efficiency of the proposed filter set should be at least competitive with the 5/3 filter.

Thus, we propose a new filter set that it is defined by the filter length N and the number of lifting steps involved (S). The filter length N here refers to the number frames being filtered. We refer to the filter set as (N, S) temporal filter. The number of frames N can vary from two to any number and need not be in some power of two. Unlike 5/3 and other longer filters, the proposed (N, S) filter can be processed independently (without reference to other GOFs), and, thus, finite-fixed-size GOFs can be created without introducing high distortions at the boundary. Any combination of (N, S) filters can be chosen to achieve the given delay requirements. The filter design and the delay analysis are explained in detail in the following sections.

A. Design of (N, S) Temporal Filter Set

For any N frames, the proposed filter set decides on the number of lifting steps required, such that after S steps, two low-pass frames at the boundaries and one high-pass frame are created. Hence, the N frames are filtered into a total of two low-pass filtered frames and $N-2$ high-pass temporal frames. The number of lifting steps S to be performed is fixed for any finite number of frames N considered. For the proposed algorithm, as described in the presented pseudocode, it can be verified that the number of lifting steps S required to process N frames and result in two low-pass and one high-pass frame is given by the following equation:

$$S = \begin{cases} 1 & if \quad 2 \le N \le 3 \\ 2 & if \quad 4 \le N \le 5 \\ 3 & if \quad 6 \le N \le 9 \\ 4 & if \quad 10 \le N \le 17. \end{cases} \tag{6}$$

We first describe the proposed filter (N, S) design without including any update step. Thus, the low-pass temporal frames are unfiltered original video frames. At the first step, the low-pass temporal sub-bands are placed at the beginning and at the end of the group of frames considered. At any step for $N \ge 3$, bi-directional motion estimation is used to evaluate the high-pass temporal sub-bands. For the case $N = 2$, forward motion estimation is used to get the high-pass temporal sub-band. Let us consider an example of (5,2) filter set and Fig. 1 shows the lifting steps for (5,2) filter set without any update on the low-pass filtered frames. In Fig. 1, F_k represents the frame $f_k(x, y)$, L_k^s and H_k^s represent the temporal low-pass sub-band $l_k^s(x, y)$ and the temporal high-pass sub-band $h_k^s(x, y)$ respectively. The superscript denotes the lifting step s, where $1 \le s \le S$ and the subscript indicates the temporal sub-band index. Thus, after two steps for $N = 5$, two low-pass and one high-pass temporal frames are created. As we can see from the figure, the low-pass frames are created at the GOF boundaries.

The lifting update step is included to further increase the compression efficiency. The update step is designed such that the low-pass frames created inside the (N, S) filter set will not depend on high-pass frames outside the given N frames. Thus, the first low-pass temporal frame after S steps is never updated and the second low-pass temporal frame is updated using the high-pass frame created at the Sth step. Fig. 2, shows the (5,2)

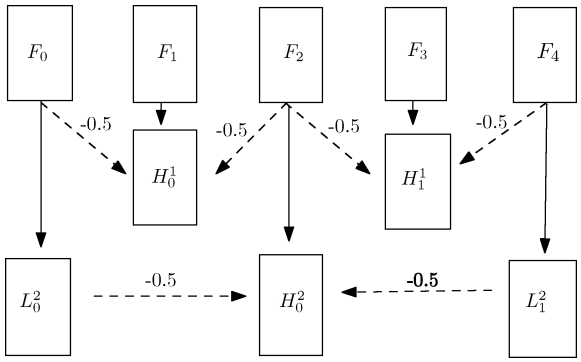

Fig. 1. Proposed filter (5,2) without update step. (Dashed lines indicate which frames are used for motion compensation. F_k represents the frame $f_k(x, y)$, L_k^s and H_k^s represent the temporal low-pass sub-band $l_k^s(x, y)$ and the temporal high-pass sub-band $h_k^s(x, y)$, respectively. Superscripts denote the lifting steps and subscripts indicate the temporal sub-band index).

filter steps with update step. At $s = 1$ and $s = 2$, the low-pass frame L_0^s is never updated. The last low-pass filtered frame inside the GOF at any lifting step is updated using the one high-pass filtered frame which lies inside the given N frames. Hence, at any instance the update step will never use frames outside the input N frames. Thus, the delay can never be more than N frames. The weights w_1, w_2, and w_3 are used to scale the high-pass temporal sub-bands before updating and the values are set according to the motion modeling. These weights can be adaptively selected based on the energy content in the high-pass frames [19] to reduce the ghosting artifacts. The procedure to obtain filtered temporal sub-bands for the proposed temporal filter set can be summarized in the form of pseudocode and is given in Algorithm 1. The proposed (N, S) filter set does not need any special boundary treatment as it uses the original frame information at the boundaries. The (N, S) filter set is very flexible since a filter can be chosen to exactly match the delay requirements. It should be noted that the proposed temporal filter set is no longer a temporal wavelet filter.

Algorithm 1 (N,S) Filter Set

Number of frames $= N$ and Lifting steps $= S$

for all $s = 1 \leq s \leq S$ **do**

$n = \text{ceiling}(N/2)$

Steps to obtain High-Pass Temporal Frames:

if $N > 2$ **then**

for all $k = 1$ **to** $(n - 2)$ **do** [see (A), shown at the bottom of the page]

end for

else $\{N == 2\}$

$$h_0(x,y) = f_1(x,y) - [W_{0\to1}(f_0(x,y))]$$

end if

Steps to obtain Low-Pass Temporal Frames:

$$l_0(x,y) = f_0(x,y)$$

if $N > 2$ **then**

for all $k = 1$ **to** $(n - 2)$ **do** [see (B), shown at the bottom of the page]

end for

for $k = (n - 1)$ **do**

$$l_k(x,y) = f_{2k}(x,y) + w_3[W_{2k-1\to2k}(h_{k-1}(x,y))]$$

end for

if $\text{modulo}(N, 2) == 0$ **then**

$$l_n(x,y) = f_{N-1}(x,y)$$

end if

end if

Reset $N \Leftarrow$ **Number of Low – Pass Temporal Frames**

Reset $f \Leftarrow$ **Low – Pass temporal frames**

end for

The 5/3 filter with a three-level temporal decomposition produces one low-pass temporal sub-band and seven bi-directionally predicted high-pass temporal sub-bands for eight input frames. However, for a (8,3) filter set, we get two low-pass temporal sub-bands and six bi-directionally predicted high-pass temporal sub-bands. Now, for every eight frames, two low-pass filtered frames have to be coded instead of one as in 5/3 filter. This will decrease the compression efficiency of the proposed filter. However, if a (3,1) filter is added to the output of the

$$h_k(x,y) = f_{2k+1}(x,y) - \tfrac{1}{2}[W_{2k\to2k+1}(f_{2k}(x,y)) + W_{2k+2\to2k+1}(f_{2k+2}(x,y))] \quad \text{(A)}$$

$$l_k(x,y) = f_{2k}(x,y) + w_1[(W_{2k-1\to2k}(h_{k-1}(x,y)))] + w_2[(W_{2k+1\to2k}(h_k)(x,y))] \quad \text{(B)}$$

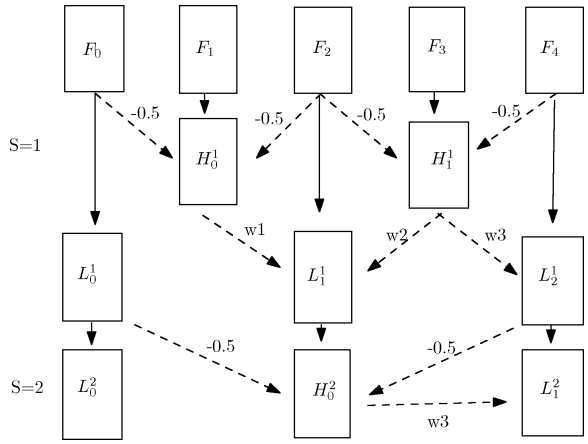

Fig. 2. Proposed filter (5,2) with update step. (Dashed lines indicate which frames are used for motion compensation).

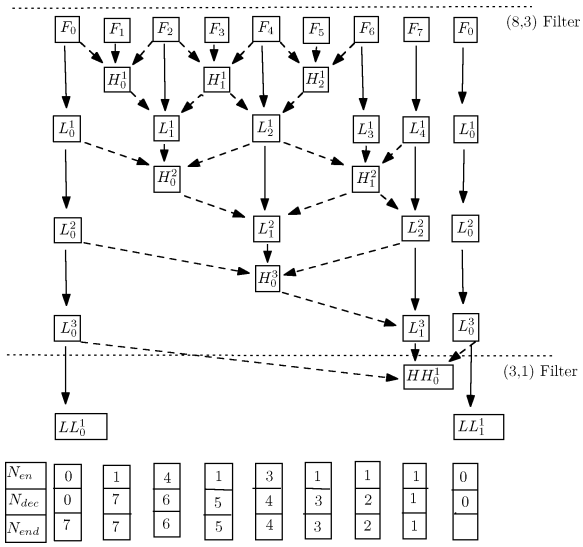

Fig. 3. Total delay for using (8,3) filter followed by (3,1).

(8,3) filter, then we get one low-pass temporal sub-band plus seven bi-directionally predicted high-pass temporal sub-band for every eight frames. When two filters are stacked, the delay will not increase beyond $N+1$ frames. This is because the first low-pass temporal frame is never updated, and, hence, there is no dependency on the future frames. Thus, any two filter sets can be stacked to achieve the desired compression efficiency and delay requirements.

B. Delay Analysis for the (N, S) Filter Set

The delay for the (N, S) filter can be calculated similar to the filter cases explained in Section II-A. Then the three delays discussed in Section II-A are given as

$$N_{\text{en}} = \text{ceiling}\left(\frac{N}{2}\right)$$
$$N_{\text{dec}} = N-2$$
$$N_{\text{end}} = N-2.$$

If we stack two (N,S) filters, the total delay is calculated by adding the two filter delays. Fig. 3 explains the delay calculation for a (8,3) + (3,1) filter set. For encoding frame F_2 (refer to Fig. 3), frame F_4 has to be coded first, which in turn depends on frame F_6. Hence, the maximum encoding delay is four for frame F_2, as it needs four future frames. While decoding, frame F_1 has the maximum decoding delay of seven frames. Again referring to Fig. 3, in order to get back frame F_1, frame F_2 has to be decoded first. The temporal sub-band position at frame F_7 and the first frame from the next GOF is required to reconstruct F_2. Hence, a total of seven frames has to be decoded to reconstruct frame F_1.

For a specific delay case, any (N, S) or two sets of (N, S) filter can be stacked to achieve the exact delay requirement without sacrificing any compression performance. If the delay requirement N_{end} is considered to be seven frames, we have two options that give the same delay: (9,3) filter set or (8,3) + (3,1) filter set. The question is which filter set should be selected for the specified delay case. The number of bi-directional predictions involved in both cases is the same. The (9,3) filter will have two low-pass sub-bands for every nine frames, while the (8,3) + (3,1) filter set has one low-pass temporal sub-band for every eight frames. Thus, the (9,3) will increase the coding cost compared to (8,3) + (3,1). Hence, for a given delay requirement, we choose the filter that will give maximum bi-directional predictions and smallest number of low-pass frames. Although, in some cases, the proposed filters require more motion vectors than the 5/3 filter, in the end, as we show in the experimental results, the proposed filters outperform the 5/3 filter, even when motion vector coding is taken into account.

IV. Rate Allocation

The rate control problem for a video coder can be roughly stated as the determination of proper coding parameters so that decoded video quality is optimized with respect to a certain fixed rate. For an embedded coder, the coding bitrate of the each sub-band can be directly controlled to achieve the required distortion. The rate control problem for an embedded video coder with a GOF of N frames can be stated as: minimize the total GOF distortion given a fixed rate budget

$$\begin{aligned} \text{minimize} \quad & D_1 + D_2 + \cdots + D_N \\ \text{subject to} \quad & R_1 + R_2 + \cdots + R_N = R \end{aligned} \tag{7}$$

where D_i and R_i are the corresponding distortion and rate for the ith frame and R is the total rate budget.

In 3-D wavelet-based video coders, the frame distortion is a linear combination of the temporal sub-band distortions, and, hence, the total distortion is also a linear combination of the distortions of all the temporal sub-bands [19], [26]. Hence, the rate control problem can be modified by selecting the appropriate rates for temporal sub-bands in order to minimize the total GOF distortion. In this section, the distortion relationship between the temporal sub-bands and the reconstructed frame are derived for the (N,S) filter set and the optimal rate allocation procedure is explained.

A. Distortion Model for (N, S) Filter Set

Let us consider an example of the (5,2) filter set as explained in Section III. Motion compensation is ignored for simplicity. The synthesis equations are given as follows:

$$\begin{aligned}
f_0(x,y) &= l_0^2(x,y)\\
f_1(x,y) &= \frac{3}{4}l_0^2(x,y) + \frac{1}{4}l_1^2(x,y) + \frac{2-w_3}{4}h_0^2(x,y)\\
&\quad + \frac{2-w_1}{2}h_0^1(x,y) - \frac{w_2}{2}h_1^1(x,y)\\
f_2(x,y) &= \frac{1}{2}l_0^2(x,y) + \frac{1}{2}l_1^2(x,y) + \frac{2-w_3}{2}h_0^2(x,y)\\
&\quad - w_1h_0^1(x,y) - w_2h_1^1(x,y)\\
f_3(x,y) &= \frac{1}{4}l_0^2(x,y) + \frac{3}{4}l_1^2(x,y) + \frac{2-3w_3}{4}h_0^2(x,y)\\
&\quad - \frac{w_1}{2}h_0^1(x,y) + \frac{2-w_3-w_2}{2}h_1^1(x,y)\\
f_4(x,y) &= l_1^2(x,y) - w_3h_0^2(x,y) - w_3h_1^1(x,y). \qquad (8)
\end{aligned}$$

The synthesis equations can also be represented in matrix form. Let the original frames of size $X \times Y$ be represented by F_k and the low-pass and the high-pass temporal sub-bands by L_k^s and H_k^s respectively. Then we can define the frame vector to be $\mathbf{f} = [F_1, F_2, F_3, F_4, F_5]^T$ and the temporal sub-band vector to be $\mathbf{t} = [L_0^2, L_1^2, H_0^2, H_0^1, H_1^1]^T$. Then

$$\mathbf{f} = \mathbf{M} \cdot \mathbf{t} \qquad (9)$$

where $\mathbf{M}$ is a 5×5 matrix for the (5,2) filter set and from (8), $\mathbf{M}$ can be formed follows:

$$\mathbf{M} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0\\
\frac{3}{4} & \frac{1}{4} & \frac{2-w_3}{4} & \frac{2-w_1}{2} & -\frac{w_2}{2}\\
\frac{1}{4} & \frac{3}{4} & \frac{2-w_3}{2} & -w_1 & -w_2\\
\frac{1}{4} & \frac{3}{4} & \frac{2-3w_3}{2} & -\frac{w_1}{2} & \frac{2-w_3-w_2}{2}\\
1 & 0 & -w_3 & 0 & -w_3
\end{bmatrix}.$$

Similarly, for any (N, S) filter set, the matrix $\mathbf{M}$ of size $N \times N$ can be formed. Let $D_{L_0^2}$, $D_{L_1^2}$, $D_{H_0^2}$, $D_{H_0^1}$, and $D_{H_1^1}$ be the corresponding temporal sub-band distortions and D_{F_n}, where $n = 0, \cdots, N-1$, be the reconstructed frame distortion. Since all the temporal sub-bands are quantized and coded separately after performing temporal filtering, it is reasonable to assume that all the temporal sub-band distortions are uncorrelated. The total distortion D of the reconstructed frames can be then calculated as

$$\begin{aligned}
D &= \sum_{n=1}^{N-1} D_{F_n}\\
&= m_1D_{L_0^2} + m_2D_{L_1^2} + m_3D_{H_0^2} + m_4D_{H_0^1} + m_5D_{H_1^1} \qquad (10)
\end{aligned}$$

where m_n is the squared norm of the nth column of matrix M. The rate allocation problem from (7) is now modified as

$$\begin{aligned}
\min \quad & D = \sum_{n=1}^{N-1} D_{F_n}\\
\text{subject to} \quad & R_{L_0^2} + R_{L_1^2} + R_{H_0^2} + R_{H_0^1} + R_{H_1^1} \le R. \qquad (11)
\end{aligned}$$

Using the Lagrangian method, the cost function to minimize becomes

$$\begin{aligned}
J = &\left(m_1D_{L_0^2} + m_2D_{L_1^2} + m_3D_{H_0^2} + m_4D_{H_0^1} + m_5D_{H_1^1}\right)\\
&+\lambda\left(R_{L_0^2} + R_{L_1^2} + R_{H_0^2} + R_{H_0^1} + R_{H_1^1} - R\right). \qquad (12)
\end{aligned}$$

The temporal sub-band distortion has to be modeled to get the solution for the optimization problem. We choose the exponential rate-distortion model [27], [28] for the temporal sub-band distortion, which is valid for relatively high rates. Then, the temporal sub-band distortion is given by

$$D_{si} = \epsilon_{si}\sigma_{si}^2 2^{-\beta_{si}R_{si}} \qquad (13)$$

where ϵ_{si}, σ_{si}^2 and β_{si} for each temporal sub-band si has to be determined. In this work, these parameters are determined with a linear mean-squared-error (LMSE) curve fitting of experimental data [28].

The solution for (12) can then be given as

$$\begin{aligned}
\frac{D_{L_0^2}}{D_{L_1^2}} &= \frac{\beta_2 m_2}{\beta_1 m_1};\ \frac{D_{L_1^2}}{D_{H_0^2}} = \frac{\beta_3 m_3}{\beta_2 m_2}\\
\frac{D_{H_0^2}}{D_{H_0^1}} &= \frac{\beta_4 m_4}{\beta_3 m_3};\ \frac{D_{H_0^1}}{D_{H_1^1}} = \frac{\beta_5 m_5}{\beta_4 m_4}. \qquad (14)
\end{aligned}$$

Thus, a similar procedure is followed for any given filter set and the optimal rates can be assigned to the temporal sub-bands to maximize the output performance of the video codec.

B. Rate Control Algorithm

In this paper, a simple search algorithm is used to decide the rates to meet the optimal distortion criteria given in (14). The algorithm for choosing the rate to minimize total distortion is given as follows.

1) Decide on the total rate R assigned to the GOF of size N (this is an input to the algorithm).
2) For each wavelet temporal sub-band in the GOF estimate m_n and β_n, using LMSE curve fitting, and q R-D points, using (13). Calculating more points will result in meeting the target bitrate more accurately, but will also result in increased computational complexity.
3) Initially, let $R_{L_0^2} = c.R/N$, where c is a multiplication constant. The corresponding distortion $D_{L_0^2}$ is calculated from (13).
4) Using the distortion ratios for temporal sub-bands (14), select $D_{L_1^2}$, $D_{H_0^2}$, $D_{H_0^1}$, and $D_{H_1^1}$ from the q points and get the corresponding rates $D_{L_1^2}$, $D_{H_0^2}$, $D_{H_0^1}$, and $D_{H_1^1}$.
5) Check if the sum of the rates of temporal sub-bands is equal to R, if equal goto next GOF.
6) If the sum is greater than R, decrease the value for c. Else increase c and goto Step 3.

Any numerical analysis method can also be used to calculate the optimal temporal sub-band rates. The accuracy of the assumed exponential model for temporal sub-band is very important to get optimal rates.

TABLE II
AVERAGE PSNR VALUES OF Y COMPONENT FOR "FOOTBALL" SEQUENCE

Rate in kbps	5/3 Filter N_{end}=21	(9,3)+(3,1) Filter set N_{end}=8	(8,3)+(3,1) Filter set N_{end}=7	(5,2)+(3,1) Filter set N_{end}= 4
1268	29.41 *dB*	29.67 *dB*	29.59 *dB*	29.32 *dB*
1024	28.47 *dB*	28.64 *dB*	28.51 *dB*	28.33 *dB*
768	27.23 *dB*	27.50 *dB*	27.37 *dB*	27.11 *dB*

TABLE III
AVERAGE PSNR VALUES OF Y COMPONENT FOR "FLOWER GARDEN" SEQUENCE

Rate in kbps	5/3 Filter N_{end}=21	(9,3)+(3,1) Filter set N_{end}=8	(8,3)+(3,1) Filter set N_{end}=7	(5,2)+(3,1) Filter set N_{end}= 4
1268	27.19 *dB*	27.881 *dB*	27.701 *dB*	27.051 *dB*
1024	26.89 *dB*	27.414 *dB*	27.264 *dB*	26.774 *dB*
768	25.58 *dB*	26.04 *dB*	26.01 *dB*	25.54 *dB*

TABLE IV
AVERAGE PSNR VALUES OF Y COMPONENT FOR "SUSIE" SEQUENCE

Rate in kbps	5/3 Filter N_{end}=21	(9,3)+(3,1) Filter set N_{end}=8	(8,3)+(3,1) Filter set N_{end}=7	(5,2)+(3,1) Filter set N_{end}= 4
380	41.79*dB*	42.27 *dB*	41.94 *dB*	41.64 *dB*
300	40.49 *dB*	41.02 *dB*	40.62 *dB*	40.37 *dB*
228	38.88*dB*	39.4 *dB*	39.14 *dB*	38.83 *dB*

TABLE V
AVERAGE PSNR VALUES OF Y COMPONENT FOR "FOREMAN" SEQUENCE

Rate in Kbps	5/3 Filter N_{end}=21	(9,3)+(3,1) Filter set N_{end}=8	(8,3)+(3,1) Filter set N_{end}=7	(5,2)+(3,1) Filter set N_{end}= 4
380	37.87 *dB*	38.58 *dB*	38.46 *dB*	37.76 *dB*
300	36.82 *dB*	37.61 *dB*	37.34 *dB*	36.74 *dB*
228	35.42 *dB*	36.18 *dB*	36.02 *dB*	35.38 *dB*

V. EXPERIMENTAL RESULTS

A. Coder Setup

A wavelet-based video coder is implemented using the low band-shift method as explained in [1]. Hence, our proposed 3-D-coder belongs to the 2-D+t category. An input frame is decomposed in the critically sampled DWT domain and the reference frame is transformed using ODWT. A Daubechies (9,7) filter with a three level spatial decomposition is used to compute the wavelet coefficients. 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. 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). A 16×16 wavelet block is matched in a search window of $[-16, 16]$. All results reported use integer pixel accuracy for ME/MC. The weights for the update case are chosen to be $w_1 = w_2 = 0.25$ and $w_3 = 0.5$. We have used standard test sequences, two in SIF (352×240) resolution, "Football" and "Flower Garden," and two in QCIF (176×144) resolution, "Foreman" and "Susie." The temporal sub-bands are compressed using the SPIHT coder [29]. Both the 5/3 filter and the proposed filter set use the method described in Section IV to minimize the total distortion. For the 5/3 filter the matrix M reduces to the synthesis gain factors [19], [27]. The model parameters are calculated for each temporal sub-band as explained in Section IV-A and the algorithm described in Section IV-B is used for the rate selection. Since it is very difficult to exactly achieve the distortions to follow the derived ratios from q points, a room for 2% error in distortion was allowed.

B. Results

We gauge the performance of the proposed temporal filters under various delay requirements. Tables II–V give the average PSNR values of the Y component for the different sequences at three different rates and three different delay conditions. Figs. 4 and 6 for the "Football" and "Susie" sequences, respectively, give the average PSNR vs Bitrate in Kbps for the three proposed filter sets and the 5/3 filter. From Fig. 4 and Table II, we can infer that the proposed filter sets (9,3) + (3,1)

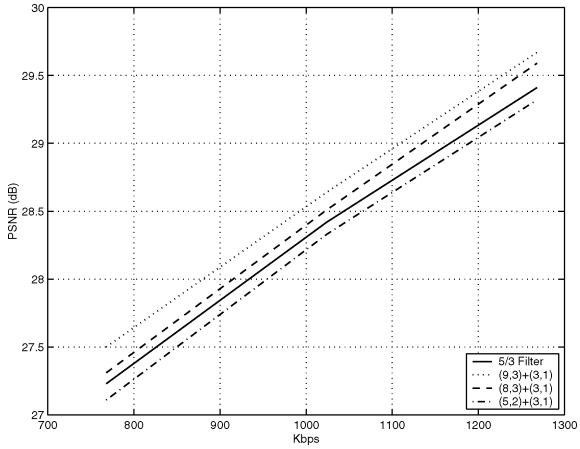

Fig. 4. Comparison of proposed filter sets with 5/3 filter for "Football" sequence.

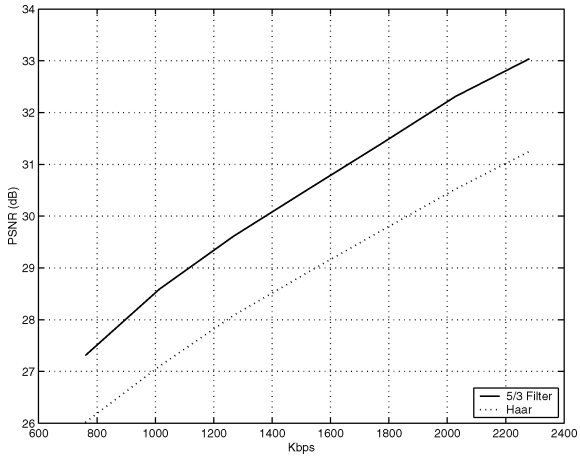

Fig. 5. Comparison of Haar filter with 5/3 filter for "football" sequence.

and (8,3) + (3,1) perform better than the 5/3 filter. The delay is reduced by a factor of 2.6 and 3 for (9,3) + (3,1) and (8,3) + (3,1), respectively, compared to the 5/3 filter. The (5,3) + (3,1) filter achieves an average PSNR slightly less than the 5/3 filter while the delay is around 150 ms compared to 700 ms for a 30 frames/s input video.

In Fig. 5, the Haar filter is compared with the 5/3 filter for the "Football" sequence for a three level temporal decomposition. The average PSNR of the Haar filter is approximately 1.5 dB less than 5/3 filter and the N_{end} for the Haar filter is six frames. Our proposed (5,2) + (3,1) filter offers less delay compared to the Haar filter while exhibiting a compression efficiency that is close to the 5/3 filter.

The (9,3) + (3,1) and (8,3) + (3,1) filter combinations outperform the 5/3 filter while having lower delay requirements. This holds for all the sequences considered. Hence, we have shown that we do not have to lose coding efficiency to reduce the delay requirements. From the PSNR values, we can infer that the compression performance does not get affected when you decrease the delay. The proposed filter set provides good compression while having flexible delay characteristics. Introducing subpixel ME/MC and adaptive update techniques can further increase the overall coding efficiency.

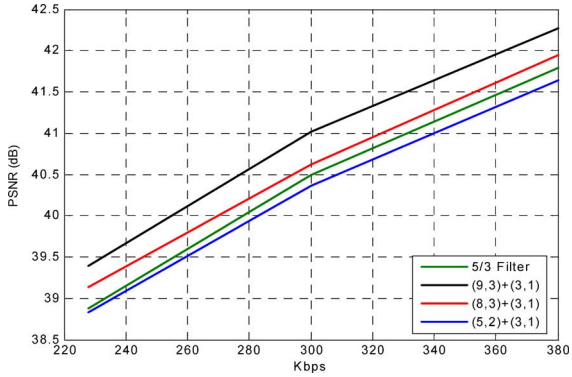

Fig. 6. Comparison of proposed filter sets with 5/3 filter for "Susie" sequence.

VI. Conclusion

In wavelet-based video coders using 3-D sub-band coding methods, drift is eliminated and high-compression efficiency is also achieved. However, the 3-D scheme has to process a group of frames to take wavelet transform and it introduces high-coding delays. We have proposed a novel temporal filter set with motion compensation for 3-D 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. The proposed (N,S) filter set offers less delay and high-compression efficiency compared to the 5/3 filter.

References

[1] H. W. Park and H. S. Kim, "Motion estimation using low-band-shift method for wavelet-based moving-picture coding," *IEEE Trans. Image Process.*, vol. 9, no. 4, pp. 577–587, Apr. 2000.

[2] Y. Andreopoulos, A. Munteanu, G. VanderAuwera, P. Schelkens, and J. Cornelis, "Wavelet-based fully scalable video coding with in-band prediction," presented at the Benelux Signal Processing Symp., Leuven, Belgium, 2002.

[3] X. Li, L. Kerofski, and S. Lei, "All-phase motion compensated prediction in the wavelet domain for high performance video coding," in *Proc. Int. Conf. Image Processing*, Thessaloniki, Greece, 2001, vol. 3, pp. 538–541.

[4] S. Cui, Y. Wang, and J. E. Fowler, "Multihypothesis motion compensation in redundant wavelet domain," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, 2003, vol. 2, pp. 53–56.

[5] V. Seran and L. P. Kondi, "Drift control in variable bitrate wireless channels for scalable wavelet based video coding in the overcomplete discrete wavelet transform domain," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2005, vol. 3, pp. 233–236.

[6] A. R. Reibman, L. Bottou, and A. Basso, "Scalable video coding with managed drift," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 2, pp. 131–140, Feb. 2003.

[7] J.-R. Ohm, "Three dimensional sub-band coding with motion compensation," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 559–571, Sep. 1994.

[8] S. Choi and J. Woods, "Motion-compensated 3-D sub-band coding of video," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 155–167, Feb. 1999.

[9] B. Pesquet-Popescu and V. Bottreau, "Three-dimensional lifting schemes for motion compensated video compression," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Salt Lake City, UT, 2001, pp. 1793–1796.

[10] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3-D wavelet transform based on lifting," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, Greece, 2001, pp. 1029–1032.

[11] Y. Andreopoulos, A. Munteanu, J. Barbarien, M. V. der Schaar, J. Cornelis, and P. Schelkens, "In-band motion compensated temporal filtering," *Signal Process.: Image Commun.*, vol. 19, pp. 653–673, Aug. 2004.

[12] Y. Wang, S. Cui, and J. E. Fowler, "3-D video coding using redundant-wavelet multihypothesis and motion-compensated temporal filtering," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, 2003, vol. 2, pp. 755–758.
[13] X. Li, "Scalable video compression via overcomplete motion compensated wavelet coding," *Signal Process.: Image Commun.*, vol. 19, pp. 637–651, Aug. 2004.
[14] A. Secker and D. Taubman, "Lifting based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1530–1542, Dec. 2003.
[15] S. T. Hsiang and J. W. Woods, "Embedded video coding using motion compensated 3-D subband/wavelet filter bank," presented at the Packet Video Workshop, Sardinia, Italy, May 2000.
[16] A. Golwelkar and J. Woods, "Scalable video compression using longer motion compensated temporal filters," in *Proc. SPIE Conf. Visual Communications and Image Processing*, 2003, vol. 5150, pp. 1406–1416.
[17] B. J. Kim, Z. Xiong, and W. A. Pearlman, "Low bit-rate scalable video coding with 3-D set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 12, pp. 1374–1387, Dec. 2000.
[18] J. Xu, S. Li, and Y. Q. Zhang, "A wavelet codec using 3-D ESCOT," presented at the IEEE PCM, Dec. 2000.
[19] N. Mehrseresht and D. Taubman, "An efficient content adaptive motion compensation 3-D-DWT with enhanced spatial and temporal scalability," in *Proc. IEEE Int. Conf. Image Processing*, 2004, vol. 2, pp. 1329–1332.
[20] T. Andre, M. Cagnazzo, M. Antonini, and M. Barlaud, "JPEG2000-compatible scalable scheme for wavelet-based video coding," *EURASIP J. Image Video Process.*, 2007, DOI: 10.1155/2007/30852.
[21] V. Seran and L. P. Kondi, "3-D based video coding in the overcomplete discrete wavelet transform domain with reduced delay requirements," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2005, vol. 3, pp. 237–240.
[22] V. Seran and L. P. Kondi, "Improved temporal filtering scheme to reduce delay and distortion fluctuation in 3-D wavelet based video coding," presented at the IEEE Western New York Image Processing Workshop, Sep. 2005.
[23] A. Golwelkar, "Motion compensated temporal filtering and motion vector coding using longer filters," Ph.D. dissertation, Rensselaer Polytechnic Inst., Troy, NY, 2004.
[24] M. V. der Schaar and D. Turaga, "Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding," presented at the IEEE Int. Conf. Acoustics, Speech, and Signal Processing, 2003.
[25] G. Pau, J. Vieron, and B. Pesquest-Posescu, "Video coding with flexible MCTF structures for low end-to-end delay," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2005, vol. 3, pp. 241–244.
[26] V. Seran and L. P. Kondi, "Distortion fluctuation control for 3-D wavelet video coding," presented at the IEEE Int. Conf. Visual Communications and Image Processing, Jan. 2006.
[27] D. S. Taubman and M. W. Marcellin, *JPEG2000, Image Compression Fundamentals, Standards and Practice*. Norwell, MA: Kluwer, 2002.
[28] P. Cheng, J. Li, and C.-C. Kuo, "Rate control for an embedded wavelet video coder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 4, pp. 696–702, Aug. 1997.
[29] A. Said and W. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, Jun. 1996.

Vidhya Seran, photograph and biography not available at the time of publication.

Lisimachos P. Kondi (S'92–M'99) received the Diploma degree in electrical engineering from the Aristotle University of Thessaloniki, Greece, in 1994, and the M.S. and Ph.D. degrees in electrical and computer engineering from Northwestern University, Evanston, IL, in 1996 and 1999, respectively.

During the 1999–2000 academic year, he was a Postdoctoral Research Associate at Northwestern University. Since August 2000, he has been with the faculty of the Department of Electrical Engineering, The State University of New York at Buffalo. He was a visiting summer faculty at the Naval Research Laboratory, Washington, DC, in 2001, and at the Air Force Research Laboratory, Rome, NY, 2005 and 2006. His research interests are in the general areas of signal and image processing and communications, including image and video compression and transmission over wireless channels and the Internet, scalable and multiple description coding, CDMA wireless communications, super-resolution of video sequences, and shape coding.

Since July 2005, Dr. Kondi has been an Associate Editor of the *EURASIP Journal of Applied Signal Processing*. He is also a Guest Editor of a special issue on Video Communications for 4G Wireless Systems of the Wiley *Journal on Wireless Communications and Mobile Computing* (2007).