Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Signal Processing: *Image Communication* 19 (2004) 877–908
SIGNAL PROCESSING: IMAGE COMMUNICATION
www.elsevier.com/locate/image

# Interframe wavelet coding—motion picture representation for universal scalability

Jens-Rainer Ohm[a,*], Mihaela van der Schaar[b], John W. Woods[c,1]

[a] *Chair and Institute of Communications Engineering, RWTH Aachen University, Melatener Str. 23, D-52074 Aachen, Germany*
[b] *Department of Electrical and Computer Engineering, University of California Davis, One Shields Avenue, 3129 Kemper Hall, Davis, CA 95616-5294, USA*
[c] *Center for Next Generation Video (CNGV), Rensselaer Polytechnic Institute, Troy, NY, 12180-3590, USA*

**Abstract**

Scalability at the bitstream level is an important feature for encoded video that is to be transmitted and stored with a variety of target rates or to be replayed on devices with different capabilities and resolutions. This is attractive for digital cinema applications, where the same encoded source representation could seamlessly be used for purposes of archival and various distribution channels. Conventional high-performance video compression schemes are based on the method of motion-compensated prediction, using a recursive loop in the prediction process. Due to this recursion and the inherent drift in cases of deviation between encoder and decoder states, scalability is difficult to realize and typically effects a penalty in compression performance for prediction-based coders. The method of interframe wavelet coding overcomes this limitation by replacing the prediction along the time axis by a wavelet filter, which can nevertheless be operated in combination with motion compensation. Recent advances in motion-compensated temporal filtering (MCTF) have proven that combination with arbitrary motion compensation methods is possible. Compression performance is achieved that is comparable with state of the art single-layer coders targeting only for one rate. The paper provides an explanation of MCTF methods and the resulting 3D wavelet representation, and shows results obtained in the context of encoding digital cinema (DC) materials.

*Keywords:* Video coding; Interframe wavelet; Motion-compensated filtering; Scalable video compression; Digital cinema

## 1. Introduction

In the future, motion pictures will mostly be transmitted over variable bandwidth channels, both in wireless and cable networks. They have to be stored on media of different capacity, such as memory cards and high-capacity DVD; they have to be replayed on a variety of devices, ranging from small mobile terminals to high-resolution projection systems. Scalable video coding schemes are intended to encode the signal once at highest resolution, but enable decoding from partial streams depending on the specific rate and resolution required by a certain application. This enables

[*] Corresponding author. Tel.: +49-241-80-27671; fax: 49-241-80-22196.

*E-mail address:* ohm@ient.rwth-aachen.de (J.-R. Ohm).

[1] Work was partially supported by the US National Science Foundation under grant EEC-9812706.

0923-5965/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2004.06.004

a simple and flexible solution for transmission over heterogeneous networks, additionally providing adaptability for bandwidth variations and error control. It enables both multicast and unicast streaming applications with minimal processing at the server or in the network, and low-complexity decoding. It further allows simple adaptation for a variety of storage devices and terminals.

For video coding, a lack of efficiency can however be observed in combining scalable coding with the popular approach of hybrid motion-compensated prediction and block transform encoding, as implemented in most of today's standards. This is mainly caused by the recursive structure of the prediction loop. Research for more efficient scalable coding techniques is still a demanding area in video compression. Recent breakthroughs in motion-compensated temporal wavelet filtering have finally given a realistic perspective to implement highly efficient scalable video codecs. These new wavelet codecs provide numerous advantages over non-scalable conventional techniques based on motion-compensated prediction:

- No recursive predictive loop as in the current standards (MPEG-x, H.26x), such that no drift occurs if decoding is performed at various bit-rates and resolutions.
- Separation of noise and sampling artefacts by usage of longer temporal filters.
- Flexible exploitation of long range as well as short range temporal redundancies.
- Adaptability in the spatial and temporal filtering methods, number of decomposition levels, and filter choices, which makes improvements possible that are not feasible in predictive coding.

As a consequence, these wavelet video coding schemes can provide flexible spatial, temporal, SNR and complexity scalability with fine granularity over a wide range of bit rates, while maintaining a very high coding efficiency. They can also be regarded as a superset of well-established still image wavelet coding techniques like JPEG2000. The inherent prioritization of data in this framework, as well as the availability of mature spatio–temporal wavelet filtering techniques combinable with any kind of motion compensation, leads to added robustness and considerably improved error resilience properties.

This paper will highlight the principles of wavelet based video coding schemes. It presents a general review of interframe wavelet video coding methods, including classification and detailed presentation of some of the motion-compensated wavelet coding schemes proposed so far. These techniques also establish the basis of an exploration activity which has been performed under the auspices of MPEG.

The organization of the paper is as follows. Section 2 introduces the framework of motion-compensated temporal filtering (MCTF), which establishes the basis of a fully three-dimensional (3D) (spatio–temporal) wavelet transform with motion compensation. Section 3 extends these methods for processing of spatial and temporal transforms in an arbitrary sequential order. Section 4 reviews and summarizes recent advances in the field. Section 5 introduces MC-EZBC (embedded zero-block coding) and its application in scalable compression of digital cinema materials, including experimental results. Section 6 concludes.

## 2. Motion-compensated filtering for interframe wavelet coding

Fig. 1 illustrates a 3D (spatio–temporal) Wavelet transform tree, where in the simplest case a Haar basis can be used for wavelet decomposition along the temporal axis. Schemes of this type without motion compensation have been proposed more than 15 years ago, see e.g. [18]. In case of a non-orthonormal transform, this can be interpreted as decomposition of a frame pair $(A, B)$ into one average (lowpass) and one difference (highpass) frame

$$L(m,n) = \tfrac{1}{2}[A(m,n) + B(m,n)],$$
$$H(m,n) = A(m,n) - B(m,n). \qquad (1)$$

If pairs of lowpass frames are then again combined, subsequent levels of a wavelet tree are established. At the end nodes of the temporal decomposition, a 2D spatial wavelet transform is applied. With a number of $T$ wavelet tree levels temporally, the resulting temporal block length in 3D wavelet transform is $W = 2^T$.

3D wavelet schemes allow utilization of contexts which implicitly relate to the shear of the 3D spectrum which is effected by motion [31]. To illustrate this, the 3D wavelet decomposition scheme of Fig. 1 is re-interpreted as a *wavelet transform cube* in Fig. 2. 3D zero-tree methods have been proposed, where typically correspondences between spatial bands are unchanged as compared to conventional 2D zero-tree structures (see front side of the cube). Additional correspondences exist however over the bands of the temporal-axis wavelet decomposition, which implicitly may reflect the fact that the 'temporal' frequency linearly increases with 'spatial' frequency if translational motion occurs. This means that, once a directional correspondence relating to a shear angle of the spectrum is found, it is most probable that the same direction can further be tracked towards higher frequencies, with a high probability of finding zero-tree correspondences. Based on these principles, a 3D version of SPIHT has been introduced in [20]. In [50], another efficient three-dimensional wavelet video coding algorithm called 3D embedded subband coding with optimized truncation (3D ESCOT) was proposed. In this algorithm, coefficients in each subband are coded independently using a fractional bit-plane coding approach. This feature makes it very easy to achieve frame-rate scalability and resolution scalability for the coded video stream. Moreover, a context-based adaptive arithmetic coder with elaborated context assignment as well as global rate-distortion optimization is used in the ESCOT algorithm to achieve high compression efficiency. Compared to the 3D SPIHT algorithm, 3D ESCOT preserves the scalability of the compressed bit-stream and shows even higher compression performance.

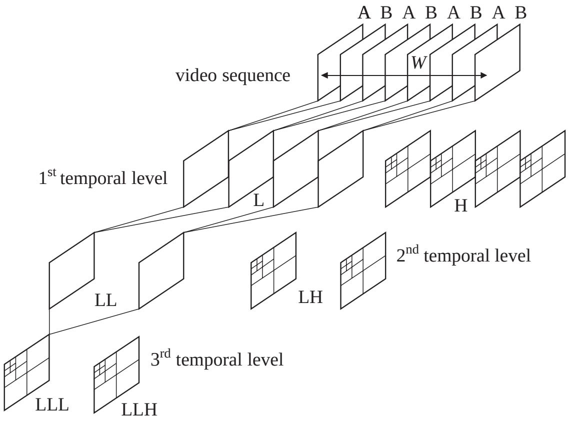


Fig. 1. Spatio–temporal wavelet decomposition using $T = 3$ levels of a temporal wavelet tree.

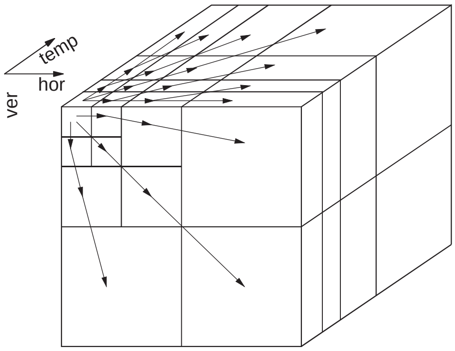


Fig. 2. 3D wavelet transform cube with possible zero-tree correspondences.

So far, spatio–temporal frequency coding methods without motion compensation were introduced. Application of motion compensation (MC) is often regarded to be implicitly coupled with frame prediction schemes. There is indeed no justification for this restriction, as MC can rather be interpreted as a method to align a filtering operation with a motion trajectory along the temporal axis [21]. In the case of MC prediction, the filters are in principle LPC analysis and synthesis filters, while in cases of transform or wavelet coding, transform basis functions are subject to MC alignment. This is denoted as *motion-compensated temporal filtering* (MCTF). If MCTF is used in combination with a 2D spatial wavelet transform, this is denoted as a 3D, or (depending on the sequence of the spatial and temporal processing) either as a 2D + $t$ or $t$ + 2D wavelet transform.

Since transform and subband/wavelet methods are fully described by linear filter operations, they can probably likewise be applied along a motion trajectory. If however motion vectors are *spatially varying*, isolated areas may be present, which are not member of any uniquely connected motion trajectory. Upon unique trajectories (Fig. 3a), all pixels can ideally be reconstructed by the respective synthesis filtering, which must include inverse MC mapping. In case of inhomogeneous motion vector fields (Fig. 3b), as they e.g. occur when objects move differently, motion trajectories can diverge, such that certain pixels or entire areas may not be members of any motion trajectory; these positions are related to newly uncovered areas, and are denoted as *unconnected*. The same may occur at frame boundaries. Another case occurs when motion trajectories converge or merge, which e.g. happens when areas are being covered. Here, certain coordinate references are *multiple connected*. In the latter case, information would be duplicated in the transform coefficients, while in the former case, information would be missing and reconstruction would be impossible.

### 2.1. Temporal-axis Haar filters with MC

A solution to the problem of unreferenced pixels in case of Haar filters can be made as follows by re-defining the coordinate references with regard to the motion shifts, first proposed in [28]. Regard a motion-compensated non-orthonormal Haar filter pair with $z$ transform

$$H_0(z) = \tfrac{1}{2}(1 + z_1^{\tilde{k}} \cdot z_2^{\tilde{l}} \cdot z_3^{-1}),$$

$$H_1(z) = -z_1^{k} \cdot z_2^{l} + z_3^{-1}. \tag{2}$$

The effect of this modification shall again be interpreted by transforming a pair of even/odd indexed frames $A$ and $B$ into one 'lowpass' frame $L$ and one 'highpass' frame $H$, such that

$$\begin{aligned} L(m,n) &= 0.5 \cdot B(m,n) \\ &\qquad + 0.5 \cdot A(m + \tilde{k}(m,n), n + \tilde{l}(m,n)) \\ H(m,n) &= A(m,n) - B(m + k(m,n), n + l(m,n)). \end{aligned} \tag{3}$$

Obviously here, for the case of temporal-axis Haar filters, the $L$ frame is the motion-compensated *average*, and the $H$ frame is the motion-compensated *difference* between the two frames. The motion vector $[k,l]^{\mathrm{T}}$ shall characterize the forward motion originating from frame $A$ towards frame $B$, while $[\tilde{k},\tilde{l}]^{\mathrm{T}}$ describes the backward motion from $B$ towards $A$.[2] If a unique motion trajectory exists, both motion vectors cannot be independent of each other, as they shall connect corresponding pixels. If e.g. estimation of $[k,l]^{\mathrm{T}}$ is performed at all positions ($m_A$,$n_A$) in frame $A$, parameters $[\tilde{k},\tilde{l}]^{\mathrm{T}}$ can uniquely be defined, whenever corresponding positions ($m_B$,$n_B$) are neither unconnected nor multiple connected:

$$\begin{aligned} \tilde{k}(m_B,n_B) &= -k(m_A,n_A) \\ \tilde{l}(m_B,n_B) &= -l(m_A,n_A) \end{aligned} \quad \text{with} \quad \begin{cases} m_B = m_A + k(m_A,n_A), \\ n_B = n_A + l(m_A,n_A). \end{cases} \tag{4}$$

In case of multiple-connected mappings, it is still possible to determine a value for $[\tilde{k},\tilde{l}]^{\mathrm{T}}$ by setting *selection rules*, e.g. to use the smaller of two or more vectors targeting one pixel. All remaining positions ($m_B$,$n_B$) then belong to unconnected areas in $B$, where parameters $[\tilde{k},\tilde{l}]^{\mathrm{T}}$ cannot be determined from (4) or by selection rules. For these latter positions, original values from $B$ are filled into $L$. For the not-selected multiple connected positions, i.e. those which violate (4) and were also rejected by the selection rules, it is possible to fill a motion-compensated prediction error in $H$, as will be described in detail in the following paragraphs. In total, this procedure does not produce any overhead or spatial discontinuity in the motion-compensated $L$ and $H$ subband frames, except for

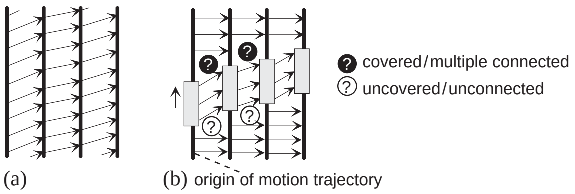


Fig. 3. Forward motion trajectories in case of (a) homogeneous (b) inhomogeneous motion vector fields.

[2] In the sequel, we will generally assume that the coordinate system of $H$ is related to the positions of $A$, while the coordinate system of $L$ relates to positions of $B$. These relationship definitions are arbitrary and can be made vice versa without any restriction.

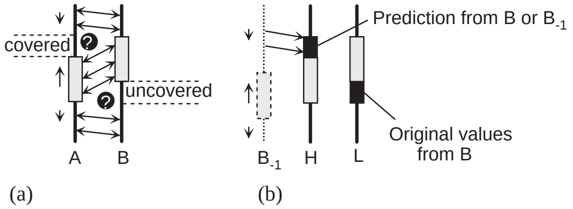


Fig. 4. (a) Covered and uncovered areas in case of frame pairs; (b) Substitution of predictive coded areas into the 'highpass' frame, original frame areas into the 'lowpass' frame.

possible effects which are caused by erroneous motion vector fields.

The information about remaining 'multiple connected' pixels from frame $A$ is integrated as prediction differences into the highpass frame, while the unconnected pixels from frame $B$ are embedded into the lowpass frame (see Fig. 4):

$$L(m,n) = B(m,n) \quad \text{if 'unconnected'},$$
$$H(m,n) = A(m,n) - \hat{A}(m,n) \quad \text{if 'multiple connected'} \tag{5}$$

The prediction reference $\hat{A}(m,n)$ can in principle refer to the (subsequent) frame $B$ or to the preceding frame $B_{-1}$. As vectors $[k,l]^{\mathrm{T}}$ are defined for any position[3] irrespective of multiple connections occurring, it is straightforward to select between the following two modes, where the mode switching information must be conveyed to the decoder,

$$\hat{A}(m,n) = B(m+k(m,n), n+l(m,n)) \quad \text{'backward mode'},$$
$$\hat{A}(m,n) = B_{-1}(m-k(m,n), n-l(m,n)) \quad \text{'forward mode'}. \tag{6}$$

All operations defined in (3)–(6) are then fully invertible. For normally-connected pixels the synthesis equations are:

$$\tilde{A}(m,n) = L(m+k(m,n), n+l(m,n)) + 0.5H(m,n),$$
$$\tilde{B}(m,n) = L(m,n) - 0.5H(m+\tilde{k}(m,n), n+\tilde{l}(m,n)), \tag{7}$$

while for the exceptional cases of unconnected or multiple-connected pixels

$$\tilde{B}(m,n) = L(m,n) \quad \text{if 'unconnected'},$$
$$\tilde{A}(m,n) = \hat{A}(m,n) + H(m,n) \quad \text{if 'multiple connected'}. \tag{8}$$

Perfect reconstruction is strictly possible, when full-pixel accuracy of motion compensation is implemented. Motion compensation using sub-pixel motion shift will lead to lossy reconstruction, as then sub-pixel position interpolations would be necessary in analysis *and* synthesis steps, which could never be perfect unless an ideal interpolator was used. Nevertheless, it was shown in [29] that *arbitrary methods of motion compensation* can be used and that the reconstruction error can be made reasonably small when interpolators of high quality are used to compute the sub-pixel positions.

Fig. 5 shows frames processed by the motion-compensated temporal axis wavelet filtering, employing four levels of temporal-axis transform, which are compared against the result of processing without motion compensation. It is obvious that without motion compensation, the low-frequency frame *LLLL* is becoming heavily blurred, while the high-frequency frame *H* carries a lot of detail information yet. In principle, the highpass frame shows the same behavior as a prediction error frame without motion compensation. In the motion-compensated case, the lowpass frame *LLLL* contains all relevant image information; it appears similar to an original frame, but indeed is an average over 16 frames here; such a frame can well be used as a member of a temporally sub-sampled sequence which can be displayed at lower frame rate. It is obvious that spatio–temporal wavelet coding *withou*t MC can hardly be used for the purpose of temporal scalability.

### 2.2. Temporal-axis lifting filters for arbitrary MC

Any pair of biorthogonal filters can be implemented in a *lifting structure* as shown in Fig. 6 [37]. The first step of the lifting filter is a decomposition of the signal into its even- and odd-indexed

[3] This must not necessarily mean that individual vectors are defined differently for any position; in fact, block-based definition of motion vector fields is often used in MCTF systems.

polyphase components. Then, the two basic operations are *prediction steps* $P(z)$ and *update steps* $U(z)$. The prediction and update filters are primitive kernels of typically 2 or 3 taps each; the number of steps necessary and the values of coefficients in each step are determined by a factorization of biorthogonal filter pairs. Finally, normalization by factors $K_L$ and $K_H$ is applied to obtain an orthonormal decomposition.

The lifting scheme can now be used to give a different interpretation of the motion-compensated transform between a pair of frames $A$ and $B$ of a video sequence, which shall be transformed into one lowpass frame $L$ and one highpass frame $H$. Herein, the frames $A$ and $B$ are interpreted as the even and odd polyphase components of the temporal-axis transform. Assume that $A^*$ and $B^*$ establish a pair of pixels which is unambiguously 'connected'. This means that unique, invertible correspondences exist by $B^* = B(m,n) \Leftrightarrow A^* = A(m+\tilde{k}, n+\tilde{l})$, respectively $B^* = B(m+k, n+l) \Leftrightarrow A^* = A(m,n)$; $A^*$ and $B^*$ are still related by integer motion shift $\mathbf{k} = [k,l]^{\mathrm{T}}$, where typically $\tilde{\mathbf{k}} = -\mathbf{k}$. The lifting structure inherently enforces the spatial coordinate relationships as defined in the previous section, where positions in $B$ shall be mapped into identical positions of the lowpass frame $L$, while positions in $A$ shall map into the

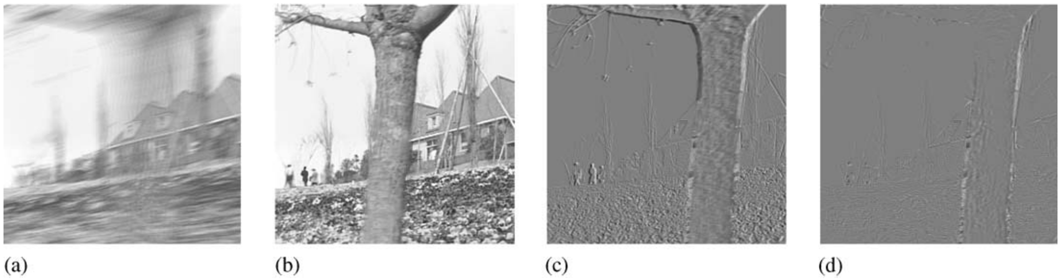


Fig. 5. Frames resulting by temporal-axis wavelet tree over $T=4$ levels: (a) Lowpass frame (LLLL) without motion compensation; (b) with motion compensation; (c) highpass frame (H) without motion compensation; and (d) with motion compensation.

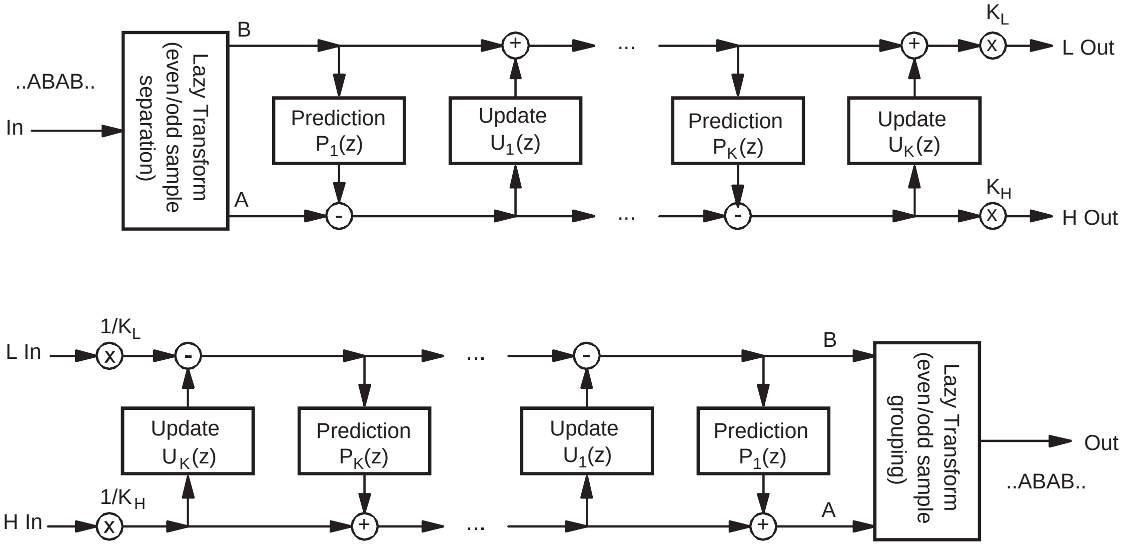


Fig. 6. Lifting structure of a biorthogonal filter.

coordinate reference positions of highpass frame $H$. With pixels connected by unique integer shift, this can be interpreted as a pair of non-orthonormal Haar filters in lifting implementation, where the prediction and update filters are in fact now 3D filters integrating the motion shift, such that $P(\mathbf{z}) = -z_1^k z_2^l$ and $U(\mathbf{z}) = 1/2 z_1^{\tilde{k}} z_2^{\tilde{l}}$,

$$H(m,n) = A(m,n) - B(m+k, n+l),$$

$$L(m,n) = B(m,n) + \tfrac{1}{2} H(m+\tilde{k}, n+\tilde{l}) = \tfrac{1}{2}[B(m,n) + A(m+\tilde{k}, n+\tilde{l})]. \quad (9)$$

The equivalence with (3) is obvious. The consequence of re-defining the motion-compensated Haar filters by a lifting structure are however more fundamental, as the lifting structure is able to guarantee perfect reconstruction in any case, when the same prediction and update filters are used during the reverse operation of synthesis. This means that it will now be possible to release the restriction of full-pixel shifts and gain *perfect reconstruction for arbitrary motion vector fields*. The interpretation by lifting filters was first made in [24,34,36]. A special case had previously been developed in [32], where it was shown that the polyphase kernels of 1D or 2D biorthogonal filter pairs can be used as perfect-reconstructing interpolation filters in the case of a half-pixel accurate motion compensation with temporal-axis Haar filters; the gain achievable by this method in an operational MCTF coding system was first reported in [8].

Assume that in addition to the pixel-wise shift ($k$,$l$), a sub-pixel displacement shall be compensated, such that the actual shift will be $k+\alpha$ horizontally, and $l+\beta$ in vertical direction, $0 \leqslant (\alpha, \beta) < 1$. For simplified explanation, the sub-pixel shift is for this example applied in vertical direction only, where a Haar lifting filter with linear interpolation of sub-pixel positions in the prediction and update steps is used. Further, the forward and backward motion shall be assumed to match the typical case of correct motion flow, $\tilde{\mathbf{k}} = -\mathbf{k}$. A flow diagram is shown in Fig. 7, where it is assumed that the full-pixel shift component has already been considered by aligning the positions of corresponding pixel pairs $A^*$ and $B^*$, which makes the diagram easier to read. For the case $\beta = 0$, the resulting lowpass and highpass samples are identical with (9). For $\beta > 0$, $H$ samples are generated in the prediction step using two-tap linear interpolation, which means that two vertically adjacent pixels from frame $B$ are weighted by factors $\beta$ and $(1-\beta)$ to gain the prediction of the $A$ pixel. In the update step, the $L$ pixels are generated from two adjacent $H$ pixels, weighted by $\beta/2$ and $(1-\beta)/2$. The prediction and update filters can then be described as

$$P(\mathbf{z}) = -z_1^k z_2^l \cdot A(z_2);$$
$$U(\mathbf{z}) = 1/2 z_1^{-k} z_2^{-l} \cdot A(z_2^{-1}) \quad \text{with}$$
$$A(z_2) = (1-\beta) + \beta z_2. \quad (10)$$

The operations to generate the $H$ and $L$ frames are

$$H(m,n) = A(m,n) - (1-\beta)B(m+k, n+l) - \beta B(m+k, n+l+1),$$

$$L(m,n) = B(m,n) + \frac{\beta}{2} H(m-k, n-l-1) + \frac{(1-\beta)}{2} H(m-k, n-l). \quad (11)$$

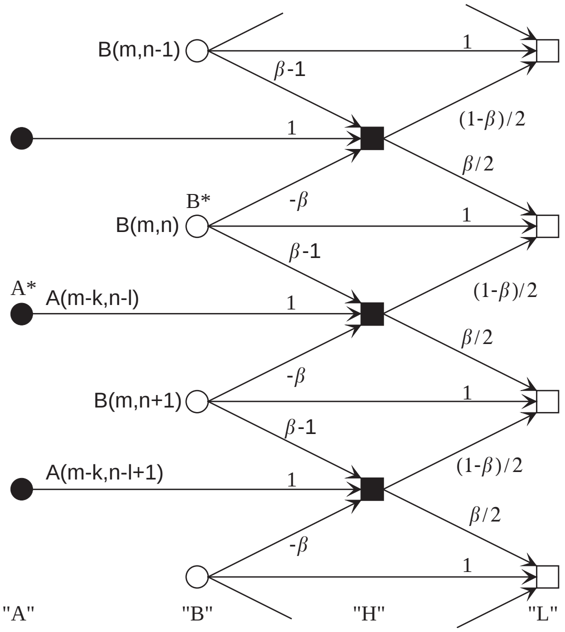


Fig. 7. Signal flow diagram of a motion-compensated lifting filter with sub-pixel shift in one (vertical) dimension; correspondence $A^*/B^*$:$B^*$ is shifted by $k$ pixels horizontally and $l+\beta$ pixels vertically relative to $A^*$.

By inversion of this principle in synthesis, perfect reconstruction is guaranteed, where first the entire frame $B$ must be reconstructed:

$$\begin{aligned}\tilde{B}(m,n) &= L(m,n) - \frac{\beta}{2}H(m-k,n-l-1)\\ &\quad - \frac{(1-\beta)}{2}H(m-k,n-l),\\ \tilde{A}(m,n) &= H(m,n) + (1-\beta)\tilde{B}(m+k,n+l)\\ &\quad + \beta\tilde{B}(m+k,n+l+1). \qquad (12)\end{aligned}$$

For $\beta=0$, (11) equals (9). For $\mathbf{k}=\mathbf{0}$ and $\beta=1/2$, frames $A$ and $B$ can be interpreted as if they were fields of an interlaced frame of double height. Then, (11) is equivalent to the biorthogonal $\frac{5}{3}$ filter pair of transfer function

$$\begin{aligned}H_0^{(5/3)}(z) &= \tfrac{1}{8}(-z^2 + 2\cdot z^1 + 6 + 2\cdot z^{-1} - z^{-2}),\\ H_1^{(5/3)}(z) &= \tfrac{1}{2}(-1 + 2\cdot z^{-1} - z^{-2}) \qquad (13)\end{aligned}$$

applied vertically on this 'big frame'.[4] The concept can straightforwardly be extended into 2D and for higher-quality interpolation filters. This could be realized by integration of higher-order interpolators into the filters $P(\mathbf{z})$ and $U(\mathbf{z})$ of the Haar lifting structure, as will further be discussed below. Alternatively, higher-quality interpolation can also be realized by using the equivalent of longer biorthogonal filters by employing additional lifting and update steps, where however each single prediction/update filter is modified by a simple bilinear interpolation with sub-pixel shift factors $\alpha$ and $\beta$ as described above. The advantage of such a strategy is the inherent integration of highly accurate sub-pixel filters into the spatio–temporal subband transform. The interpretation of sampling position shifts by introduction of branch-weight factors in the lifting flow diagrams in principle allows re-definition of the sub-pixel shifts at each sampling position, though still guaranteeing perfect reconstruction. Further development of such systems would enable a generic wavelet decomposition of irregularly-sampled signals. This could further include the implementation of arbitrary geometric mapping as part of the wavelet synthesis filter and would also allow scaling into arbitrary (non-dyadic) sampling resolutions of the synthesis output.

So far, it was assumed that the integer motion shift $[k,l]$ shall be constant at least over the area which is analyzed. A basic concept to cope with the problem of discontinuities in the motion vector field, by defining reasonable substitutions at the positions of unconnected and multiple-connected pixels, was given in (5). This integrates seamlessly with the lifting filters. Fig. 8a shows the case of an 'unconnected' pixel. Notice that the references $A^*/B^*$ and also the sub-pixel shifts $\beta$ change at the motion discontinuity. In this case, the backward motion vector field diverges from the view of a position in $A$, such that one pixel from $B$ (highlighted by '#') stays isolated.

The multiple-connected case is shown in Fig. 8b. Here, the motion vector field converges by the view of a position in $A$. Again, the references $A^*/B^*$ change at the motion boundary, and one pixel from $A$ (indicated by '#') stays isolated, which means that it is not used in the update filtering step. Pixels from $B$ around the motion boundary may become multiple-connected.

*Lifting filters extended over the temporal axis*: One single analysis level of the wavelet tree, again by view of a pair-wise frame decomposition, is illustrated in Fig. 9a, giving yet another interpretation of the motion-compensated Haar filters. As was shown above, the motion-compensated prediction step in the lifting filter structure (resulting in the $H$ frame) is almost identical with conventional motion-compensated prediction. However, at any transform level, no further recursion is performed evolving from positions of predicted frames $A/H$, such that the motion-compensated wavelet scheme is naturally non-recursive, and it is not necessary to reconstruct frames at the encoder side. The interpolation mechanisms included in the lifting filter structures are now illustrated as simple 'MC' blocks. In fact, for the purpose of sub-pixel accurate motion compensation, arbitrary spatial interpolation filters can be used here; the quality of interpolation should be high in general.

An example how MC operates is given in Fig. 10 for a block-based motion compensation scheme. Here, the block positions are fixed with regard to

[4] This is exactly the solution proposed in [32] for perfect reconstruction MCTF with half-pixel accuracy.

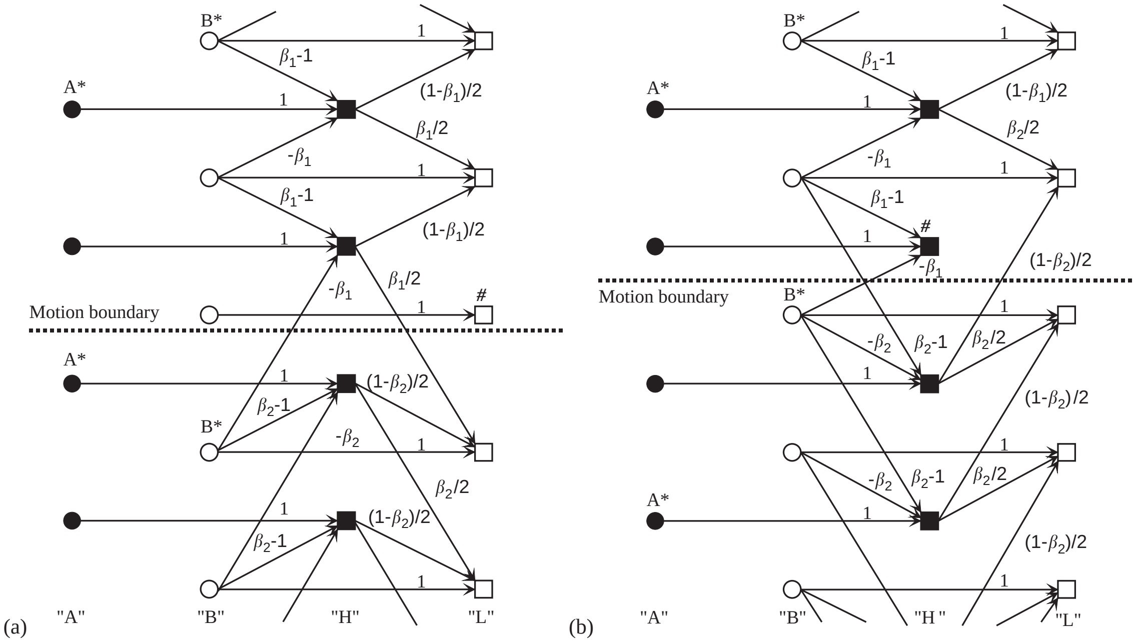


Fig. 8. Lifting structure with cases of (a) unconnected (b) multiple-connected pixels.

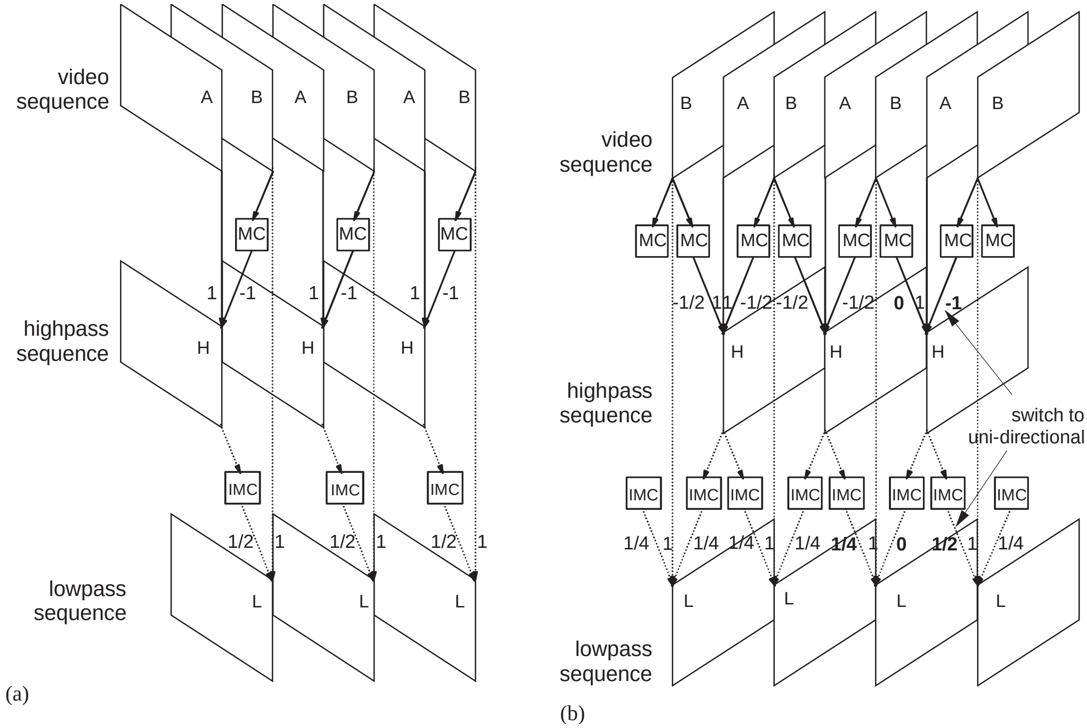


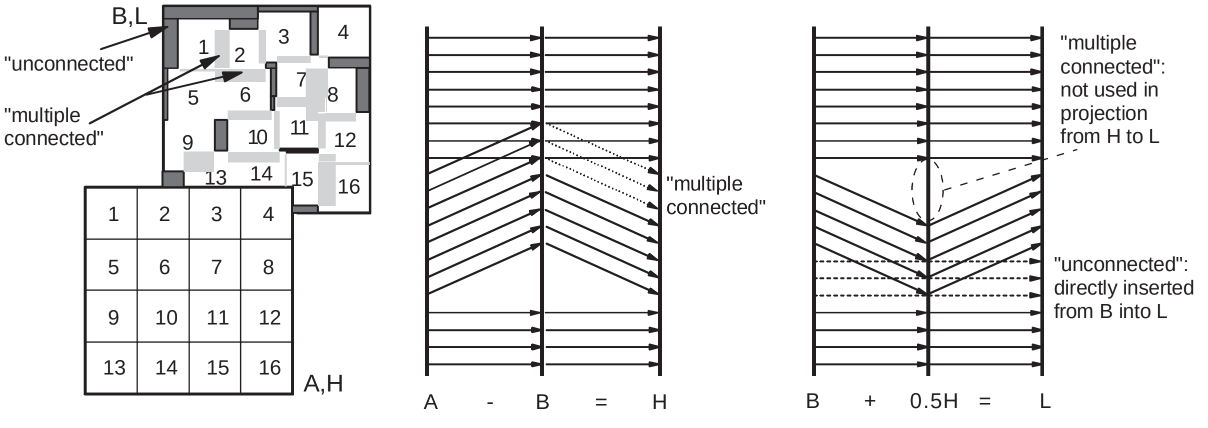
Fig. 9. MC wavelet transformation step $A/B \rightarrow H/L$ in lifting structure: (a) Haar filter with uni-directional prediction and update; (b) $\frac{5}{3}$ filter with bi-directional prediction and update.

the coordinates of frame $A$, which are identical to the coordinates of frame $H$. Hence, it is possible to predict any pixel, regardless of overlapping motion vector fields. The second step of the lifting filter is the update step, which generates the $L$ frame. If this shall be performed in a consistent way in combination with MC, any pixel being mapped from frame $B$ into frame $H$ during the prediction step must be projected back to its original position in the $L$ frame during the update step. This appears reasonable, as the $L$ positions are defined by the same coordinate reference as for pixels in $B$. Hence, the MC applied to $H$, which is used to generate $L$ during the update step, should as close as possible be the inverse of the MC (IMC) that was used during the prediction step. If this is not observed, ghosting artefacts could appear in the lowpass frame, and it would not be fully usable for temporal scalability. As typical in block-based MC, the blocks are fixed in $A$ and $H$ but floating in $B$ and $L$, which has two consequences (see Fig. 10b):

- Pixels which remain blank after IMC are the ‘unconnected’ pixels. As then the information mapped from $H$ into $L$ is zero, original values from $B$ are automatically filled in.
- For duplicate mappings by IMC, a rule must be defined which one is valid; this is the case of “multiple connections”.

The classification into “connected, “unconnected” and “multiple connected” pixels is done at any level of the temporal wavelet tree; in principle, it is not necessary to convey any side information for this purpose, as the classification is uniquely possible from the motion vector field.

It is now also straightforward to extend this scheme into bi-directional frame prediction concepts, which have a good potential to achieve higher coding efficiency than uni-directionally predicted frames for MC prediction coders. The principle is shown in Fig. 9b. Here, also the update step is performed bi-directionally, wherein still the reverse correspondence between MC and IMC must be observed due to the reasons given above. Similar to the case of MC prediction coders, it is also possible to switch dynamically between forward, backward and bi-directional prediction, or implement an intraframe mode. If for example an $H$ frame shall only be computed by the prediction of $A$ from the subsequent $B$, the left-branch weight of the prediction step generating that frame must be set to 0, and the right-branch weight will be set to $-1$. To observe symmetry of the update step, the branch weight corresponding to the zero weight within the prediction step must also be set to 0. An example is shown for the rightmost $H$ frame in Fig. 9b. It should be emphasized that in principle the MC in the prediction and the IMC in the update step could



(a) (b) (c)

Fig. 10. (a) Unconnected and multiple-connected areas in block matching; (b) backward MC in prediction step; (c) projection-based IMC in update step.

be independent. This still would guarantee perfect reconstruction by the inherent properties of the lifting structure, as was shown in [36]). Nevertheless, the match between MC and IMC is important to guarantee undistorted $L$ frames and retain high compression performance.

The filter as realized in Fig. 9b is a $\frac{5}{3}$ biorthogonal filter operating along the temporal axis. An advantage as compared to Haar filters is the *symmetry*, which means that neither the forward nor the backward direction is favored in any way unless explicitly activated by mode switching. By usage of Haar filters, a well-defined frame grouping structure is implicit, where a group of pictures (GoP) of length $2^T$ establishes a self-contained access unit. For the case of $\frac{5}{3}$ temporal filters, the concept of a temporal-axis block transform can be given up; useful decoding could start at any position and would guarantee availability of full information after a limited number of frames decoded, depending on the depth of the temporal wavelet tree. Finally, it must be observed that the number of motion parameters is doubled by introduction of the bi-directional scheme. This can however be avoided by usage of proper motion vector field encoding such as the *direct mode* used in state of the art standards such as MPEG-4 and H.264/AVC.

If the schemes as shown in Fig. 9 are arranged in a wavelet tree as of Fig. 1, the flexibility of temporal scalability would be constrained to dyadic levels, when only the lowpass output shall be used. Cases like down-scaling from 30 to 10 Hz sequences, which is often used in temporal scalability by MC prediction coding, would not be possible. As Fig. 11 shows, a more generalized view of the MCTF lifting concept can overcome this limitation, allowing non-dyadic temporal-axis decompositions. In this example, two frames $B1$ and $B2$ are uni-directionally predicted from one $A$ frame, such that two frames $H1$ and $H2$ are generated in a group with one $L$. Following the MC/IMC principles explained above, the $L$ processing is a binomial filter, $1/4B1 + 1/2A + 1/4B2$.

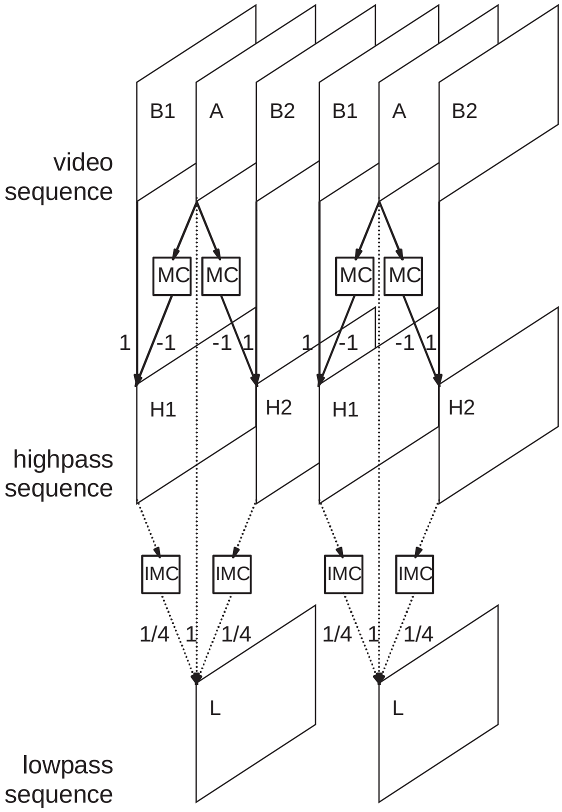


Fig. 11. MC wavelet with non-dyadic transformations $A/B \rightarrow L/H$, ratio of $L{:}H$ is 1:2.

### 2.3. Flexibility of motion compensation in MCTF

*Mode switching*: It is not necessary to enforce all possible references between pixels $A^*$ and $B^*$ from frames $A$ and $B$. It can even be advantageous to embed more 'unconnected' pixels in the $L$ frame when no unique match is found between the two frames. Enforcing the temporal filtering over dissimilar pixels might affect the coding gain, or produce artefacts in the sub-sampled frame sequences, which shall be used for temporal scalability. Typical reliability criteria for motion estimation, in the simplest case unexpectedly high frame differences, can be used as decision criteria. This establishes an equivalent case as switching to intraframe coding in MC prediction. To avoid an amplitude discontinuity, which might be problematic for the subsequent spatial wavelet decomposition, a smooth transition can be realized by gradually adapting the weights of the prediction step between the areas of different type [13]. Another approach of mode switching is the usage of forward motion vectors instead of backward vectors for the multiple-connected pixels in frame $A$, cf. (6).

*De-blocking processing*: Block-based motion compensation, even though being attractive by its low-computational complexity, is unnatural in the context of spatial wavelet decomposition, and a possible source of artefacts. Most of the concepts for improved motion compensation, as developed for MC prediction coders, can be applied in MCTF based wavelet coders likewise. They are even more beneficial here as being a more natural choice for the combination with wavelet basis functions, while block transforms are in better harmony with block-based motion compensation anyway. Usage of variable block size motion compensation was proposed in [30,11], Results on 3D subband and wavelet coders using warping MC were reported in [30,36]. Alternatively, overlapping-block methods can be used [13], which in principle means that weighted superpositions are performed at motion boundaries. The block overlapping method blurs prediction differences in the $H$ frame in the vicinity of motion boundaries, but will also produce more blurred areas in the $L$ frame where the motion is inconsistent. This is beneficial for higher compression efficiency and for the usage of $L$ frame sequences in temporal scalability, achieving better subjective quality (see Fig. 12).

In general, 3D wavelet schemes will take more advantage by *true motion* estimation than hybrid coders do. This can be justified by the fact that for high compression ratios it is very likely that most information contained in the $H$ frames will be suppressed, such that the reconstruction of the original frames is more or less a motion projection from the information contained in the $L$ frames. As no prediction loop exists, it would also consistently be possible to improve the reconstruction quality by integrating methods of frame interpolation into the synthesis process at the different levels of the wavelet tree. Methods for motion estimation as applied in existing 3D wavelet coders have mainly been developed from related hybrid coders, which are typically optimized for the prediction step, but not necessarily *jointly for prediction and update steps*. A first approach to solve this problem was a combined forward/backward motion estimation [29]. Further, criteria can be applied which prefer motion vector fields that are spatially and temporally consistent over the levels of the wavelet pyramid [30]. Rate constraints for variable block size motion vector fields have been introduced, but optimum motion estimation in a rate-distortion sense, where the vector should be applicable over a broad range of rates in a scalable representation, is a problem which is yet unresolved.

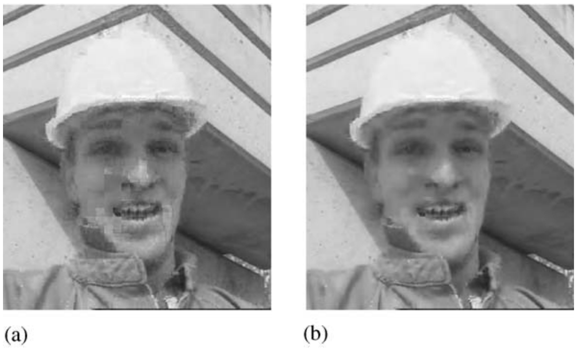


Fig. 12. Reconstruction of video frames from an MCTF wavelet coded representation with block-based MC (a) and with OBMC (b) [Source: HANKE].

### 2.4. Quantization and encoding of 3D wavelet coefficients

The transforms introduced so far in motion-compensated wavelet filtering are not orthonormal. For quantization, it is important to investigate the effect of expected transform domain quantization error to the expected variance of the decoding error. It shall be assumed that a spatial transform is used where orthonormality applies (or at least approximately, as it is, e.g. the case for some bi-orthogonal filters). Then, the aspect of optimum quantization can be analyzed separately for the temporal transform, which could then be realized by linear weights to any spatial coefficient within a given temporal band of the 3D representation. The case of Haar filters is now regarded; a more detailed discussion can be found in [31].

To obtain an orthonormal representation from (9), normalization must be performed such that $H$ frames are multiplied by $K_H = 1/\sqrt{2}$, while $L$ frames must be multiplied by $K_L = \sqrt{2}$. In practice, these up- and down-scaling operations of amplitude ranges must not actually be performed, but can be implemented by proper definitions of quantization step sizes during encoding, which is in particular beneficial for integer implementations. Orthonormality by setting $K_H = 1/\sqrt{2}$ means that the unnormalized $H$ frames in (9) can be encoded with *less accuracy* than a prediction error frame in conventional hybrid MC prediction, while by $K_L = \sqrt{2}$ the unnormalized $L$ frames must be encoded with higher accuracy than $I$ frames.

In fact, the lowpass component must be quantized using double accuracy or half quantization step size as compared to the highpass component. Under normal conditions, half of the highpass quantization error affecting the reconstructed frame $A$ cancels out in the reconstruction process (12), as it is conveyed with a negative sign via the corresponding position in the reconstructed frame $B$. This is however only true if the operations of MC and IMC are *exactly inverses* of each other. Further, except for the case of full-pixel shifts, MC and IMC cannot perfectly match, as sub-pixel interpolations are involved. It follows that the quality of the interpolation filters has a direct influence on the minimization of the reconstruction error.

For the multiple-connected and unconnected positions, it is appropriate to use normalization factors $K_H = K_L = 1$, as the entire quantization error from the $L$ frame is fed into the reconstructed frame $B$, while the quantization error from the $H$ frame exclusively affects $A$ during the reconstruction steps (8). It is hence advisable to adjust the quantization weighting (or the normalization factors) depending on the positions of unconnected and multiple-connected pixels. In principle, to determine the effect of quantization errors accurately, it is necessary to track the evolution of the errors through the entire wavelet tree (temporally and spatially); the additional cost for this optimization is one additional spatio–temporal transform to be performed at the encoder side [29,35].

Fig. 13 shows PSNR results which were obtained by a fully rate-scalable 3D wavelet coder using Haar filters for MCTF and $T = 4$ levels of temporal transform, motion compensation with 1/4 pixel accuracy and a variable-block size motion compensation. For spatial wavelet decomposition was done by a 9/7 bi-orthogonal filter kernel, and encoding was performed by the EZBC algorithm described in [15], including adaptive arithmetic coding. For comparison, results obtained by an H.264/AVC reference software coder (version JM 2.1) are also shown, with all optimization options turned on. It is obvious that the performance of the scalable wavelet coder is very close to a high-performance single-layer standard coder for these sequences. Both coders use block-based motion compensation with variable block sizes down to $4 \times 4$ pixels. Typically, the performance of this particular type of a $t+2$D codec deteriorates towards lower rates, which is due to the fact that no scalability of motion information has been implemented.

The unequal weighting of $L$ and $H$ components is one of the main reasons effecting that 3D transform schemes have potential to become superior in performance compared to hybrid (prediction based) coders. Even though at the first sight the highpass frame $H$ seems to be very similar to an MC prediction frame, the partial compensation of coding errors via the synthesis flow, and the systematic spreading over different adjacent frames give an advantage regarding the total balance of the squared error. A theoretical analysis of this gain is given in [10].

As the $L$ frames (scaling band of the temporal wavelet transform) have to be quantized by increasingly finer quantizers when stepping up the levels of the wavelet tree, this effect increases exponentially by the number of levels; for example, the *LLLL* frame from a four-level temporal Haar wavelet tree contains all relevant information of $2^4 = 16$ frames, and should be quantized by a factor of $\sqrt{2}^4 = 4$ more accurately than a single intra-coded frame. As a by-product, noise, sampling inconsistencies, etc. are discarded at lower rates by the temporal filtering process. From this point of view, motion-compensated wavelet coding can realize advantages of joint multiple-frame

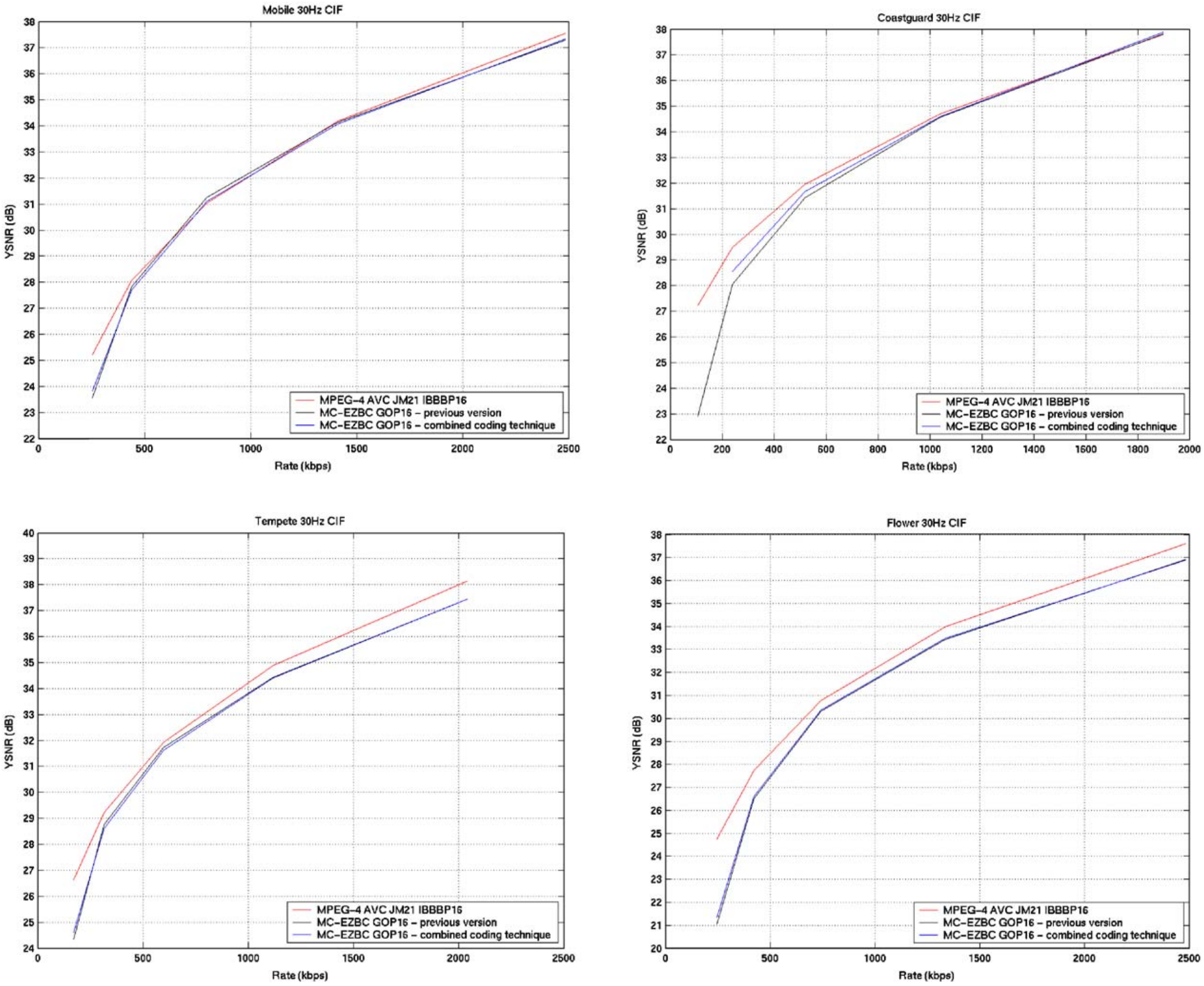


Fig. 13. PSNR results of a motion-compensated 3D wavelet coder (MC-EZBC) on four CIF sequences, compared against the JM 2.1 implementation of the H.264/AVC coder.

compression straightforwardly, which in a hybrid coder could only be achieved by extremely complex look-ahead decisions over a large number of frames.

Methodologies to encode the motion-compensated 3D wavelet coefficients as developed until now are not much different yet from 2D wavelet coding or 3D wavelet coding without MC. Embedded quantizers are used, which can straightforwardly be applied without penalty, as the synthesis filter structure is still non-recursive by principle. Conventional 2D wavelet coders can directly be run on the subband frames resulting by the temporal wavelet tree processing; this is particularly suitable in a configuration where the entire temporal transform is performed first. This case is denoted as a $t+2\text{D}$ transform, corresponding to the scheme shown in Fig. 1.

The optimum strategy of spatio–temporal decomposition is a significant topic of further exploration. The scalability property of the spatial/temporal wavelet transform may, e.g. be utilized to reduce the size of the frame memory necessary to perform encoding and decoding. An example is shown in Fig. 14a, where the spatial size of the frames is reduced by a factor of 4 after each temporal decomposition step (by one level of spatial 2D wavelet transform). Inherently, the depth of the spatial tree is now much lower for the higher temporal frequency bands, which is also reasonable as these signals have less spatial correlation anyway. The related wavelet cube is shown in Fig. 14b. The best spatio–temporal decomposition structure could be found by wavelet-packet design criteria, where the next split in the 3D wavelet tree is made either spatially or temporally, depending on best effect in coding gain. This would implicitly include criteria of temporal similarity between frames and scene cut detection, as the gain by further splitting in temporal direction at the deeper levels of the tree is clearly highest for sequences of low motion. Additional constraints must be set by scalability requirements, such that at least splits which support the required operational ranges of spatial or temporal scalability must be provided by default. As an example, the wavelet cube shown in Fig. 14b would allow spatial scalability between sub-QCIF and HD resolutions spatially, and temporal scalability for frame rates between 7.5 and 60 Hz temporally; for HD indeed, no lower frame rates than 15 Hz would be supported, which appears reasonable.

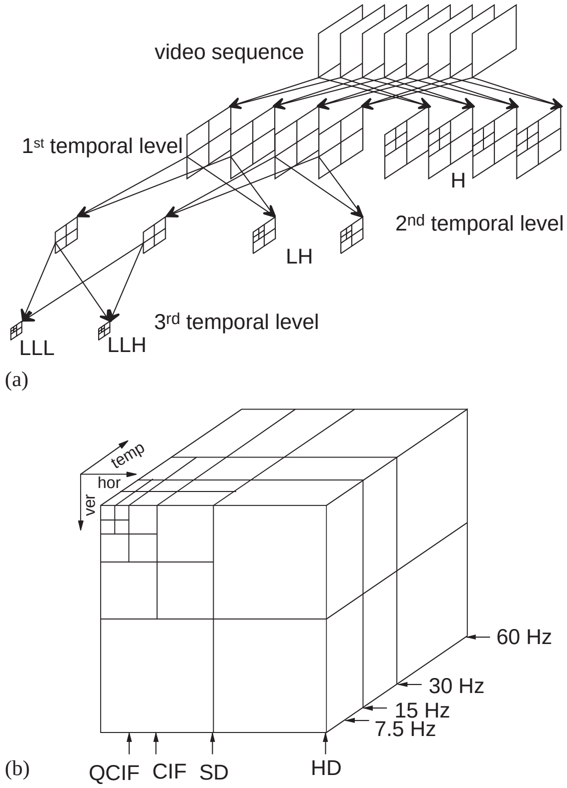


Fig. 14. (a) Wavelet tree with reduction of spatial size throughout the temporal levels; (b) corresponding wavelet cube.

## 3. Switching spatial and temporal transforms

The interframe wavelet video coding schemes presented in the previous section employ MCTF before the spatial wavelet decomposition is performed. Throughout the paper we refer to this class of interframe wavelet video coding schemes as $t+2$D MCTF. Despite their good coding efficiency performance and low complexity, these types of MCTF structures have also several drawbacks:

1. *Limited motion-estimation efficiency*: $t+2$D MCTF are inherently limited by the quality of the matches provided by the employed motion estimation algorithm. For instance, discontinuities in the motion boundaries are represented as high frequencies in the wavelet subbands and the “Intra/Inter” mode switch for motion estimation is not very efficient in $t+2$D MCTF schemes, since the spatial wavelet transform is applied globally and cannot encode efficiently the resulting discontinuities. Moreover, the motion estimation accuracy, motion model and adopted motion estimation block size are fixed for all spatial resolutions, thereby leading to sub-optimum implementations compared with non-scalable coding that can adapt the motion estimation accuracy based on the encoded resolution. Also since the motion vectors are not spatially scalable in $t+2$D MCTF, it is necessary to decode a large set of vectors even at lower resolutions.
2. *Limited efficiency spatial scalability*: If the motion reference during $t+2$D MCTF is e.g. at HD-resolution and decoding is performed at a low resolution (e.g. QCIF), this leads to “subsampling phase drift” for the low resolution video.

3. *Limited spatio–temporal decomposition structures*: In $t+2$D MCTF, the same temporal decomposition scheme is applied for all spatial subbands. Hence, the same levels of temporal scalability are provided independent of the spatial resolution.

A possible solution for the aforementioned drawbacks is to employ "in-band temporal filtering" schemes, where the order of motion estimation and compensation and that of the spatial wavelet transform (2D-DWT) are interchanged, which we denote as 2D$+t$ MCTF schemes. The spatial wavelet transform for each frame is entirely performed first and multiple separate motion compensation loops are used for the various spatial wavelet bands in order to exploit the temporal correlation present in the video sequence (see Fig. 15). In contrast to the method of Fig. 14a, where spatial decomposition steps were interleaved with the temporal tree, MCTF can now also be applied to spatial highpass (wavelet) bands. Subsequently, the coding of the wavelet bands after temporal decorrelation can be done using spatial-domain coding techniques like bitplane coding followed by arithmetic coding, or transform-domain coding techniques based on DCT, wavelets, etc.

### 3.1. Motion estimation and compensation in the overcomplete wavelet domain

Due to the decimation procedure in the spatial wavelet transform, the wavelet coefficient are not shift-invariant with reference to the original signal resolution. Hence, translation motion in the spatial domain cannot be accurately estimated and compensated from the wavelet coefficients, thereby leading to a significant coding efficiency loss (see Haar DWT case in Fig. 16). To avoid this inefficiency, motion estimation and compensation should be performed in the overcomplete wavelet domain rather than in the critically sampled domain (see Haar ODWT case in Fig. 16). The overcomplete discrete wavelet data (ODWT) can be obtained through a similar process as the critically sampled discrete wavelet signals (DWT) by omitting the sub-sampling step. Consequently, the ODWT generates more samples than DWT, but enables accurate wavelet domain motion compensation for the high frequency components, and the signal does not bear frequency-inversion alias components.

Despite that fact that ODWT generates more samples, an ODWT-based encoder still needs to only encode the critically sampled coefficients. This is because the overcomplete transform coefficients can be generated locally within the

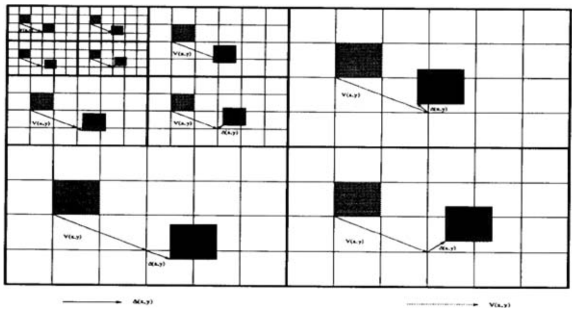


Fig. 15. Multi-resolution motion compensation (MRMC) coder using "in-band prediction".

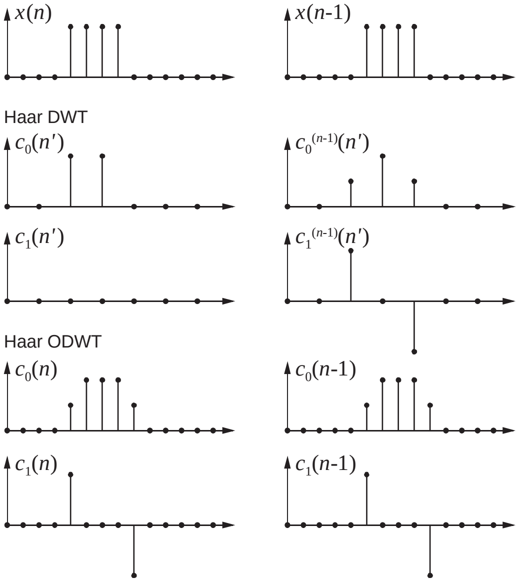


Fig. 16. Shift variance of the Haar wavelet transform. Right signal shifted by one sample to the right, lowpass and highpass coefficients in Haar DWT and Haar ODWT.

decoder. Moreover, when the motion shift is known before analysis and synthesis filtering are performed, it is only necessary to compute those samples of the overcomplete representation that correspond with the actual motion shift.

The $t+2D$ MCTF schemes (Fig. 17a) can be easily modified into $2D+t$ MCTF (Fig. 17b).

More specifically, in $2D+t$ MCTF, the video frames are spatially decomposed into multiple subbands using wavelet filtering, and the temporal correlation within each subband is removed using MCTF (see [2,46]). The residual signal after the MCTF is coded band-by-band using any desired texture coding technique (DCT-based, wavelet-based, matching pursuit, etc.). Also, all the recent advances in MCTF can be employed for the benefit of $2D+t$ schemes, which have been first introduced in [1,2,46,52].

### 3.2. Lifting structure for $2D+t$ MCTF

In order to derive the lifting structure for $2D+t$ MCTF, let us consider a simple two level decomposition of a $B$ frame as shown in Fig. 18. The extension of the lifting equations for $2D+t$ MCTF leads to [52]:

$$H_j^i[m,n] = A_j^i[m,n] - B_j^i[m+k_j^i,\, n+l_j^i],\quad i = 0,\ldots,3; \tag{14}$$

where $k_j^i = k/2^j$, $l_j^i = l/2^j$ , and $(k,l) = (-\tilde{k},-\tilde{l})$ denote the forward and backward motion vectors in full resolution spatial domain. However, in this structure, the interpolation operation for the $B_j^i$ frame is not optimal because it does not incorporate the dependencies of the cross-phase wavelet coefficients. Instead, an interleaving structure

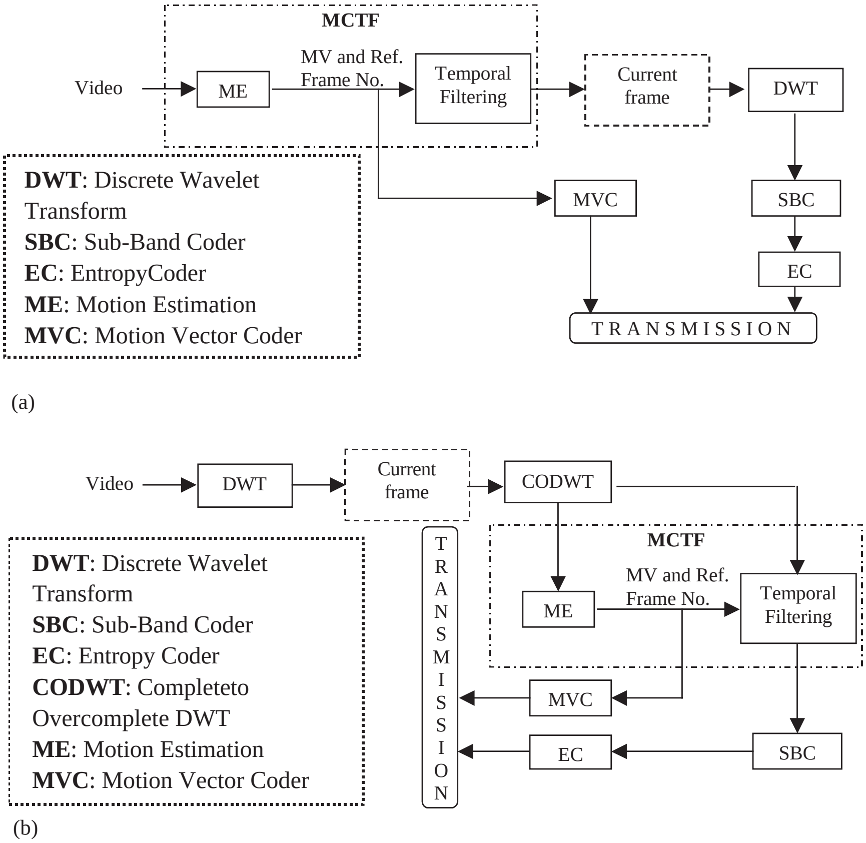


Fig. 17. (a) The encoding structure that performs open-loop encoding in the spatial domain—$t+2D$ MCTF; (b) the encoding structure of that performs open-loop encoding in the wavelet domain (in-band)—$2D+t$ MCTF.

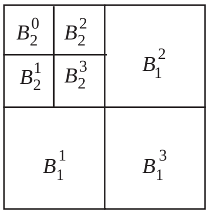


Fig. 18. Two level wavelet decomposition example.

described as the low-band shift (LBS) method [33] provides a *simple and optimal* motion compensated filtering structure:

$$H_j^i[m,n] = A_j^i[m,n] - LBS\_B_j^i[2^j m + k, 2^j n + l], \quad i = 0, \ldots, 3, \tag{15}$$

where $LBS\_B_j^i$ denotes the interleaved overcomplete wavelet coefficients and $LBS\_B_j^i[2^j m + k, 2^j n + l]$ denotes its interpolated pixel value at location $[2^j m + k, 2^j n + l]$. After interleaving, the interpolation operation is a simple spatial domain interpolation of the neighboring wavelet coefficients, similar to that used in $t+2$D MCTF. For the temporally low-pass filtered frame, we have

$$L_j^i[m,n] = \tfrac{1}{2}\, LBS\_H_j^i[2^j m + \tilde{k}, 2^j n + \tilde{l}] + B_j^i[m,n], \quad i = 0, \ldots, 3, \tag{16}$$

where $LBS\_H_j^i$ denotes the interleaved overcomplete wavelet coefficients of the $H_j^i$ frame.

At the decoder side, a perfect reconstruction is still guaranteed:

$$\tilde{B}_j^i[m,n] = L_j^i[m,n] - \tfrac{1}{2}\, LBS\_H_j^i[2^j m + \tilde{k}, 2^j n + \tilde{l}] \tag{17}$$

and

$$\tilde{A}_j^i[m,n] = H_j^i[m,n] + LBS\_\tilde{B}_j^i[2^j m + k, 2^j n + l]. \tag{18}$$

Note that perfect reconstruction can be realized independent on the interpolation method used, as long as the same method is employed at the encoder and decoder.

Unconnected pixels in $A$ are processed as in (16), and unconnected (unreferred) pixels in $B_j^i$ are processed as

$$L_j^i[m,n] = B_j^i[m,n]. \tag{19}$$

### *3.3. Adaptive 2D + t MCTF structures*

The previously described $2\text{D}+t$ MCTF method can adapt the temporal filtering process in the various bands independently based on the spatial resolution, existing temporal correlations and content characteristics [46]. Subsequently, we list briefly the various options enabled by $2\text{D}+t$ MCTF:

- Different accuracy of motion estimation. In $t+2$D interframe wavelet schemes the accuracy of the motion estimation and filtering is fixed. This is unfortunate, since different spatial resolutions require different accuracy. Alternatively, in $2\text{D}+t$ MCTF, the accuracy per band can be varied (each band corresponds to a specific resolution). Hence, for instance, coarse motion accuracy can be employed for the lowest resolution subband while a finer motion accuracy is employed for finer resolution subbands.
- Different prediction structures. Different temporal filters can be used at the various resolutions. For instance, bi-directional temporal filtering can be used for the low bands, while only forward temporal filtering can be used for the higher bands. Choosing a different filter can be done based on minimizing a distortion measure or a complexity measure (e.g. the low bands have less pixels and hence bi-directional and multiple reference temporal filtering can be employed, while for the high-pass bands that have a larger number of pixels, only forward estimation is performed.) For the implementation of the temporal filters, lifting filters can be employed and each prediction and update step can be designed differently for each band of the wavelet domain to optimize the coding efficiency/complexity constraint. Furthermore, adaptive MCTF can be employed to maximize the coding efficiency depending on each subband context [1,52].
- Different GOF structures. The group of frames (GOF) to be filtered together by MCTF can

also be adaptively determined per band. For instance, the LL-bands might have a very large GOF, while the H-bands can use limited GOFs. The GOF sizes can be varied based on the sequence characteristics, complexity or resiliency requirements.

Such a flexible choice of temporal filtering options makes the 2D + *t* MCTF framework deviate from the strict decomposition scheme as performed in *t* + 2D MCTF (Fig. 19a) to a more generic and flexible 3D decomposition scheme, like in the example shown in Fig. 19b.

The true 3D wavelet scalable video-coding scheme proposed in [46] can employ different temporal decomposition levels and GOF sizes for each band. For example, in Fig. 20, GOF sizes for LL, LH (HL), and HH are 8, 4, and 2 frames, respectively, which allow a maximum of temporal decomposition levels of 3, 2, and 1, respectively. In this example, the higher spatial frequency subbands are not filtered using long temporal filters. Also, the motion-vectors for the various bands can be coded differentially to reduce the motion information bits.

This structure allows true spatial scalability. This is because each subband is temporally filtered from the same subband wavelet coefficients, hence loss of information from the finer resolution band does not incur any sub-sampling position drift in the temporal direction.

Another advantage of 2D + *t* MCTF is the complexity scalability of the decoder. Whenever many devices with different computation power and displays access the same scalable bitstream, the decoder with low complexity can decode only low resolution spatial and temporal decomposition level, which incurs only small computational burden, while a decoder with sophisticated decoding power can decode the entire bit stream to reconstruct the full spatial and temporal resolution. Also, the complexity of the temporal filtering or texture coding can increase for higher spatial subbands.

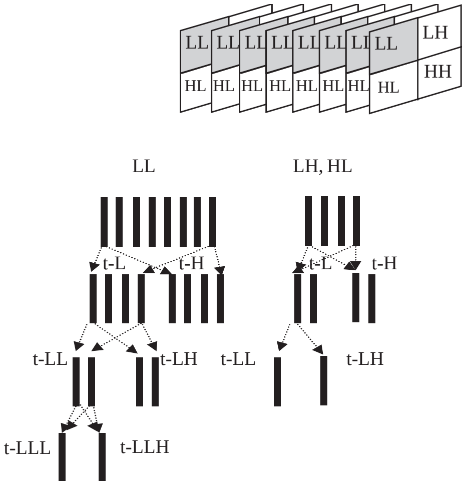


Fig. 20. Example of flexible spatio–temporal decomposition with the proposed framework.

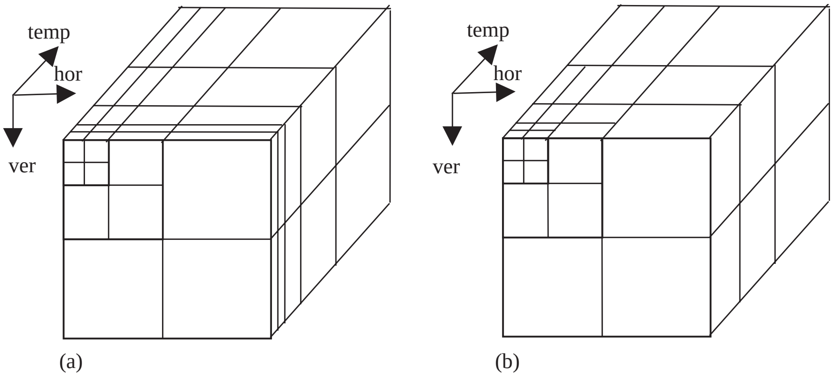


Fig. 19. Example of 3D wavelet decomposition with the *t* + 2D MCTF (a) and 2D + *t* MCTF (b) schemes.

### 3.4. *Standards-compatible interframe wavelet coding techniques*

An important requirement for the new class of interframe wavelet coding schemes is that they should allow for backward compatibility to existing standards. One possibility to provide this backward compatibility is to use JPEG-2000 for the texture coding of $t+2\text{D}$ MCTF schemes. Alternatively, in [3] a standard compliant base-layer has been employed for the low-frequency bands and MCTF coding for the high-frequency bands using the $2\text{D}+t$ MCTF structure. Such a codec described can be employed in many applications that require the base-layer to be compliant with existing coding standards. For instance, digital cinema applications can benefit from having an SD-resolution base-layer that can be viewed using existing consumer products and an efficient higher resolution enhancement-layer.

This standard compliant scheme can be derived as a direct extension of the architecture of Fig. 17b. This is shown in Fig. 21. The low-frequency subband is scaled down and quantized to fit the dynamic range of the standard-compliant MC-DCT coder. During the coding process, the decoded pictures are scaled up and subtracted from the original low-frequency subband content to produce the residual low-frequency signal. This signal is coded by the subband coding technique used for the high-frequency subbands. In many cases the decoded pictures can be generated without performing the actual decoding operation since the MC-DCT framework buffers the coded references during the motion estimation routine. In total, the output bitstream consists of the standard-compliant base-layer and the scalable bitstream of the MCTF-based coding plus the scalable coding of the residual information of the low-frequency band. The latter can be used as an SNR-enhancement layer both for the low and high-resolution decoding. Note that for the enhancement layer compression, any embedded coding scheme can be used. For instance, this embedded coding scheme could be wavelet-based, DCT-based or matching pursuit based.

## 4. Recent advancements in MCTF and 3D wavelet coding

The promise of highly scalable video compression techniques, which are also very efficient in terms of their rate-distortion performance, has led to extensive research and a flood of publications in interframe wavelet coding recently. Here, we briefly highlight some of these recent advancements.

Tillier et al. present in [41] a scalable video codec based on a $\frac{5}{3}$ adaptive temporal lifting decomposition that enables for different adaptation in order

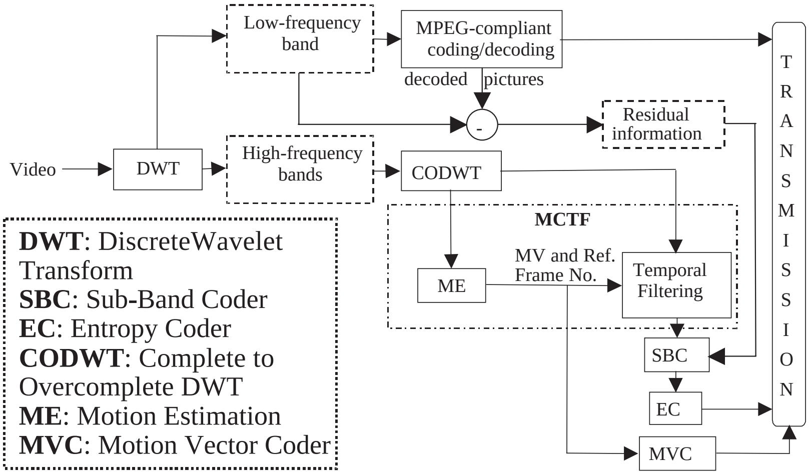


Fig. 21. The modified $2\text{D}+t$ MCTF encoding architecture with standard-compliant base-layer.

to appropriately cope with occluded areas. Furthermore, a memory constrained "on-the-fly" implementation is adopted for simulating the temporal scalability properties of the proposed new structure.

In [10], Flierl and Girod investigate experimentally and theoretically interframe wavelet video using MCTF with dyadic Haar and $\frac{5}{3}$ wavelets. Their results showed the clear superiority of the $\frac{5}{3}$-based MCTF schemes. Based on an ideal signal model and the additivity of estimated displacements, they developed equivalent transforms without displacement operators in the lifting steps. Furthermore, they determined performance bounds with the Karhunen–Loeve transform and observed that interframe wavelet video based on MCTF outperform motion-compensated prediction based video coding schemes by at most 0.5 bits per sample.

In [47], van der Schaar and Turaga presented a new and flexible framework for adaptive temporal filtering in wavelet interframe codecs, called unconstrained motion compensated temporal filtering (UMCTF). This framework allows flexible and efficient temporal filtering by eliminating the update step. UMCTF combines the best features of motion compensation, used in predictive coding, with the advantages of interframe scalable wavelet video coding schemes. UMCTF provides higher coding efficiency, improved visual quality and flexibility of temporal and spatial scalability, higher coding efficiency and lower decoding delay than conventional Haar-based MCTF schemes. Furthermore, UMCTF can also be employed in alternative open-loop scalable coding frameworks using DCT for the texture coding. This method has been extended in [42], where by appropriately choosing the UMCTF "controlling parameters" easy adaptation can be obtained to the desired video/network/device characteristics. The paper describes various content-adaptive filter selection possible in the UMCTF framework. It is shown how this adaptivity can increase both the coding efficiency, as well as the decoded visual quality. In [25], Mehrseresht and Taubman presented an improved method of the adaptive filtering method described in [42] by including the update step. They proposed a new approach to reduce the ghosting artefacts in low-pass temporal subbands by adaptively weighting the update steps according to the energy in the high-pass temporal sub-bands at the corresponding location. Experimental results show that the proposed algorithm can substantially remove ghosting from low-pass temporal frames. Also, they show that the proposed method for adaptively weighting the update steps leads to a better performance as compared with implementations like UMCTF that skip the update step, especially in the presence of additive noise.

A scalable context-based motion vector coding for video compression has been proposed by Valentin et al. in [45]. Their new technique of motion estimation, called constrained motion vectors, allows good estimation with respect to the usual unconstrained search and a reduced rate requirement, leading to a better overall performance of the interframe wavelet coding schemes. The proposed method of motion vector field encoding introduces scalability in motion vectors representation. This property can be exploited in order to improve the scalability of the encoded video stream. In fact, using this motion vector coding technique, a new layered bitstream structure can be created, which contains within the base layer a rough description of motion vector field that can be progressively refined in successive layers. This structure allows a better resource allocation between different scalability layers, as the motion information does not need to be transmitted in a lossless fashion at the beginning of the scalable bitstream.

An alternative scalable motion vector representation and coding has been proposed by Turaga and van der Schaar in [43], where a new method for temporal prediction and differential coding of motion vectors in MCTF is proposed. The paper proposes to exploit the temporal correlations between motion vectors to code and estimate them efficiently. They investigated several prediction methods, and proposed to use motion vector prediction across different temporal decomposition levels in MCTF during motion estimation, i.e. change the search center and the search range based on the prediction, and described the tradeoffs to be made between rate, distortion, and complexity.

This method has been further improved in [44], where the motion estimation complexity is reduced using adaptive selection of UMCTF controlling parameters and adaptive spatio–temporal decomposition order in conjunction with temporal prediction of motion vectors, with variable search ranges. They showed that by performing this tradeoff in an optimized fashion, significantly reduced complexity implementations can be obtained with minor degradation in the R-D performance.

Taubman and Secker also investigated the problem of motion information scalability in [40], where they proposed a wavelet-based highly scalable video coder with scalable motion coding. Their method involves the construction of quality layers for the coded wavelet sample data and a separate set of quality layers for scalably coded motion parameters. When the motion layers are truncated, the decoder receives a quantized version of the motion parameters used to generate the wavelet sample data. A linear model is used to infer the impact of motion quantization on reconstructed video distortion. An optimum tradeoff between the motion and subband bit-rates may then be found. Experimental results indicate that the cost of scalability is small and at low bit-rates, significant improvements are observed relative to lossless coding of the motion.

A method describing several solutions for compressing motion vectors generated by in-band motion estimation is proposed in [5] by Barbarien et al. The presented algorithms are based on motion vector prediction and prediction error coding. The performance of the proposed coding schemes was compared on motion vector sets generated by a hybrid in-band video codec. Their results indicate that the performance of the MV coding algorithms decreases with decreasing quality of the decoded images. The motion vector coding schemes that give the best performance are those based upon either spatio–temporal prediction or spatio–temporal and cross-subband prediction combined with JPEG-alike prediction error coding.

Boisson et al. [6] have investigated different spatio–temporal analysis structures for a fully scalable representation and coding of video signals. In this context, a spatial analysis followed by different techniques to exploit temporal redundancy between consecutive temporal frequency bands has been considered. In particular, a motion compensated spatio–temporal arithmetic coder (MC-STAC) has been evaluated as an alternative solution to MCTF-based interframe wavelet video coding schemes. The importance of the motion information in spatio–temporal context modeling has been proven and even though the proposed MC-STAC does not outperform approaches based on MCTF, both techniques could be combined advantageously by using adaptive selection on the basis of high-temporal frequency energy or of connected/unconnected region criteria.

In [22], Leung and Taubman studied the impact of random accessibility within interframe wavelet coding schemes using adaptive lifting. They compare the merits of several 3D context coding strategies from an information-theoretic perspective and analyzed the variation in random access cost in response to coding parameter adjustments, for a variety of spatial and temporal configurations.

In [27], Munteanu et al. proposes a new framework for the control of the distortion variation in video coding schemes based on MCTF. The distortion in an arbitrary decoded frame at any temporal level in the MCTF pyramid is expressed as a function of the distortions in the reference frames at the same temporal level. The approach is formulated for the bi-directional unconstrained MCTF (UMCTF) scheme of [47], which does not include the update-lifting step. The proposed framework can be extended to the generalized form of MCTF by utilizing additional control parameters. Their experimental results demonstrate the control of the distortion variation in video coding systems based on spatial-domain and wavelet-domain MCTF. The proposed framework provides the means of controlling the tradeoff between the average distortion and the distortion variation in each group-of-pictures (GOPs) within the decoded sequence.

In [51], Xu et al. present a memory efficient transform technique via lifting that effectively computes wavelet transforms of a video sequence continuously on the fly, thus eliminating the boundary effects due to limited length of individual GOPs. Their coding results show that the proposed scheme completely eliminates the boundary effects and gives good video playback quality.

Luo et al. describe in [23] an advanced motion-threading technique to improve the coding efficiency of the 3D wavelet coding, where their original motion-threading technique presented in [24] is extended using the lifting wavelet structure. This extension solves the artificial motion thread truncation problem in long support temporal wavelet filtering, and enables the accuracy of motion alignment to be fractional-pixel with guaranteed perfect reconstruction. Furthermore, the mismatch problem in the motion-threading caused by occlusion or scene-change is considered. Unlike the original motion-threading scheme, in their new proposed scheme each layer owns one set of motion vectors so as to achieve both high coding efficiency and temporal scalability. To reduce the motion cost, direct mode is used to exploit the motion vector correlation. An R-D optimized technique is also introduced to estimate motion vectors and select proper prediction modes for each macroblock.

## 5. The MC-EZBC scalable wavelet coder applied to digital cinema pictures

The basic EZBC image coder [14] has its roots in SWEET [4] and SPECK [16]. These coders replaced zero-trees of the earlier embedded coders with quad-trees, which when restricted to individual subbands, added resolution scalability to the quality or SNR scalability that embedded coders are famous for. The EBCOT coder [39] and the JPEG 2000 standard [17] are similar to these. EZBC provides carefully adjusted context models for the adaptive arithmetic coding to more fully exploit sample/coefficient dependence on both subband and quadtree levels. The EZBC coder was shown to offer slightly better PSNR performance [14] than JPEG 2000 generic scalable mode [17] on the JPEG 2000 test set. EZBC has a potential computational efficiency advantage over EBCOT in that the average number of encoded binary samples per image sample is significantly less, around 1/2 for bit rates of 1.0 bpp or less. This translates into having to revisit less pixels, as the scanning proceeds down the bit planes from MSB to stopping level $\Delta$.

The MC-EZBC video coder [15] replaces the adaptive, non-scalable, finite-state subband coder of [9], with the embedded and scalable compression engine EZBC. Since the 3D subband coder of [9] made use of MCTF, and had no hybrid loops, by combining this front end with an EZBC back end, one is able to transfer all the scalability advantages of embedded subband/wavelet image coders to the video arena. The initial results, published in [9], showed impressive gains of 3 dB versus global motion compensated LZC [38] and 5 dB versus the non-motion compensated 3D-SPIHT [19] on *flower garden* (SIF) at 2.7 Mbps. The MC-EZBC coder was initially presented to MPEG in the context of its digital cinema investigation but then stimulated an investigation on interframe wavelet and advanced scalable video coding. In MPEG's recent *Call for Evidence on Scalable Video Coding Advances* [26], nine coding algorithms were proposed and tested, with the several based on MC-EZBC or quite similar, showing very good performance.

### 5.1. MC-EZBC for digital cinema

The remaining part of this section talks about application of the MC-EZBC coder to the compression of motion pictures of digital cinema content type. To date there are a number of incompatible codecs used for such digital cinema releases, ranging from extensions of the MPEG-2 HD profile to intraframe subband/wavelet methods and variable blocksize DCT. Due to the existence of these divergent approaches, the movie industry had become interested in an international standard that would permit free interchange of materials and allow multiple vendors to compete, thereby lowering equipment costs. MPEG investigated this area through an ad hoc group activity in 2000–2002 and held a test in Hollywood, CA at the Entertainment Technology Center (ETC) of the University of Southern California in Spring 2001. After much statistical analysis and discussion though, the results of this test turned out to be non-definitive, and a clear perspective still needs to be found. Present discussions in the *Digital Cinema Initiative* (DCI) is on a *digital cinema distribution master* (DCDM) of 4096 × 2048 pixels at 4:4:4 resolution in the XYZ tri-stimulus color space, with a 12 bit pixel depth and frame rate of 24 fps. There was some early interest in higher frame rates, e.g. 48 or 72 fps, but the need to go to 4 K, i.e. 4096 × 2048, was the stronger pull, some say in order to differentiate digital cinema from emerging HDTV, which is 1920 × 1080, or almost 2 K, i.e. 2048 × 1024. Initial digital cinema will probably be distributed in both 2 K and 4 K formats in a layered mode. This is because theaters are only now getting true 2 K projectors, while 4 K projectors are thought to be years away. Definitely, there is a need for resolution scalability in digital cinema as currently envisioned.

Several characteristics of digital-cinema type motion pictures make compressing them different from the ordinary CIF and SD resolution test clips used to demonstrate video compression:

1. Very high resolution makes them potentially easier to compress, in that it is expected that the data better match the smooth basis functions used in the coder.
2. As a consequence of this first property, correlation can be expected to exist over a larger number of pixels, hence coders with larger *footprint* than 8 × 8 block transforms should be useful.
3. High noise levels, e.g. grain noise for film origination and high sensor noise for electronic origination make the frames harder to code, and call into question the effectiveness of predictive coding and motion compensation.
4. Much greater bit depth, 12 versus 8 bits, is needed by the fact that the motion picture is shown on a huge screen in a darkened auditorium, permitting the human visual system to adapt to low light levels. Film has a relatively huge dynamic range and motion pictures make good use of it.
5. There is the need for long term constant quality with a control on the average bit rate only, or equivalently total file size for the compressed motion picture. This is different from the usual CBR or VBR coding where only a short buffer is being optimized for PSNR performance. In digital cinema, a very long buffer can be used that may hold the entire compressed movie.

Property 1 suggests that compression will be very important for 4 K digital cinema, while property 2 suggests that a subband/wavelet based method with adjustably large footprint, by control of the number of levels used in the spatial decomposition should fare well in comparison to a method based on 8 × 8 (MPEG-2 and MPEG-4) or 4 × 4 (AVC/H.264) transform blocks.

As a consequence of property 3, the large level of grain noise present, a special $2D+t$ architecture was used, putting the first level of spatial decomposition ahead of the MCTF, but only performing MCTF and subsequent spatial wavelet filtering on the embedded sequence of LL spatial subbands. The diagram for this DC *system* is shown in Fig. 22.

Of course, compressionists know that grain noise should be removed not coded, yet the grain noise is often regarded as important for a "natural" perception of movies. So motion picture coding must allow for the proper reconstruction of the entire original image frame with all, or most of its grain showing. Within the image processing community, an existing approach to deal with noisy originals is to perform possibly motion compensated noise suppression, followed by the video compression. With this approach, artificially synthesized noise can be added at the receiver, with its statistical characteristics parameterized and sent along with the video as meta data. While this has been proposed for digital cinema data [49,12], no consensus yet exists as to whether it would be sufficiently robust and artifact free for the demanding motion picture environment.

Property 4 has led to problems at the projector, due to a limited bit depth of the display sensor chips. Dithering and error diffusion, borrowed from halftone printing, have reportedly been used to address this problem. The film grain noise at higher levels of compression starts to disappear, said to be *crushed*, and is not acceptable to the movie industry. So, some level of dithering and diffusion may be used to help solve this problem, too.

In terms of property 5, a method was developed to potentially get near constant objective quality (PSNR) in MC-EZBC by constraining the entire movie to use the same fractional bit plane of the EZBC spatial coder. We now explain this method with Fig. 23. During the encoding, each GOP is decoded at the various bit planes available and then linear interpolation of bit rate is used in between these levels. Then it is possible to drop a vertical line at the fractional bit plane level where the total of all the estimated GOP bit totals closely targets the required total file size. In addition to this, more constant quality within a GOP is achieved by interleaving the coded bitstreams for all the temporal subbands as well as the spatial subbands. The overall result is a potentially significant increase in uniformity of PSNR throughout the whole movie.

The MC-EZBC coder also has incorporated checks on the quality of motion matches in a mostly successful attempt to restrict MCTF to pixel blocks that have good connection. Also there is an *adaptive GOP size* feature that avoids doing MCTF across shot boundaries, and also during some dissolves where the motion is not unique.

### 5.2. Experimental results

The grain noise characteristics were measured in the test clips that were used in the MPEG DC investigation. We found that the level of the grain noise is much higher than in conventional SD video and SIF test clips. This is shown in the histograms of Fig. 24a for the Y, Cr, and Cb components of the MPEG DC test clip *Beavis*, taken from a flat region in mid luminance range.

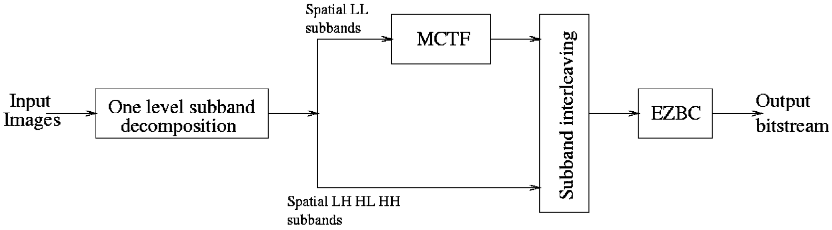


Fig. 22. Diagram of modified MC-EZBC system for digital cinema—DC system.

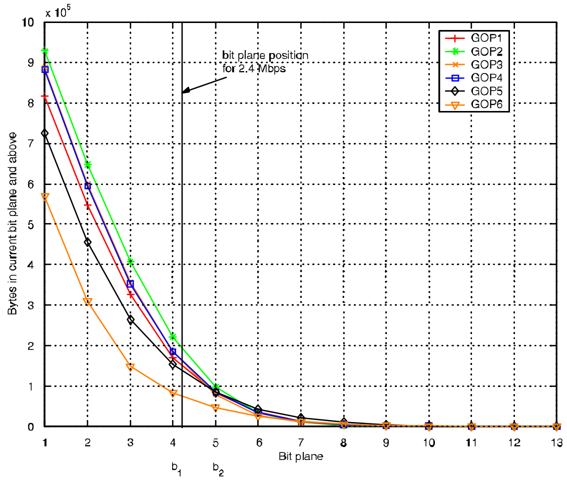


Fig. 23. Simultaneous plot of Rate versus bit plane for all GOPs in a clip or movie, *Table Tennis* sequence (from [7]).

This high (but typical for film) grain noise level motivates the usage of MCTF only on the embedded LL subband sequence. For comparison, a more “normal” amount of noise was found in the Susie test clip with corresponding noise histograms as shown in Fig. 24b.

Table 1 shows some experimental results of intraframe (Intra-EZBC) and MC-EZBC for both MCTF on the full resolution data (T-S) and also for MCTF on the LL subband (DC) for the 1920 × 1080 *car through landscape* clip. The visual improvement due to motion estimation could only be noted at the lowest bit rate of 16 Mbps, where the image looked clearly sharper with MC-EZBC and somewhat soft with Intra-EZBC.

Fig. 25 shows the benefit of interleaving the bit streams from the temporal subbands to get quite near to constant quality (much more uniform PSNR) within the GOP for this CBR coding example on *car through landscape*. This performance was typically also observed with other test clips.

Fig. 26 shows the result of the adaptive GOP size with a max size of 8 that has been a common choice for digital cinema work. The bit rate is 28 Mbps.

The GOP size is often reduced for this clip, even down to one (intraframe), during dissolves and fadeouts. Fig. 27 shows that the method of adaptive GOP size also reduces the fluctuation in PSNR.

A method was implemented to code all frames to the same fractional bit plane, to even out the variations in quality within a motion picture sequence. Using the MPEG DC test clip *stress* results were found as shown in Fig. 28. Obviously, the bit rate can become highly variable in this approach, but the PSNR variation, relative to CBR enforced at the GOP level, is reduced from ±4 to ±2 dB on average for this test clip. The average bit rate was 28 Mbps in this test.

Table 1
PSNR comparison of LL-only MC-EZBC, regular MC-EZBC, and intra-EZBC for the DC test clip *car through landscape*

| Rate (Mbps) | Coder | Y | Cb | Cr |
|---|---|---|---|---|
| 16 | Intra- EZBC | 41.6 | 46.0 | 45.7 |
| | MC-EZBC (T-S) | 41.8 | 45.9 | 45.4 |
| | MC-EZBC (LL) | 42.3 | 46.3 | 45.9 |
| 28 | Intra- EZBC | 43.3 | 47.6 | 47.2 |
| | MC-EZBC (T-S) | 43.4 | 47.0 | 46.5 |
| | MC-EZBC (LL) | 44.2 | 47.5 | 47.0 |
| 48 | Intra- EZBC | 45.7 | 49.8 | 49.3 |
| | MC-EZBC (T-S) | 45.3 | 48.6 | 48.1 |
| | MC-EZBC (LL) | 46.7 | 49.2 | 48.7 |

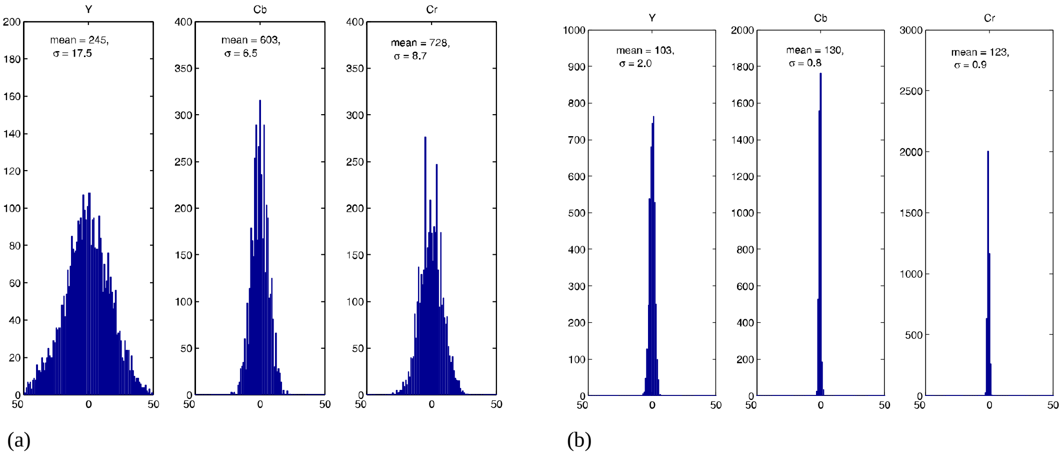


Fig. 24. Histograms of grain noise in color space YCrCb for DC test clip *Beavis* (a) and SD test clip *Susie* (b).

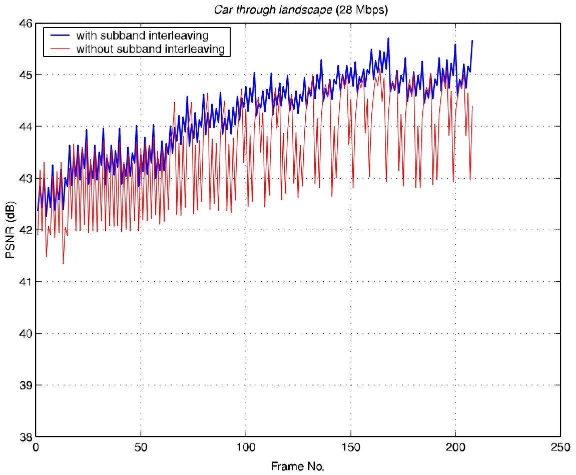


Fig. 25. PSNR plot versus frame number with (blue) and without (red) subband interleaving for car *through landscape*, using CBR coding.

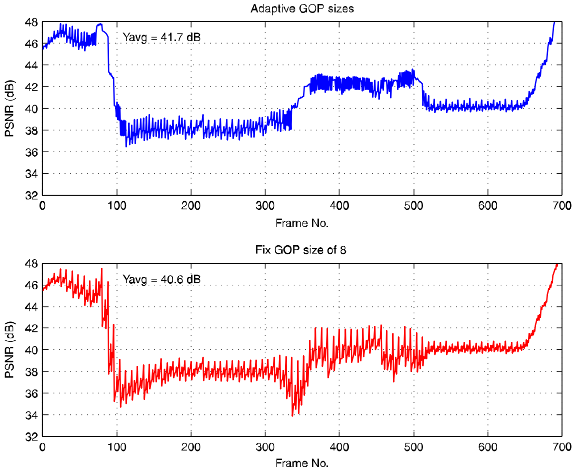


Fig. 27. Plots of PSNR versus frame number with (blue) and without (red) adaptive GOP feature, and using CBR coding.

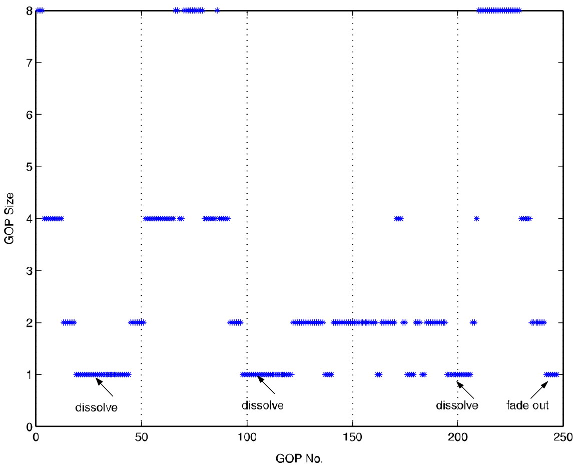


Fig. 26. Illustration of adaptive GOP size decisions with max GOP size = 8 frames.

Just noticeable difference thresholds were incorporated into the MC-EZBC coder, and subjective tests were conducted. Just noticeable difference (JND) was determined using the method of [48]. The bit rates were around 50 Mbps and none of the test ensemble of viewers could see the difference. The JND threshold was then doubled for the high temporal level, whereby the bit rates were obtained shown in Table 2 for three MPEG DC test clips.

A group of 10 subjects was seated at 36 in from a Sony GDM-FM900 CRT monitor with viewable image size 19 × 12.25 in with monitor horizontal and vertical resolution exceeding 1920 × 1080 pixels and 10 bit depth. A two-alternative forced choice (2AFC) test [48] was employed, where the subjects were shown an original sequence, a pause, a test sequence, a pause, and another testing sequence. One of the testing sequences was the original. Then this was repeated for each clip, with random ordering in each trial. The observer was then asked which test clip was the original. The three graphs in Fig. 29 show the number of test subjects, out of 10 graduate students in image processing, who guessed the original correctly. The distributions are centered on 4–5 for *book* and *car through landscape* with a slight bias in favor of the coded version of *as good as it gets*. Thus from this small test, with an ensemble of viewers consisting of graduate students from the Center for Image Processing Research (CIPR) at Rensselaer, it can tentatively be concluded that visual losslessness is achieved for the three clips at these bit rates. Further details can be found in [7].

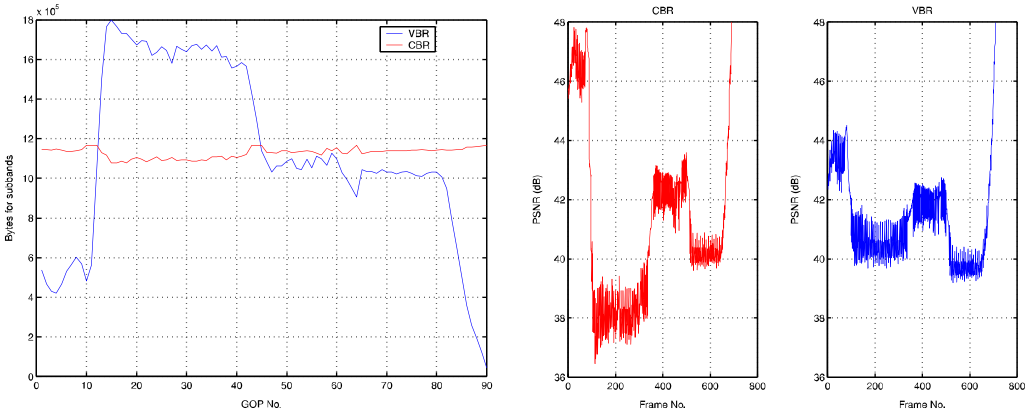


Fig. 28. VBR versus CBR coding results on DC test clip *stress*. Red curves are CBR and blue curves are VBR.

Table 2
Bit rates determined for thresholds used in visual lossless test (from [7]).

| Test clip | Bit rate (Mbps) |
|---|---|
| As good as it gets | 31 |
| Book | 28 |
| Car through landscape | 25 |

## 6. Conclusions

New perspectives in video compression are enforced by the recent advances in MCTF. The non-recursive structure of MCTF based encoders provides high flexibility in bitstream scalability for different temporal, spatial and quality resolutions and better error resilience than conventional (prediction based) coders. In fact, due to the MCTF process the coded representation provides new capabilities to better separate relevant and irrelevant parts of the information. The lowpass frames highlight those information parts of the movie which are consistent over a large number of frames, establishing a means for powerful exploitation of multiple-frame redundancies as hardly achievable by conventional frame-to-frame or multi-frame prediction methods. This is directly supported by the higher encoding accuracy that must be applied on the lowpass frames. On the other hand, noise components and components that express fast changes that cannot be handled by motion compensation, appear in the highpass frames and can be supplemented to the reconstructed signal whenever desirable, provided that sufficient data rate is available. Hence, a denoising process which is often applied as a pre-processing step before conventional video compression, is an implicit part of scalable MCTF-based coders. The gain in PSNR performance, as compared to intraframe coding, is much lower for the case of DC content than for typical video sequences of CIF or SD resolution. Nevertheless, the subjective quality is considerably improved at lower rates, where the grain noise effects severe temporal fluctuation of coding artefacts when a coder without motion compensation is used.

Due to the non-recursive structure, higher degrees of freedom are possible both for encoder and decoder optimization. In principle, a decoder could integrate additional signal synthesis elements whenever the received information is incomplete, such as frame-rate up-conversion, film-grain noise overlay or other elements of texture and motion synthesis, which could easily be integrated as a part of the MCTF synthesis process without losing any synchronization between encoder and decoder. From this point of view, even though in the lifting interpretation many elements of MCTF can be regarded as extensions of proven techniques from MC prediction based coders, this framework exhibits and enables a number of radically new options in video encoding. On the other hand, when a wavelet

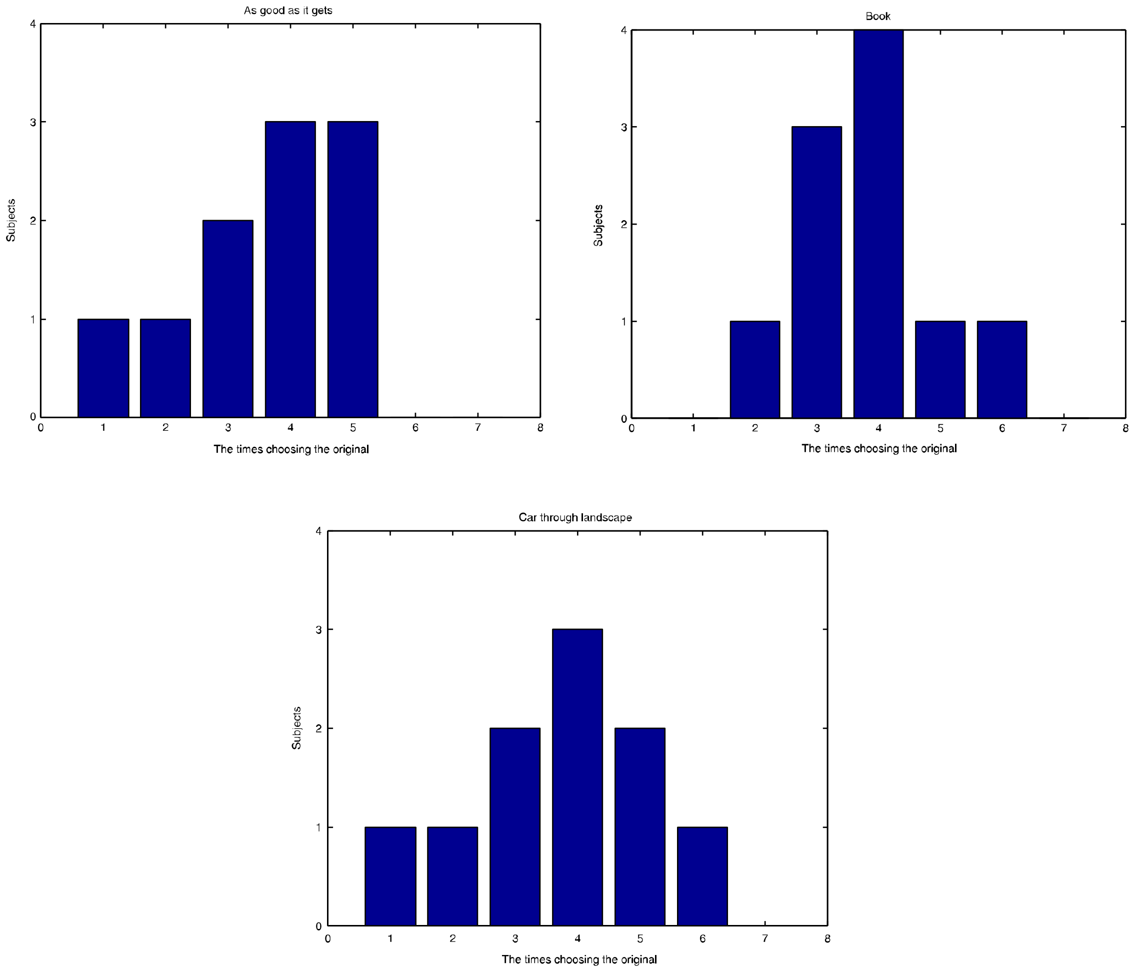


Fig. 29. Plots of number of subjects' correct responses about which DC test clip is original for three DC test clips.

transform is applied for the encoding of the lowpass and highpass frames resulting from the MCTF process, the commonalities with 2D wavelet coding methods are obvious. If the sequence of spatial and temporal filtering is exchanged (2D + $t$ instead of $t$ + 2D wavelet transform), MCTF could also be interpreted as a framework for further interframe compression of (intraframe restricted) 2D wavelet representations such as JPEG 2000. From this point of view, a link between the previously separate worlds of 2D wavelet coding with their excellent scalability properties and compression-efficient motion-compensated video coding schemes is established by MCTF. This shows the high potential for future developments in the area of motion picture compression, even allowing seamless transition between intraframe and interframe coding methods, depending on the application requirements for flexible random access, scalability, high compression and error resilience. Furthermore, scalable protection of content, allowing access management for different resolution qualities of video signals, is a natural companion of scalable compression methods.

Nevertheless, a number of topics can be identified which still require further research, but may also lead to even higher compression performance of this new class of video coding algorithms. These include:

- Strategies for motion estimation and motion vector encoding, including consideration of prediction and update steps, bi-directional prediction and update filtering, as well as combined estimation over different levels of the temporal wavelet tree.
- Application and optimization of non-block based motion compensation, which is more natural to be used in combination with spatial wavelet decomposition.
- Scalability of motion information.
- Optimum adaptation of the spatial/temporal decomposition trees, including consideration of integrated solutions of spatial/temporal filtering.
- Optimization of spatial/temporal encoding, including psychovisual properties.
- Rate-distortion optimum truncation of scalable streams, including the tradeoffs at various rates.

MPEG's recent Call for Proposals for new highly efficient scalable video coding technology and the current plans to develop such a scalable video framework as part of the MPEG-21 standard reflects this situation. Even though it is premature to predict the technical perspectives of such a new standardization effort still under development, it is well possible that the interframe wavelet technologies described in this paper or similar technology developed from this ground could become one of the key players in future video standardization.

## Acknowledgements

The authors would like to acknowledge the collaboration of the following people: Yiannis Andreolopoulos, Peisong Chen, Konstantin Hanke, Thomas Rusert and Jong Ye.

## References

[1] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, Complete-to-overcomplete discrete wavelet transforms for fully scalable video coding with MCTF, Proceeding of the SPIE VCIP, Vol. 5150, 2003, pp. 719—731.

[2] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, J. Cornelis, Fully scalable wavelet video coding using in-band motion-compensated temporal filtering, Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2003.

[3] Y. Andreopoulos, M. van der Schaar, A. Munteanu, P. Schelkens, J. Cornelis, Spatio-temporal-SNR scalable wavelet coding with motion-compensated DCT base-layer architectures, Proceeding of the IEEE International Conference on Image Processing (ICIP), 2003.

[4] J. Andrew, A simple and efficient hierarchical image coder, Proceeding of the IEEE ICIP '97, Vol. 3, 1997, pp. 658–661.

[5] J. Barbarien, Y. Andreopoulos, A. Munteanu, P. Schelkens, J. Cornelis, Coding of motion vectors produced by wavelet-domain motion estimation, Proceeding of the PCS 2003, April 2003.

[6] G. Boisson, E. Francois, D. Thoreau, C. Guillemot, Motion-compensated spatio–temporal context-based arithmetic coding for full scalable video compression, Proceeding of the PCS 2003, April 2003.

[7] P. Chen, Fully scalable subband/wavelet coding, Ph.D. Thesis, Electrical Engineering, Rensselaer Polytechnic Institute, Troy, NY, May 2003.

[8] S.-J. Choi, J.W. Woods, Motion-compensated 3D subband coding of video, IEEE Trans. Image Process. 8 (1999) 155–167.

[9] S.-J. Choi, J.W. Woods, Motion compensated 3D subband coding of video, IEEE Trans. Image Process. 8 (1999) 155–167.

[10] M. Flierl, B. Girod, Investigation of motion-compensated lifted wavelet transforms, Proceeding of the PCS 2003, April 2003.

[11] A. Golwelkar, J.W. Woods, Scalable video compression using longer motion compensated temporal filters, Proceeding of the SPIE VCIP, Vol. 5150, 2003, pp. 1406–1417.

[12] C. Gomila, A. Kobilansky, SEI message for film grain encoding, Joint Video Team doc. JVT-H022, ISO/IEC JTC1/SC29/WG11 and ITU-T SG 16 Q.6, Geneva, May 2003.

[13] K. Hanke, T. Rusert, J.-R. Ohm, Motion-compensated 3D video coding using smooth transitions, Proceeding of the SPIE VCIP, Vol. 5022, 2003, pp. 933–940.

[14] S.-T. Hsiang, J.W. Woods, Embedded image coding using zeroblocks of subband/wavelet coefficients and context modeling, Proceeding of the IEEE ISCAS 2000, Geneva, May 2000.

[15] S.-T. Hsiang, J.W. Woods, Embedded video coding using invertible motion compensated 3D subband/wavelet filter bank, Signal Process. Image Commun. 16 (May 2001) 705–724,

[16] A. Islam, W.A. Pearlman, An embedded and efficient low-complexity hierarchical image coder, Proceeding

of the SPIE VCIP '99, Vol. 3653, San Jose, 1999, pp. 294–305.

[17] JPEG committee, JPEG-2000 VM3.1A Software, ISO/IEC JTC1/SC29/WG1, N1142, January 1999.

[18] G. Karlsson, M. Vetterli, Subband coding of video signals for packet switched networks, Proc. Visual Commun. Image Process. 845 (1987) 446–456.

[19] B.-J. Kim, W.A. Pearlman, An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (SPIHT), Proceeding of the IEEE Data Compression Conference, Snowbird, 1997, pp. 251–260.

[20] B.-J. Kim, Z. Xiong, W.A. Pearlman, Low bit-rate scalable video coding with 3D SPIHT, IEEE Trans. Circuit Systems Video Technol. 10 (2000) 1374–1387.

[21] T. Kronander, Some aspects of perception based image coding, Ph.D. Thesis, Linköping University, 1989.

[22] R. Leung, D. Taubman, Context modeling and accessibility for 3D scalable compression, Proceeding of the IEEE International Conference on Image Processing (ICIP2003), 2003.

[23] Lin Luo, Feng Wu, Shipeng Li, Zhenquan Zhuang, Advanced lifting-based motion-threading technique for the 3D wavelet video coding, Proceeding of the VCIP 2003, Vol. 5150, 2003, pp. 707–718.

[24] L. Luo, J. Li, S. Li, Z. Zhuang, Y.-Q. Zhang, Motion compensated lifting wavelet and its application in video coding, IEEE International Conference on Multimedia and Expo (ICME 2001), Tokyo, Japan, August 2001.

[25] N. Mehrseresht, D. Taubman, Adaptively weighted update steps in motion compensated lifting based on scalable video compression, Proceeding of the IEEE International Conference on Image Processing (ICIP2003), 2003.

[26] MPEG Committee, Report on Call for Evidence on Scalable Video Coding (SVC) Technology, ISO/IEC JTC1/SC29/WG11 MPEG N5701, Trondheim, July 2003.

[27] A. Munteanu, Y. Andreopoulos, M. van der Schaar, P. Schelkens, J. Cornelis, Control of the distortion variation in video coding systems based on motion compensated temporal filtering, Proceeding of the IEEE International Conference on Image Processing (ICIP), 2003.

[28] J.-R. Ohm, A hybrid image coding scheme for ATM networks based on SBC-VQ and tree encoding, Proceedings of the Fourth International Workshop Packet Video, Kyoto, August 1991, pp. B.2-1–B.2-6.

[29] J.-R. Ohm, Three-dimensional subband coding with motion compensation, IEEE Trans. Image Process. 3 (1994) 559–571.

[30] J.-R. Ohm, Motion-compensated 3D subband coding with multiresolution representation of motion parameters, Proceeding of the IEEE ICIP, Vol. III, 1994, pp. 250–254.

[31] J.-R. Ohm, Multimedia Communication Technology, Springer, New York, 2003.

[32] J.-R. Ohm, K. Rümmler, Variable-raster multiresolution video processing with motion compensation techniques, Proceeding of the IEEE ICIP, Vol. I, 1997, pp. 59–762.

[33] H.-W. Park, H.-S. Kim, Motion estimation using low-band-shift method for wavelet-based moving-picture coding, IEEE Trans. Image Process. 9 (4) (April 2000) 577–587.

[34] B. Pesquet-Popescu, V. Bottreau, Three-dimensional lifting schemes for motion-compensated video compression, Proceeding of the IEEE ICASSP, 2001, pp. 1793–1796.

[35] T. Rusert, K. Hanke, J.-R. Ohm, Transition filtering and optimized quantization in interframe wavelet video coding, Proceeding of the SPIE VCIP, Vol. 5150, 2003, pp. 682–694.

[36] A. Secker,D. Taubman, Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting, Proceeding of the IEEE ICIP, 2001, pp. 1029–1032.

[37] W. Sweldens, The lifting scheme: a new philosophy in biorthogonal wavelet constructions, Proc. SPIE 2569 (1995) 68–79.

[38] D. Taubman, A. Zakhor, Multirate 3D subband coding of video, IEEE Trans. Image Process. 3 (1994) 272–288.

[39] D.S. Taubman, M.W. Marcellin, JPEG 2000: Image Compression Fundamentals, Standards and Practice Kluwer Academic Publishers, Dordrecht, 2001.

[40] D. Taubman, A. Secker, Highly scalable video compression with scalable motion coding, Proceeding of the IEEE International Conference on Image Processing (ICIP2003), 2003.

[41] C. Tillier, B. Pesquet-Popescu, Y. Zhan, H. Heijmans, Scalable video compression with temporal lifting using $\frac{5}{3}$ filters, Proceeding of the PCS 2003, April 2003.

[42] D. Turaga, M. van der Schaar, B. Pesquet, Content-adaptive filtering in the UMCTF framework, Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2003.

[43] D. Turaga, M. van der Schaar, B. Pesquet, Differential motion vector coding for scalable coding, Proceeding of the SPIE-Image and Video Communications and Processing, January 2003.

[44] D.S. Turaga, M. van der Schaar, B. Pesquet-Popescu, Reduced complexity spatio–temporal scalable motion compensated wavelet video encoding, IEEE Trans. Circuit Systems Video Technol. (March 2003), submitted for publication.

[45] V. Valentin, M. Cagnazzo, M. Antonini, M. Barlaud, Scalable context-based motion vector coding for video compression, Proceeding of the PCS 2003, April 2003.

[46] M. van der Schaar, J. Ye, Y. Andreopoulos, A. Munteanu, Fully scalable 3D overcomplete wavelet video coding using adaptive motion compensated temporal filtering, M9037, October 2002.

[47] M. van der Schaar, D. Turaga, Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding, Proceeding of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2003.

[48] A.B. Watson, G.Y. Yang, J.A. Solomon, J. Villasenor, Visibility of wavelet quantization noise, IEEE Trans. Image Process. 6 (August 1997) 1164–1175.

[49] J.W. Woods, P. Chen, Coding of animation for digital cinema, ICIP-2002, May 2002, submitted for publication.

[50] J. Xu, S. Li, Y.-Q. Zhang, A wavelet codec using 3D ESCOT, Proceedings of the IEEE-PCM2000, December 2000.

[51] J. Xu, Z. Xiong, S. Li, Y.-Q. Zhang, "Memory-constrained 3D wavelet transform for video coding without boundary effects, IEEE Trans. Circuits Systems Video Technol. 12 (September 2002) 812–818.

[52] J. Ye, M. van der Schaar, Fully scalable 3D overcomplete wavelet video coding using adaptive motion compensated temporal filtering, Proceeding of the SPIE VCIP, 2003.