Advances in Scalable Video Coding

JENS-RAINER OHM, MEMBER, IEEE

Invited Paper

Scalable video coding is attractive due to the capability of reconstructing lower resolution or lower quality signals from partial bit streams. This allows for simple solutions in adaptation to network and terminal capabilities. Different modalities of scalability are specified by video coding standards like MPEG-2 and MPEG-4. This paper gives a short overview over these techniques and analyzes in more detail the encoder/decoder drift problem, which is the major reason why scalable coding has been significantly less efficient than single-layer coding in most of these implementations. Only recently, new scalable video coding technology has evolved, which seems to close the gap of compression performance compared to state of the art single-layer video coding. New methods of efficient enhancement layer prediction were developed to improve traditional (motion-compensated hybrid) scalable coders, providing more flexible compromises on the drift problem. As a new technology trend, motion-compensated spatiotemporal wavelet coding has matured which entirely discards the drift and allows most flexible combinations of spatial, temporal, and signal-to-noise ratio (SNR) scalability with fine granularity over a broad range of data rates.

Keywords*—Motion compensation, motion picture encoding, scalability, wavelet transforms.*

I. INTRODUCTION

In the future, motion pictures will often be transmitted over variable bandwidth channels, both in wireless and cable networks. They have to be stored on media of different capacity, ranging from memory sticks to high-capacity DVD. They have to be displayed on a variety of devices, including a range from small mobile terminals up 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 a simple and flexible solution for transmission over heterogeneous networks, additionally providing adaptability for bandwidth variations and error conditions. Both multicast and unicast streaming applications are possible with minimal processing at server/network and low-complexity decoding. It further allows simple adaptation for a variety of storage devices and terminals. For highest flexibility, scalability providing a fine granularity at the bitstream level and universality in terms of different dimensions is desirable. The most important dimensions are for different spatial, temporal, and quality-level resolutions; the latter is often referred to as signal-to-noise ratio (SNR) scalability.

For video coding, a lack of efficiency can generally 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, which causes a drift problem whenever incomplete information is decoded. This has led to an situation where—even though numerous scalable tools have been integrated into video coding standards available today—a wide acceptance in the market of prospective applications has never occurred.

Hence, research for more efficient scalable coding techniques is a demanding topic in video compression. Provisions are possible which minimize the effect of drift by modifying the structure of the prediction loop. Recent breakthroughs in motion-compensated temporal wavelet filtering, which entirely abandons any recursion in encoding and decoding, have finally enabled implementation of highly efficient scalable video codecs. As a consequence, wavelet video coding schemes can provide flexible spatial, temporal, SNR, and complexity scalability with fine granularity over a large range of bit rates, while maintaining a very good compression performance. These methods can also be interpreted as a superset of established still image wavelet coding techniques like JPEG2000. The inherent prioritization of data in this framework, as well as the availability of mature spatiotemporal wavelet filtering techniques combinable with any kind of motion compensation, leads to added robustness and considerably improved error concealment properties.

The organization of the paper is as follows. Section II summarizes scalability tools as available in existent standards. A technique recently amended to the MPEG-4 video codec is

Manuscript received December 16, 2003; revised May 10, 2004.

The author is with the Institute of Communications Engineering, Aachen University of Technology (RWTH), Aachen D-52074, Germany (e-mail: ohm@ient.rwth-aachen.de).

Digital Object Identifier 10.1109/JPROC.2004.839611

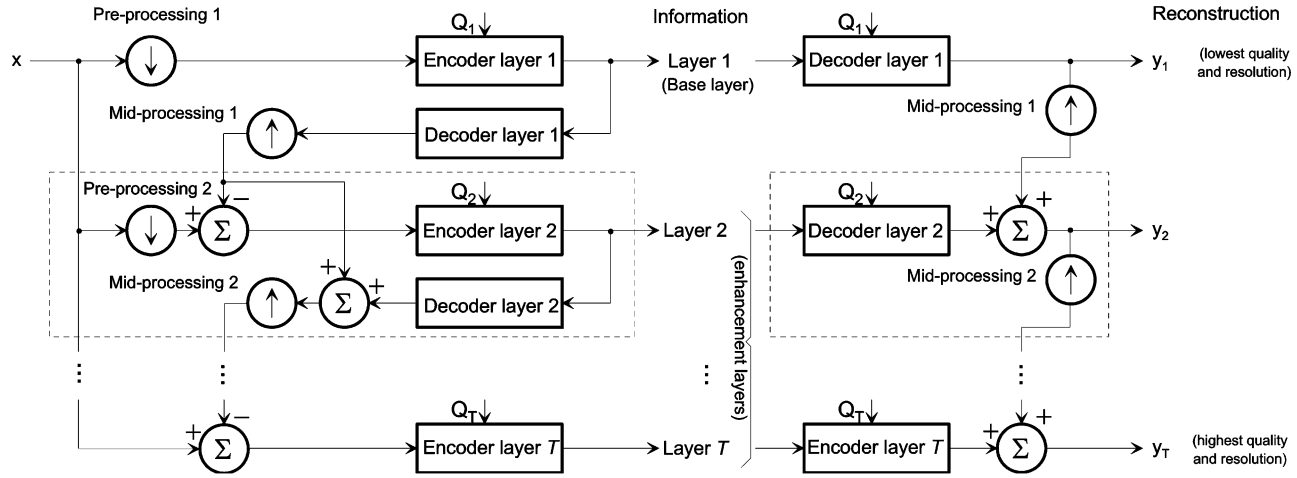

Fig. 1. Principle of scalable coding using T layers.

presented in more detail, which provides the feature of fine granularity scalability (FGS). A more general analysis on the difficulty of combining true bitstream-level scalability with the recursive loop of hybrid coders is given in Section III, which includes discussion of possible workaround solutions. Section IV introduces the framework of motion-compensated temporal filtering (MCTF), which establishes the basis of a fully three-dimensional (3-D) (spatiotemporal) wavelet transform with motion compensation; recent advances in the field are reviewed and summarized. Section V draws conclusions and summarizes future perspectives of scalable video coding.

II. Scalability in Existing Video Coding Standards

Early video compression standards such as ITU-T H.261 [1] and ISO/IEC MPEG-1 [2] did not provide any scalability mechanisms. One reason for this was the dedicated design for specific applications such as conversational services or storage, which did not require scalability. In fact, scalability can nevertheless be achieved by providing different bitstreams targeting at different decoded resolutions: The method of *simulcast* ties together two or several streams for the purpose of parallel transmission, parallel storage can also be implemented. ISO/IEC MPEG-2 [3], which is identical to ITU-T H.262, was the first general-purpose video compression standard which also includes a number of tools providing scalability. One of the reasons was the desire for forward compatibility with MPEG-1, where eventually a base information could be encoded and decoded by the old standard, while higher quality enhancement information is processed by the new standard [4]. MPEG-2 was the first standard to include implementations of *layered coding*, where the standalone availability of enhancement information (without the base layer) is useless, because differential encoding is performed with reference to the base layer. All dimensions of scalability as mentioned above are supported (spatial, temporal, SNR); however, the number of scalable bitstream layers is generally restricted to a maximum of three in any of the existing MPEG-2 profiles. In addition, *data partitioning* allows the separation of the bitstream into different layers, according to the importance of the underlying elements for the quality of the reconstructed signal.

The video codec of the ISO/IEC MPEG-4 standard [5] provides even more flexible scalability tools, including spatial and temporal scalability within a more generic framework, but also SNR scalability with fine granularity and scalability at the level of (eventually semantic) *video objects*. In "Simple Profile" mode, MPEG-4 video is equivalent with the ITU-T H.263 [6] baseline codec, which provides no scalability. Extensions of H.263 define spatial, temporal, and SNR scalabilities as well. *Advanced Video Coding*, as recently defined as part 10 of the MPEG-4 standard [7], aka ITU-T H.264, can in principle be run in different temporal scalability modes, due to its flexibility in the definition of prediction frame references.

The basic approach of FGS as defined in the MPEG-4 standard is a requantization of coefficients in the discrete cosine transform (DCT) domain, where the motion compensation prediction loop of the base layer is self-contained. As a base layer, the standard defines the "Advanced Simple Profile," which is forward compatible with the "Simple Profile," but includes more coding-efficient tools such as B-frames and quarter-pixel accuracy of motion compensation (MC). As clipping to the original value range of the signal may occur in the base-layer prediction loop, it is necessary to use this clipped signal as a reference before the residual error is calculated. Due to this fact, it is necessary to perform the residual computation in the image domain, rather than the DCT domain. The basic approach of DCT residual encoding is a bit plane coding technique, which offers SNR scalability with the desired fine granularity.

Nevertheless, it must be noted that any of the video coding standards existing so far restricts scalability at the bitstream level to a predefined number of *layers* which must be known at the time of encoding.

III. Principles of Scalable Predictive Coding

A very general principle of (layered) scalable coding and decoding is shown in Fig. 1, where by supplementing further building blocks of the intermediate-level type (highlighted

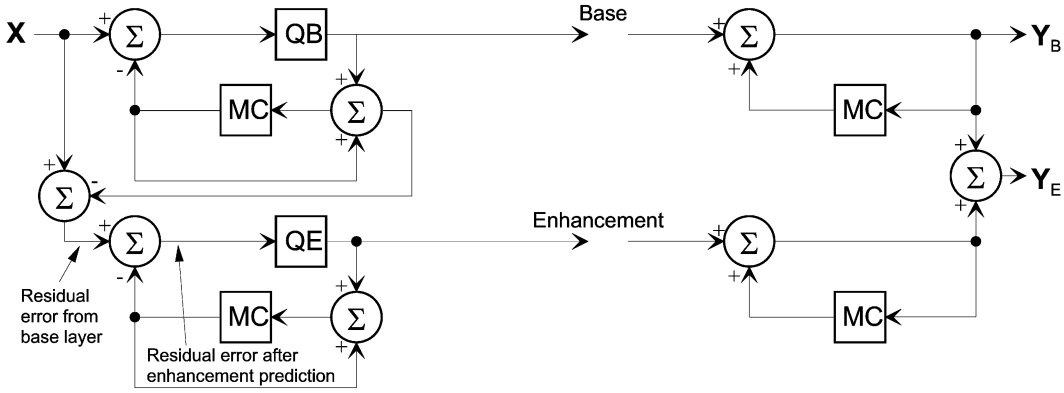

Fig. 2. Two-loop structure for SNR scalability in a hybrid coder (for simplicity, the transform and inverse transform in the loop is omitted).

by a dotted rectangle), an arbitrary number of scalable layers can in principle be realized. The spatiotemporal signal resolution to be represented by the base layer is first generated by decimation (preprocessing). In the subsequent encoding stage, an appropriate setting of the quantizer will then lead to a certain overall quality level of the base information. The base-layer reconstruction is an approximation of all the higher layer resolution levels and can be utilized in the decoding of the subsequent layers. The *midprocessing* unit performs up-sampling of the next lower layer signal to the subsequent layer's resolution. Typically, preprocessing and midprocessing are performed by decimation and interpolation throughout all stages, whereas the particular action to be taken can be quite different depending on the dimension of scalability, e.g., motion-compensated processing can be implemented for frame-rate up-sampling in temporal scalability. The information is propagated from the lower into the higher resolution layers both during encoding and decoding. In all types of scalability (temporal, spatial, or quantization/quality), the constraints imposed by the frame-recursive processing of hybrid video coding have to be carefully considered. The base layer and any composition from layers should in the ideal case be self-contained, which means that the prediction should not use any decoded information from higher layers. Otherwise, different estimates would be used at the encoder and decoder sides, and a *drift effect* would occur. The prediction of the base-layer information will, however, always be worse than it could be if all enhancement layer information was allowed in the prediction. This does not penalize the operation of the coder at the base layer, which will implicitly perform like a single-layer coder at the same rate; however, as the base-layer information is used for prediction of the enhancement layer, the rate-distortion performance toward higher rates will be worse than it could be in a single-layer coder. In an extreme case, which is in fact implemented in MPEG-4 FGS, no temporal prediction at all is performed for the enhancement layers, which may dramatically affect the overall compression performance when the base-layer quality is low. Alternatively, the full enhancement information could blindly be used for prediction;[1] in this case, the reconstruction quality of the highest enhancement layer approaches the performance of a single-layer coder, while the reconstruction quality of the base layer and all intermediate layers would eventually suffer dramatically due to the drift. This basic dilemma to *penalize either enhancement or base-layer* performance is inherently caused by the recursive frame prediction nature of the hybrid coding concept.

[1]This is a mode of *SNR scalability* as defined in the MPEG-2 standard.

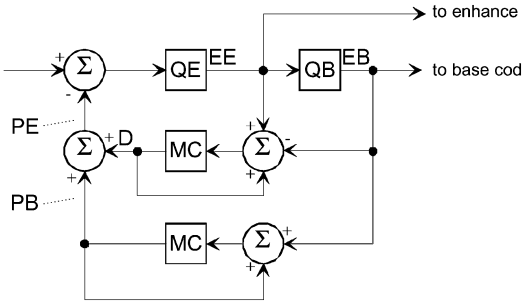

Fig. 3. Hybrid SNR scalability structures tracking the possible drift at the base layer (MC: motion compensation; QB: quantizer of base layer; QE: quantizer of enhancement layer; PE: prediction signal from enhancement layer; PB: prediction signal from base layer).

If base-layer stability shall be observed, it is inevitable to implement a separate prediction loop for the base layer both at the encoder and decoder sides. To achieve higher compression performance, interframe prediction with a separate loop can be applied to the enhancement layer coding (Fig. 2). This will nevertheless still provide a worse rate-distortion performance than a single-layer codec where the full reconstructed information can be used in a single prediction loop.

It is, however, possible in a similar double-loop method to track the drift within the local loop of the encoder that would occur in a decoder only receiving the base-layer information. A basic structure of such a hybrid encoder (omitting the transform for better readability) is shown in Fig. 3. The loop holds two quantizers arranged in a tandem configuration, where the base-layer quantizer (having larger step size) performs requantization of the quantized enhancement information, which is equivalent to residual quantization error encoding. The quantized signals EB (base layer) and EE (enhancement layer) then correspond to representations of the prediction error signals using different accuracy.

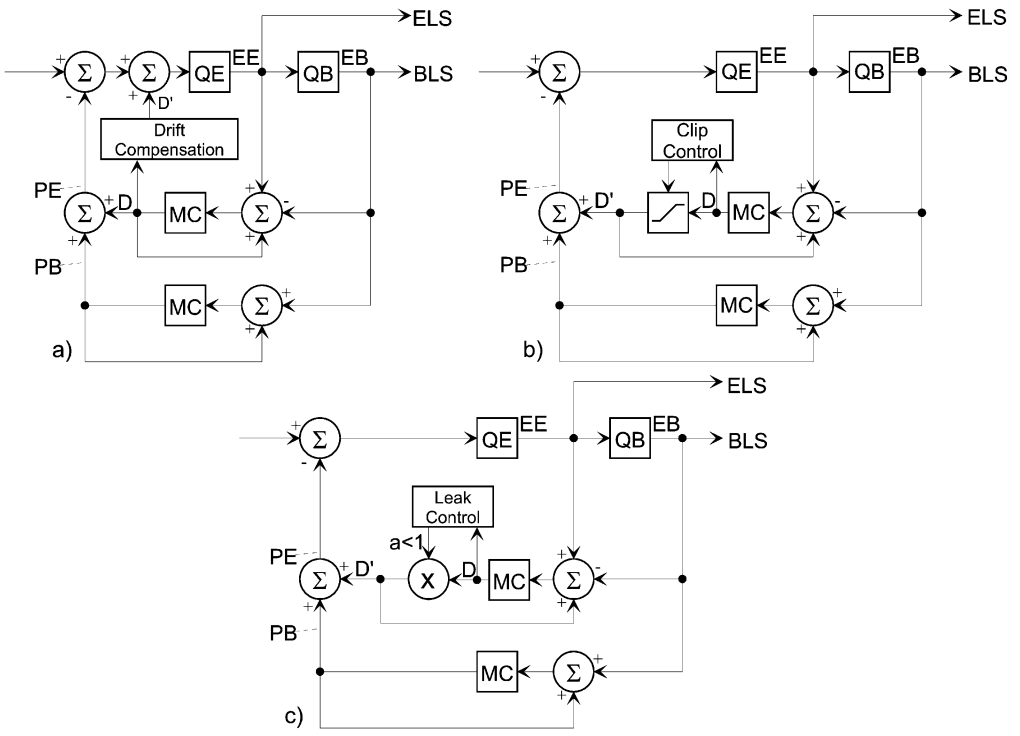

Fig. 4. SNR scalability structures for limitation of drift. (a) Drift compensation. (b) Drift clipping. (c) Drift leaking (BLS/ELS: Base/enhancement layer stream).

These are now formally split to trace the signal flows. The prediction PE corresponds to the full enhancement layer and can be produced as a sum of the base-layer prediction PB and a drift component D, which is generated recursively from the quantization differences between base and enhancement layers. This structure shall now be used to implement a control of drift.

It is obvious that a drift-free base-layer prediction would be possible when only PB was used. Turning off the path of D would be equivalent to an FGS system, which is clearly suboptimum. If the full signal D is used, the enhancement layer prediction can be perfect, but the base-layer quality will suffer due to the drift. Alternatively, it can be tried to keep drift from running out of control. Due to the recursive accumulation of differences EE-EB, D could eventually become much larger than the difference between base and enhancement layer step sizes. This will, however, not be the typical case, but can occur in extreme situations of high temporal changes in the video signal. Different concepts to keep the drift under control are shown in Fig. 4.

- In *drift compensation* [Fig. 4(a)], a value D' is added to the prediction error prior to quantization [8]. For the case $D' = -D$, no drift would occur, but also no usage of enhancement information would be made. For $D' = 0$, the drift would be fully present. It is assumed here that the decoder side is not aware of the drift compensation made, which means that the usual MC decoder loop could be used without any modifications. The problem is to optimize the component D' at the encoder side, such that a good balance between the penalties for the base and enhancement decoding is achieved; this depends on the operational target (whether more advantage should be achieved when operating decoders near base or near enhancement rates). Even though it is a clear advantage to keep the hybrid decoder unchanged and leave the drift control as an issue of encoder optimization, this method performs worse than the following solutions, where the drift control is symmetrically run in the decoder loop as well.
- In *drift clipping* [Fig. 4(b)], the drift is dynamically limited if a maximum value D_{max} is reached. A good choice for D_{max} is approximately by the base-layer quantizer step size. Strategies are then either to set $D' = 0$ or $D' = -D$; the latter method would immediately resynchronize the standalone base-layer loop to the drift-free case, while the first method imposes less penalty on the quality of the enhancement layer. It has

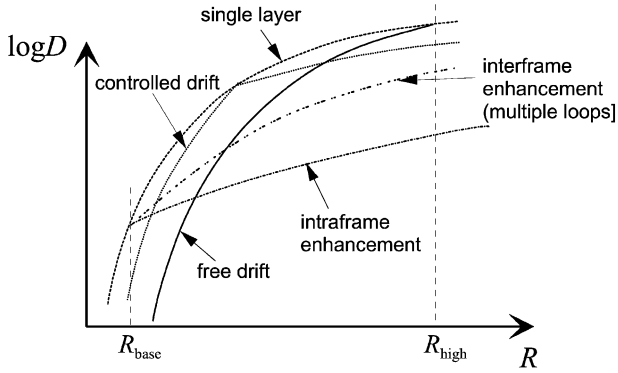

Fig. 5. Qualitative rate-distortion behavior of different scalable hybrid coders.

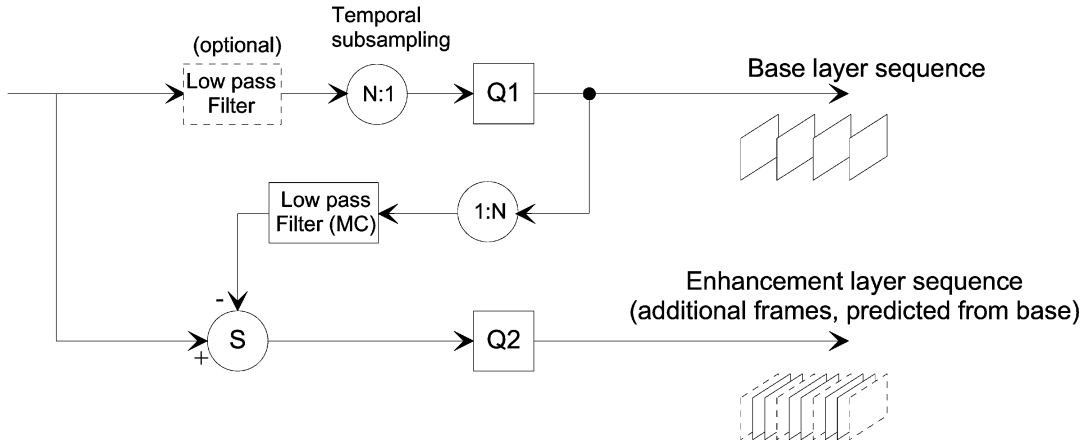

Fig. 6. General concept of temporal scalability based on MC difference (prediction) encoding of enhancement layer.

been shown that the first method gives a better SNR performance over a broad range of rates, when compared to the conventional double-loop configuration of Fig. 2, [8]. Identical drift clipping must be performed at the decoder side. The clipping rules could in principle also be adapted to the signal characteristics, which would then require the transmission of clipping mode parameters as side information. $\mathrm{D}_{\max} = 0$ would correspond to the drift-free case, and $\mathrm{D}_{\max} = \infty$ is the case of unlimited drift.

- In *drift leaking* [Fig. 4(c)], the accumulation of drift is limited by multiplying $\mathrm{D}' = a \cdot \mathrm{D}$ with a leak coefficient $a < 1$. Here, for $a = 0$ the drift-free case and for $a = 1$ the unlimited-drift case are given. The best selection of the drift coefficient is also dependent on the operational target and sequence characteristics, such that an adaptive setting is appropriate. Usage of a similar method has been studied in [9].

Another method of drift control, which is a combination with the double-loop method of Fig. 2 and guarantees unconditional base-layer stability, is denoted as *Progressive FGS* [10]. Here, enhancement layer information of a bit-plane representation is partially used for prediction by a sophisticated prediction mechanism which terminates error propagation after a fixed number of frames in cases where the full enhancement information is not available at the decoder.

In summary, implementation of SNR scalability within hybrid video coders will always cost a penalty as compared to single-layer coders, which is mainly due to the recursive structure and the drift problem. The amount of penalty is sequence dependent, but by the different adaptation mechanisms described above, it would be possible to optimize the performance for different sequence characteristics under the constraints of expected rate targets. Qualitatively, the rate-distortion behavior of the different methods discussed is sketched in Fig. 5, which is in coincidence with measurements that were, e.g., reported in [8] and [11]. It is assumed here that rates are flexibly scalable with fine granularity between lowest and highest rate points. Any of the schemes can in principle be tuned to a best rate point where the performance of a single-layer coder is either exactly obtained

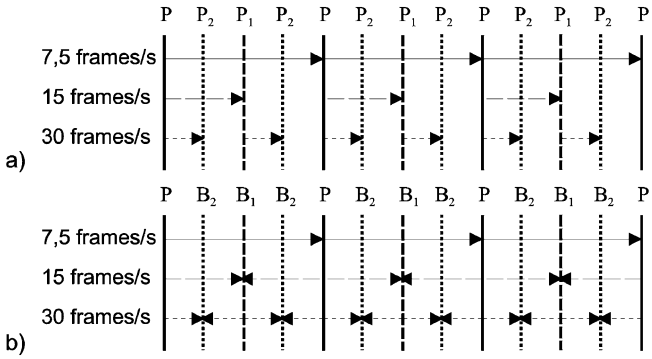

Fig. 7. Temporal scalability with two enhancement layers (1 and 2) supporting three different frame rates. (a) Based on P-type enhancement frames. (b) Based on B-type enhancement frames.

or at least nearly approached. It can be concluded that the methods of drift control establish a reasonable compromise, showing moderate penalty toward both lower and higher rates. When these methods shall cover a sufficiently wide range of rates with sufficient compression performance and stability, implementation of multiple loops is necessary, which results in systems of considerable complexity both at the encoder and decoder sides.

Temporal scalability is often used in practice, as reduction of the video frame rate is a common approach in cases where insufficient transmission capacity is available. Assume that the base layer shall relate to a reconstructed sequence of lower frame rate. If the base information shall be self-contained, it can be established as a subsequence from which frames are skipped, while the enhancement layer supplements these frames for the higher frame rate, which are then predicted from the base-layer frames (Fig. 6). In principle, this will not lead to an increased frame rate as compared to a single-layer coder with same prediction structure, if bidirectionally predicted (B) frames establish the enhancement layer in a hybrid standard coder like MPEG. In this case, the up-sampling filter can be interpreted as a motion-compensated low-pass interpolation filter.

Temporal scalability based on B frames is the only case where scalable hybrid coding may not be inferior as com-

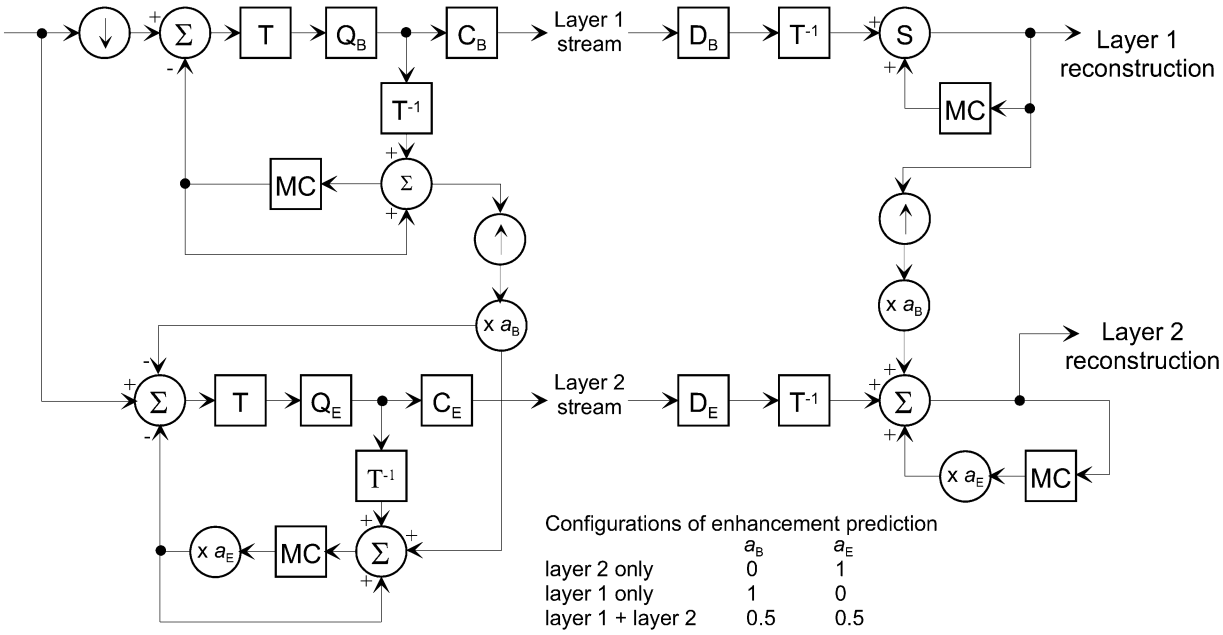

Fig. 8. Spatial scalability combined with quantizer scalability in a hybrid coder: Double loop, supporting switchable MC prediction in the enhancement layer [T: Transform; ↓ / ↑: decimation/interpolation].

pared to a single-layer hybrid coder, as scalability does not change in principle the single-layer compression method. This is the case, as no prediction recursion originates from the B frames which establish the enhancement layer, such that the drift problem does not apply. If P frames shall be used for the enhancement layer, the normal frame-recursive processing sequence must be broken and replaced by a hierarchy of self-contained layers. This necessarily leads to constellations where the distance between the frames and their prediction references becomes larger, which will cause a loss in performance (see Fig. 7(a) for an example of temporal scalability over three different frame rates). A similar scalability configuration can also be realized using B frames [Fig. 7(b)]. Here, unlike in the bidirectional prediction definition introduced earlier, the B-type frames $B1$ of the first enhancement layer are indeed used to predict the frames $B2$ of the second layer; as the number of frames increases by two with each additional level. This can be denoted as a *B-frame pyramid*. This structure can be realized by the flexible frame prediction definitions of the MPEG-4 AVC/H.264 standard.

Observe that in any case, only frames from lower layers are used to predict a frame in a higher layer. As the B_1 frames are partially used to predict the B_2, drift may occur whenever these are not available to the full resolution or are affected by transmission losses. This drift is, however, less severe, as it is by guarantee restricted to one or two steps, and is further diminished by the fact that for the subsequent prediction of another B-type frame, the drift propagates only by a factor of 0.5. In principle, the concept of B frames inherently breaks the infinite prediction loop.

Wavelet transform methods have evolved as an optimum solution for highly efficient scalable coding of still images, and are e.g., implemented in the JPEG2000 standard [12]. In hybrid video coding, incompatibilities of *block-overlapping* wavelet basis functions with *block-based* motion compensation are the main reason for giving a preference to block transforms. This directly influences possible implementations of spatial scalability, where the operations of decimation and interpolation must be applied outside of the prediction loop. Spatial scalability is typically realized as a differential pyramid, where motion compensated prediction is applied within each pyramid level in addition to the coarse-to-fine prediction. These methods can also be interpreted as extensions of the principles for quantizer scalability described above. There may be cases, however, where using *only* the previous frame enhancement layer reconstruction allows better prediction of the actual enhancement frame, without referencing the current base-layer frame. Such a more flexible structure is depicted in Fig. 8. Here, the enhancement layer frame can either be predicted entirely from the up-sampled base layer, from the previous enhancement layer reconstruction, or from the mean value of both. In MPEG-4 video coding, this latter case is also denoted as "bidirectional" prediction mode in spatial scalability, even though it is somewhat different from the original concept of B frames. Nevertheless, spatial scalability in hybrid coding suffers from the same drift problem as SNR scalability.

IV. Interframe Wavelet Coding

To overcome the limitations which are caused by the drift problem, it would be desirable to discard the temporal recursion, which could eventually be done by extending a block transform over the temporal axis as well. Fig. 9 illustrates a 3-D (spatiotemporal) wavelet transform tree, where in the simplest case a Haar basis can be used for

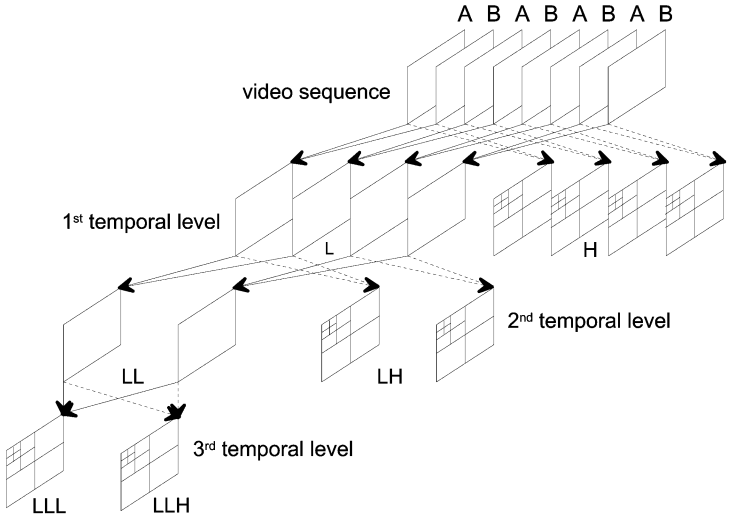

Fig. 9. Spatiotemporal wavelet decomposition using $T = 3$ levels of a temporal wavelet tree.

wavelet decomposition along the temporal axis. Schemes of this type without motion compensation have in fact been proposed more than 15 years ago; see, e.g., [13]. In the case of a nonorthonormal transform, this can be interpreted as decomposition of a frame pair (A, B) into one average (low-pass) and one difference (high-pass) frame

$$\begin{aligned} \mathrm{L}(m,n) &= \frac{1}{2}[\mathrm{A}(m,n) + \mathrm{B}(m,n)]; \\ \mathrm{H}(m,n) &= \mathrm{A}(m,n) - \mathrm{B}(m,n). \end{aligned} \tag{1}$$

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

Application of MC is a key for high compression performance in video coding, but still is often understood 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 along the temporal axis with a motion trajectory [14]. In the case of MC prediction, the filters are in principle linear predictive coding (LPC) analysis and synthesis filters, while in cases of transform or wavelet coding, transform basis functions extended over the temporal axis are subject to MC alignment. This is known as *motion-compensated temporal filtering*. If MCTF is used in combination with a 2-D spatial wavelet transform, this shall be denoted as a 3-D or (depending on the sequence of the spatial and temporal processing) either as a 2-D+t or t+2-D 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. 10(a)], all pixels can ideally be reconstructed by the respective synthesis filtering, where inverse MC mapping must be applied

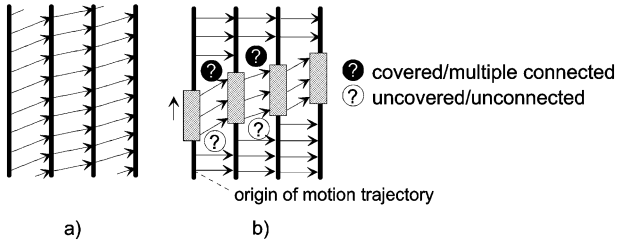

Fig. 10. Forward motion trajectories in the case of: (a) homogeneous and (b) inhomogeneous motion vector fields.

as part of the transform synthesis. In case of nonconstant motion vector fields [Fig. 10(b)], 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*. Another case occurs when motion trajectories converge or merge, which, e.g., happens when areas are being covered. Then, certain coordinate references are *multiple connected*. In the latter case, information would be duplicated in the transform coefficients, while in the former case reconstruction would be impossible, unless the missing part would be sent as side information.

A solution to the problem of unreferenced pixels in the case of Haar filters can be made as follows by redefining the coordinate references with regard to the motion shifts, which was first proposed in [15]. Regard a motion-compensated nonorthonormal Haar filter pair with z transform

$$\begin{aligned} H_0(z) &= \frac{1}{2} + \left(1 + z_1^{\tilde{k}} \cdot z_2^{\tilde{l}} \cdot z_3^{-1}\right) \\ H_1(z) &= -z_1^{k} \cdot z_2^{l} \cdot z_3^{-1}. \end{aligned} \tag{2}$$

The effect of this modification shall again be interpreted by transforming a pair of even/odd indexed frames A and B into one "low-pass" frame L and one "high-pass" frame H, such that

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

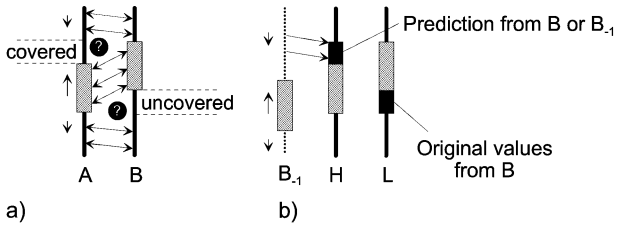

Fig. 11. (a) Covered and uncovered areas in case of frame pairs. (b) Substitution of predictive coded areas into the "high-pass" frame, original frame areas into the "low-pass" frame.

The L frame is the motion-compensated *average*, and the H frame is the motion-compensated *difference* between the two frames. The motion vector $[k,l]^T$ shall characterize the forward motion originating from frame A toward frame B, while $[\tilde{k},\tilde{l}]^T$ describes the backward motion from B toward A.[2] If a unique motion trajectory exists, both motion vectors cannot be independent of each other, as they shall connect corresponding pixels.

The information about remaining "multiple connected" pixels from frame A is integrated as prediction differences into the high-pass frame, while the unconnected pixels from frame B are embedded into the low-pass frame (see Fig. 11)

$$\begin{aligned} \mathrm{L}(m,n) &= \mathrm{B}(m,n), && \text{if unconnected} \\ \mathrm{H}(m,n) &= \mathrm{A}(m,n) - \hat{A}(m,n), && \text{if multiple connected.} \end{aligned} \tag{4}$$

The prediction reference $\hat{\mathrm{A}}(m,n)$ can in principle refer to the (subsequent) frame B or to the preceding frame B_{-1}. Remark that any references to future frames are possible, but incur delay. All operations are now fully invertible. Perfect reconstruction is strictly possible, when full-pixel accuracy of motion compensation is implemented. Motion compensation using subpixel motion shift would, however, lead to lossy reconstruction, as then subpixel 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 [16] that *arbitrary methods of motion compensation* can be used in MCTF and that the reconstruction error can be made reasonably small when interpolators of high quality are used to compute the subpixel positions.

Fig. 12 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 detailed information yet. In principle, the high-pass frame shows the same behavior as a prediction error frame without motion compensation. In the motion-compensated case, the low-pass 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 subsampled sequence which can be displayed at lower frame rate. It is obvious that spatiotemporal wavelet coding *without MC* can hardly be used for the purpose of temporal scalability.

Any pair of biorthogonal wavelet filters can be implemented by the *lifting structure* as shown in Fig. 13 [17]. The first step of the lifting filter is a decomposition of the signal into its even- and odd-indexed 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 typical filter lengths two to three; 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 low-pass frame L and one high-pass 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 $\mathrm{B}^* = \mathrm{B}(m,n) \Leftrightarrow \mathrm{A}^* = \mathrm{A}(m+\tilde{k}, n+\tilde{l})$, respectively, $\mathrm{B}^* = \mathrm{B}(m+k, n+l) \Leftrightarrow \mathrm{A}^* = \mathrm{A}(m,n)$; A^* and B^* may in this first consideration still be 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 low-pass frame L, while positions in A shall map into the coordinate reference positions of high-pass frame H. With pixels connected by unique integer shift, this can be interpreted as a pair of nonorthonormal Haar filters in lifting implementation, where the prediction and update filters are in fact now 3-D filters including the motion shift, such that

$$\begin{aligned} P(\mathbf{z}) = -z_1^k z_2^l \quad \text{and} \quad U(\mathbf{z}) &= 1/2 z_1^{\tilde{k}} z_2^{\tilde{l}}, \\ H(m,n) &= \mathrm{A}(m,n) - B(m+k, n+l) \\ L(m,n) &= B(m,n) + \frac{1}{2} H(m+\tilde{k}, n+\tilde{l}) \\ &= \frac{1}{2}[B(m,n) + \mathrm{A}(m+\tilde{k}, n+\tilde{l})]. \end{aligned} \tag{5}$$

The equivalence with (3) is obvious. The consequence of redefining 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*,

[2] In the following, it will generally be assumed that the spatial coordinate system and time references of H is related to frame A, while L likewise relates to B. These reference definitions are somewhat arbitrary and can be made *vice versa* without any restriction or might even be changed dynamically.

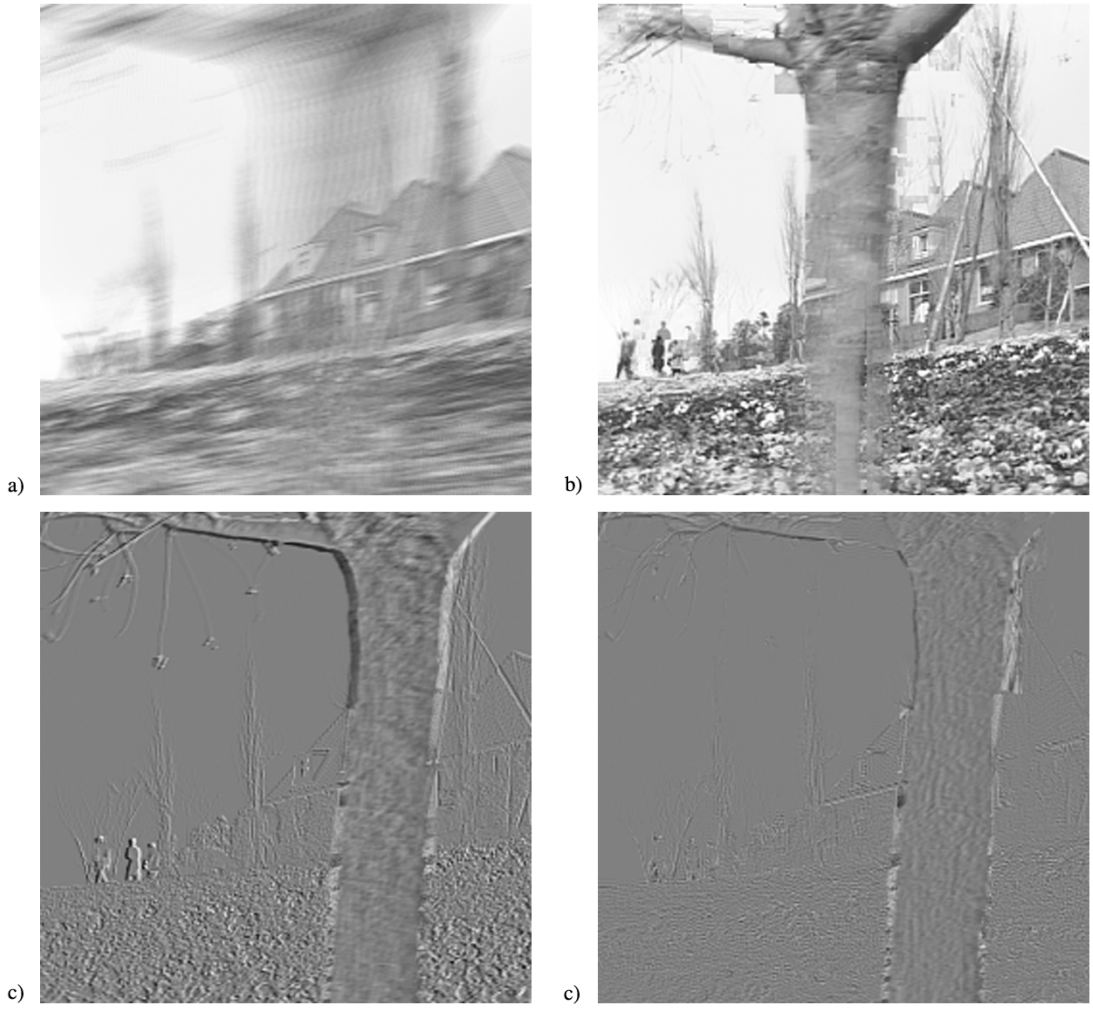

Fig. 12. Frames resulting by temporal-axis wavelet tree over $T = 4$ levels. (a) Low-pass frame (LLLL) without motion compensation and (b) with motion compensation. (c) High-pass frame (H) without motion compensation and (d) with motion compensation.

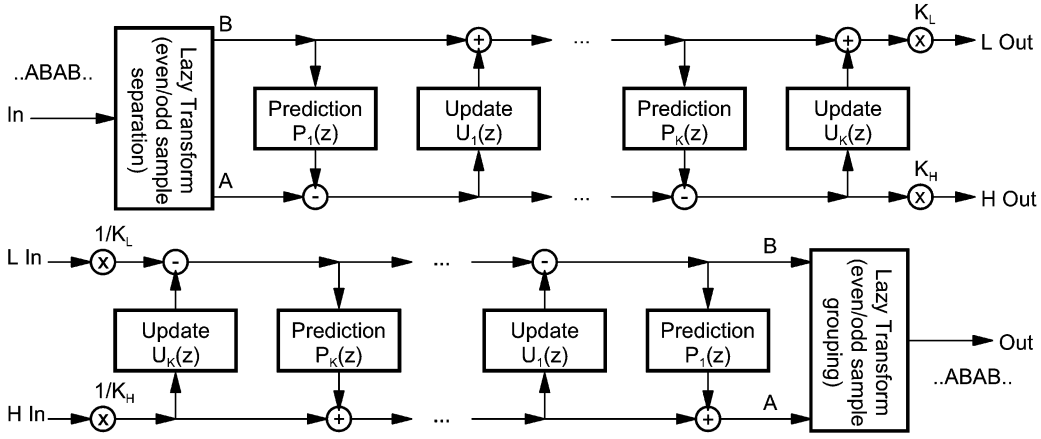

Fig. 13. Lifting structure of a biorthogonal filter.

when motion compensation and possibly subpixel value interpolation are included in the lifting branches. This interpretation of implementing MCTF by lifting filters was first made in [18]–[20]. A special case had previously been proposed in [21], where it was shown that the polyphase kernels of one-dimensional or 2-D biorthogonal filter pairs can be used as perfect-reconstructing subpixel interpolation filters interpreted as modified temporal-axis Haar filters; an implementation of this latter method in an operational MCTF coding system was first reported in [23].

One single analysis level of the wavelet tree, again by view of a pairwise frame decomposition, is illustrated in

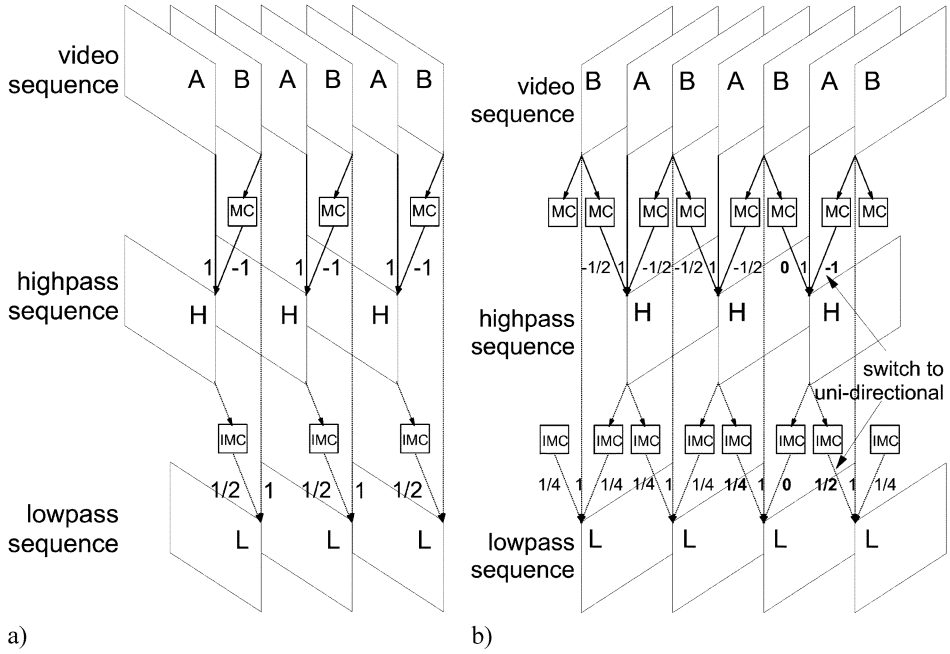

Fig. 14. MC wavelet transformation step A/B → H/L in lifting structure. (a) Haar filter with unidirectional prediction and update. (b) 5/3 filter with bidirectional prediction and update.

Fig. 14(a), 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 to 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 nonrecursive, 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 subpixel accurate motion compensation, arbitrary spatial interpolation filters can be used here; the quality of interpolation should be high in general.

A. Operation of Motion Compensation

An illustration of how MC operates is given in Fig. 15 for the example of a block-based motion compensation scheme. Here, the block positions are fixed with regard to 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 references 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 would not be observed, ghosting artifacts could appear in the low-pass frames, such that these would not be fully usable for the purpose of 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. 15(b)].

- 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."

It is now also straightforward to extend this scheme into bidirectional frame prediction concepts, which have a good potential to achieve higher coding efficiency than unidirectionally predicted frames for MC prediction coders. The principle is shown in Fig. 14(b). Here also, the update step is performed bidirectionally, 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 bidirectional prediction or to 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 zero, 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 should also be set to zero: otherwise, any coding error from H would additionally spread into the previous frame B during reconstruction. An example is shown for the rightmost H frame in Fig. 14(b). Potentially, application of bidirectional prediction and update would increase the motion vector rate. This can,

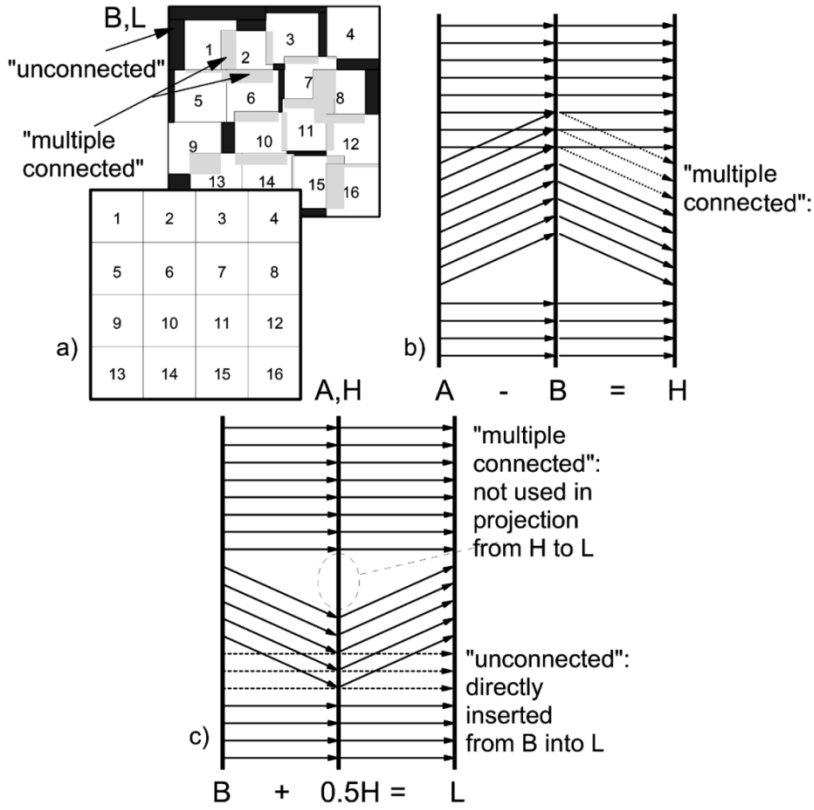

Fig. 15. (a) Unconnected and multiple-connected areas in block matching. (b) Backward MC in prediction step. (c) Projection-based IMC in update step.

however, be avoided by joint encoding of forward and backward motion vectors, which is similar to the "direct mode" employed in recent video coding standards.

It should be emphasized that in principle the MC in the prediction and the IMC in the update step could be independent. This still would guarantee perfect reconstruction by the inherent properties of the lifting structure, as was shown in [20]. However, a mismatch between MC and IMC would effect a smoothing of low-pass frames and eventually deteriorate compression performance.

In general, 3-D wavelet schemes will take more advantage by *true motion* estimation than hybrid coders. 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 3-D 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 [16]. Further, criteria can be applied which prefer motion vector fields that are spatially and temporally consistent over the levels of the wavelet pyramid [25]. 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 not yet resolved in all of its aspects.

The compression performance of scalable MCTF-based wavelet video coders was found to be very close to high-performance single-layer standard coders for many types of sequences [43]. Typically, the performance is often found to be better than for MC prediction methods in cases of relatively slow moving background. Typically, even though H frames appear to be conceptually similar with prediction frames of hybrid coding, they need to be encoded with less accuracy, because the error energy is spread over several frames during MCTF synthesis (for an in-depth analysis of this property, see [16] and [22]). On the other hand, the frames at the higher levels of the wavelet tree require more accurate encoding. This is one of the main reasons by which 3-D transform schemes have potential to become superior in performance compared to hybrid (prediction based) coders. A theoretical analysis of this gain has been given in [28].

In the case of scalable MCTF-based coding, it is, however, often found that the performance of this type of codecs deteriorates toward lower rates. This is mainly due to the fact that motion vectors as optimized for higher rates can then consume a big percentage of the overall rate, unless the motion information is encoded in a scalable fashion as well. The first promising approaches for scalable motion encoding are scaling of accuracy (e.g., from quarter-pel precision to half-

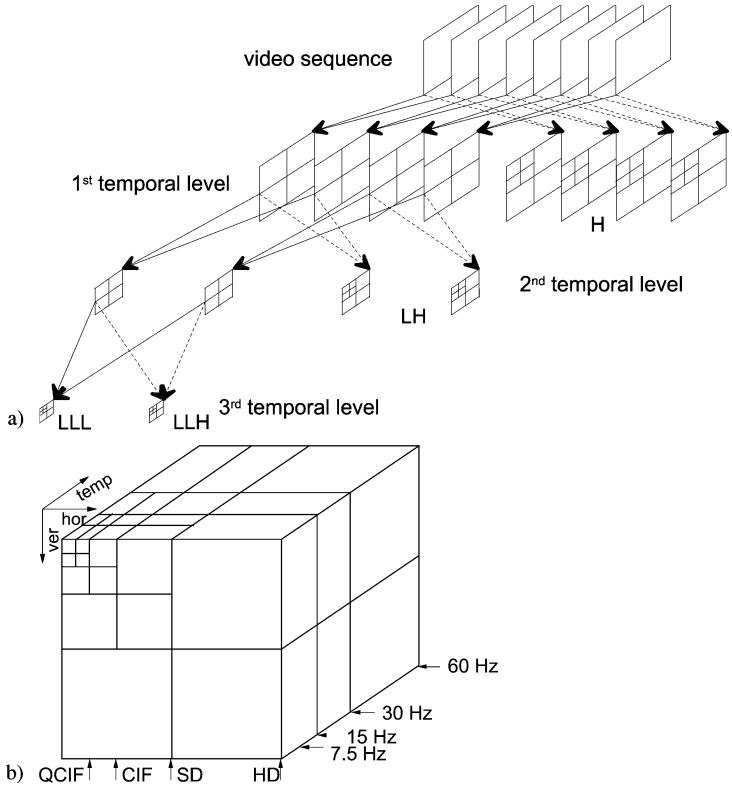

Fig. 16. (a) Wavelet tree with reduction of spatial size throughout the temporal levels. (b) Corresponding wavelet cube.

pel), of the spatial resolution (in terms of length of vectors depending on image size, and block size of motion vectors), and temporal resolution of motion vector fields. To minimize effects of different motion vectors usage during MCTF analysis and synthesis, scalable motion encoding should inherently tied with the variation of video signal resolution.

As a by-product, noise, sampling inconsistencies, etc., are discarded by the MCTF process. From this point of view, motion-compensated wavelet coding can realize advantages of joint multiple-frame 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 3-D wavelet coefficients as developed until now are not much different yet from 2-D wavelet coding or 3-D wavelet coding without MC. Embedded quantizers are used, which can straightforwardly be applied without penalty, as the synthesis filter structure is still nonrecursive by principle. Conventional 2-D 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 $t + 2$D transform, corresponding to the scheme shown in Fig. 9.

The optimum strategy of spatiotemporal 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. 16(a), where the spatial size of the frames is reduced by a factor of four after each temporal decomposition step (by one level of spatial 2-D 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. 16(b). The best spatiotemporal decomposition structure could be found by wavelet-packet design criteria, where the next split in the 3-D 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. 16(b) would allow spatial scalability between sub-Quarter Common Intermediate Format (QCIF) and high definition (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.

Alternatively, a spatial/temporal decomposition can be realized where the spatial wavelet filtering is done as the first step, and the temporal filtering is applied to the coefficients of the spatial transform (2-D+t transform). Within the lifting filters which are used for the temporal transform, it is then possible to use an *overcomplete discrete wavelet transform* (ODWT) which can generate any subsampling phase which

matches with the motion trajectory locally [29]. In terms of coding efficiency, this method is reported to be not inferior compared to the t+2-D transform, but has a number of advantages.

1) Spatial scalability of motion vector fields, combined with consistent hierarchical motion estimation can be applied. This is particularly important when a fully-scalable representation over several spatial transform levels shall be realized. In the case of the t+2-D transform, even though spatial scaling can be applied, no consistent way exists to decode the lower resolution frames by using a lower resolution motion vector field as well.
2) Even though spatial scalability can be applied to the t+2-D transform, methods to explicitly optimize for shift invariance of the 2-D transform are not straightforward. In fact, if the subsampled resolutions are generated by conventional down-sampling in the spatial wavelet pyramids of L and H frames, the inverse motion compensation may implicitly mix different subsampling phases; this can be a cause of artifacts in the vicinity of motion boundaries, but also may lead to temporally varying alias effects in the reconstructed signal.
3) As discussed before, spatially varying properties of filters, including partial violations of the constant norm principles at unconnected and multiple-connected positions, are inherent to MCTF. In order to keep these effects under control by appropriate quantization, it is advantageous to introduce them only by the last step of the 3-D wavelet decomposition. It is then possible to adapt the quantization step sizes sample by sample, where the adaptation parameters can directly be determined directly from the motion vectors with minimum effort.

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 appeared recently; see, e.g., [24], [26]–[28], [30]–[42].

V. Conclusion

New perspectives in video compression turn out through recent advances in scalable video coding, with MCTF being the most promising development. The fully-open-loop property of MCTF provides high flexibility in bitstream scalability for different temporal, spatial, and quality resolutions and better error resilience than conventional (closed-loop prediction based) coders. This is closely tied to an expansion of video codec operations over a larger number of frames, which causes an additional delay and may not always be appropriate in cases of fast changes. Therefore, combination of MCTF with the new drift-controlled prediction strategies as introduced in Section III, which are somewhat between fully closed- and open-loop methods, is also a promising path.

When in fact based on MCTF, a coded video representation can provide inherent capabilities to distinguish relevant and irrelevant parts of the information. The MCTF generated low-pass 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 multiframe prediction methods. A denoising process which is often applied as a preprocessing step before conventional video compression is applied, appears as an implicit part of scalable MCTF-based coders.

The implementation by a motion-compensated lifting structure allows to employ almost any motion compensation method developed previously for MC prediction coders. On the other hand, when low delay is required by certain applications, the update step must be omitted, or the number of temporal wavelet decomposition levels must be kept low, as these are the main causes for the increased coding delay. This would, however, decrease the coding performance of MCTF based coders, which again could partially be avoided by a combination with the methods for drift-free or controlled-drift MC prediction coders presented in Section III.

Due to the open-loop structure of MCTF, 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. MCTF is generally suitable to be combined with block-based 2-D transforms as well as 2-D wavelet transforms, and it may indeed depend on the concrete MC approach (e.g., block-based or flow-field based) to define the best combined solution. When, however, a 2-D wavelet transform is applied for encoding of the low-pass and high-pass frames resulting from the MCTF process, the commonalities with 2-D wavelet coding methods are also becoming obvious. If the sequence of spatial and temporal filtering is exchanged (2-D+t instead of t+2-D wavelet transform), MCTF could also be interpreted as a framework for further interframe compression of (intraframe encoded) 2-D wavelet representations such as JPEG2000. From this point of view, a bridge between the previously separate worlds of 2-D 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. One important aspect is the capability for seamless transition between intraframe and interframe coding methods, which is much more flexible when compared to the hard-decision intra/inter mode switching

that is traditionally implemented. In this context, the best mode decision not only relies on optimization of compression performance, but also on application requirements for flexible random access 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.

MPEG's recent Call for Proposals for new highly-efficient scalable video coding technology [43] and the current plans to develop such a scalable video framework 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.

REFERENCES

[1] "Video codec for audiovisual services at $p \times 64$ kbit/s," Int. Telecommun. Union-Telecommun. (ITU-T), Geneva, Switzerland, Recommendation H.261.

[2] "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s, Part 2: Video," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 11 172-2.

[3] "Generic coding of moving pictures and associated audio information—Part 2: Video," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 13 818-2 (identical to ITU-T Recommendation H.262.

[4] S. Okubo, "Requirements for high quality video coding standards," *Signal Process. Image Commun.*, vol. 4, pp. 141–151, 1992.

[5] "Coding of audiovisual objects—Part 2: Visual," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), 14 496-2.

[6] "Video coding for low bit rate communication," Int. Telecommun. Union-Telecommun. (ITU-T), Geneva, Switzerland, Recommendation H.263.

[7] "Coding of audiovisual objects—Part 10: Advanced video coding," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 14 496-10 (identical to ITU-T Recommendation H.264).

[8] C. Mayer, H. Crysandt, and J.-R. Ohm, "Bit plane quantization for scalable video coding," *Proc. SPIE, Visual Commun. Image Process.*, vol. 4671, pp. 1142–1152, 2002.

[9] P. Amon, K. Illgner, and J. Pandel, "SNR scalable layered video coding," presented at the Int. Packet Video Workshop, Pittsburgh, PA, 2002. [Online] Available: http://amp.ece.cmu.edu/packetvideo2002/papers/59–ethpsnsons.pdf.

[10] F. Wu, S. Li, and Y.-Q. Zhang, "DCT-prediction based progressive fine granularity scalable coding," in *Proc. IEEE Int. Conf. Image Processing*, 2000, pp. 1903–1906.

[11] M. van der Schaar and H. Radha, "A hybrid temporal-SNR fine-granular scalability for internet video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 318–331, Mar. 2001.

[12] "JPEG 2000 image coding system," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC 15 444.

[13] G. Karlsson and M. Vetterli, "Subband coding of video signals for packet switched networks," *Proc. SPIE, Visual Commun. Image Process.*, vol. 845, pp. 446–456, 1987.

[14] T. Kronander, "Some aspects of perception based image coding," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1989.

[15] J.-R. Ohm, "A hybrid image coding scheme for ATM networks based on SBC-VQ and tree encoding," in *Proc. 4th Int. Workshop Packet Video*, 1991, pp. B.2-1–B.2-6.

[16] ——, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Processing*, vol. 3, pp. 559–571, Sept. 1994.

[17] W. Sweldens, "The lifting scheme: A new philosophy in biorthogonal wavelet constructions," *Proc. SPIE*, vol. 2569, pp. 68–79, 1995.

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

[19] L. Luo, J. Li, S. Li, Z. Zhuang, and Y.-Q. Zhang, "Motion compensated lifting wavelet and its application in video coding," in *IEEE Int. Conf. Multimedia and Expo (ICME 2001)*, pp. 365–368.

[20] A. Secker and D. Taubman, "Motion-compensated highly-scalable video compression using an adaptive 3D wavelet transform based on lifting," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 2001, pp. 1029–1032.

[21] J.-R. Ohm and K. Rümmler, "Variable-raster multiresolution video processing with motion compensation techniques," in *Proc. IEEE Int. Conf. Image Processing*, vol. 1, 1997, pp. 759–762.

[22] J.-R. Ohm, *Multimedia Communication Technology*. New York: Springer-Verlag, 2004.

[23] S.-J. Choi and J. W. Woods, "Motion-compensated 3-D subband coding of video," *IEEE Trans. Image Processing*, vol. 8, pp. 155–167, Feb. 1999.

[24] K. Hanke, T. Rusert, and J.-R. Ohm, "Motion-compensated 3D video coding using smooth transitions," *Proc. SPIE Visual Commun. Image Process.*, vol. 5022, pp. 933–940, 2003.

[25] J.-R. Ohm, "Motion-compensated 3-D subband coding with multiresolution representation of motion parameters," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 1994, pp. 250–254.

[26] A. Golwelkar and J. W. Woods, "Scalable video compression using longer motion compensated temporal filters," *Proc. SPIE Visual Commun. Image Process.*, vol. 5150, pp. 1406–1417, 2003.

[27] T. Rusert, K. Hanke, and J.-R. Ohm, "Transition filtering and optimized quantization in interframe wavelet video coding," *Proc. SPIE Visual Commun. Image Process. 2003*, vol. 5150, pp. 682–694.

[28] M. Flierl and B. Girod, "Investigation of motion-compensated lifted wavelet transforms," in *Proc. Picture Coding Symp.*, 2003, pp. 59–62.

[29] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, P. Schelkens, and J. Cornelis, "Fully-scalable wavelet video coding using in-band motion-compensated temporal filtering," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2003, pp. III-417–III-420.

[30] Y. Andreopoulos, M. van der Schaar, A. Munteanu, J. Barbarien, and P. Schelkens, "Complete-to-overcomplete discrete wavelet transforms for fully-scalable video coding with MCTF," in *Proc. SPIE Visual Commun. Image Process.*, vol. 5150, 2003, pp. 719–731.

[31] C. Tillier, B. Pesquet-Popescu, Y. Zhan, and H. Heijmans, "Scalable video compression with temporal lifting using $5/3$ filters," in *Proc. Picture Coding Symp.*, 2003, pp. 55–58.

[32] M. van der Schaar and D. Turaga, "Unconstrained motion compensated temporal filtering (UMCTF) framework for wavelet video coding," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2003, pp. III-81–III-84.

[33] V. Valentin, M. Cagnazzo, M. Antonini, and M. Barlaud, "Scalable context-based motion vector coding for video compression," in *Proc. Picture Coding Symp. 2003*, 2003, pp. 63–70.

[34] D. Turaga, M. van der Schaar, and B. Pesquet, "Differential motion vector coding for scalable coding," *Proc. SPIE, Image Video Commun. Process.*, vol. 5022, pp. 87–97, 2003.

[35] J. Barbarien, Y. Andreopoulos, A. Munteanu, P. Schelkens, and J. Cornelis, "Coding of motion vectors produced by wavelet-domain motion estimation," in *Proc. Picture Coding Symp.*, 2003, pp. 193–197.

[36] G. Boisson, E. Francois, D. Thoreau, and C. Guillemot, "Motion-compensated spatio-temporal context-based arithmetic coding for full scalable video compression," in *Proc. Picture Coding Symp.*, 2003, pp. 71–76.

[37] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, "Memory-constrained 3-D wavelet transform for video coding without boundary effects," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 812–818, Sept. 2002.

[38] L. Luo, F. Wu, S. Li, and Z. Zhuang, "Advanced lifting-based motion-threading technique for the 3D wavelet video coding," in *Proc. SPIE, Visual Commun. Image Process.*, vol. 5150, 2003, pp. 707–718.

[39] N. Mehrseresht and D. Taubman, "Adaptively weighted update steps in motion compensated lifting based on scalable video compression," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 2003, pp. 771–774.

[40] D. Taubman and A. Secker, "Highly scalable video compression with scalable motion coding," in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, 2003, pp. 273–276.

[41] R. Leung and D. Taubman, "Context modeling and accessibility for 3D scalable compression," in *Proc. IEEE Int. Conf. Image Processing*, vol. 2, 2003, pp. 65–68.

[42] "Subjective test results for the CfP on scalable video coding technology," Int. Standards Org./Int. Electrotech. Comm. (ISO/IEC), ISO/IEC JTC1/SC29/WG11 Document N6383, 2004.

Jens-Rainer Ohm (Member, IEEE) received the Dipl.-Ing., Dr.-Ing., and Habilitation degrees from the Technical University of Berlin (TUB), Berlin, Germany, in 1985, 1990, and 1997, respectively.

From 1985 to 1990, he was a Research and Teaching Assistant with the Institute for Telecommunications, TUB. From 1990 to 1995, he performed work within government-funded research projects on image and video coding at the same location. From 1992 to 2000, he also served as Lecturer on the topics of digital image processing, coding, and transmission at TUB. From 1996 to 2000, he was Project Manager/Coordinator in the Image Processing Department, Heinrich Hertz Institute (HHI), Berlin. He was involved in research projects on motion-compensated, stereoscopic, and three-dimensional (3-D) image processing, image/video coding, and content description for image/video database retrieval. Since 1998, he has participated in the work of the Motion Pictures Experts Group (MPEG), where he has been active in the development of MPEG-4 and MPEG-7 standards. Currently, he chairs the Video Subgroup of MPEG. Since 2000, he has been Chair of the Institute of Communication Engineering at the Aachen University of Technology (RWTH), Aachen, Germany. He has authored textbooks on multimedia signal processing, analysis, and coding, on communications engineering, and on signal transmission, as well as numerous papers in the various fields mentioned above. His current research and teaching activities are in the areas of multimedia communication, multimedia signal processing/coding, and services for mobile networks, with emphasis on video signals, also including fundamentals of digital communication systems.