Complexity Scalable Motion Compensated Wavelet Video Encoding

Deepak S. Turaga, Mihaela van der Schaar, *Senior Member, IEEE*, and Beatrice Pesquet-Popescu

***Abstract*—We present a framework for the systematic analysis of video encoding complexity, measured in terms of the number of motion estimation (ME) computations, that we illustrate on motion compensated wavelet video coding schemes. We demonstrate the graceful complexity scalability of these schemes through the modification of the spatiotemporal decomposition structure and the ME parameters, and the use of spatiotemporal prediction. We generate a wide range of rate-distortion-complexity (R-D-C) operating points for different sequences, by modifying these options. Using our analytical framework we derive closed form expressions for the number of ME computations for these different coding modes and show that they accurately capture the computational complexity independent of the underlying content characteristics. Our framework for complexity analysis can be combined with rate-distortion modeling to determine the encoding structure and parameters for optimal R-D-C tradeoffs.**

***Index Terms*—Complexity analysis, rate-distortion-complexity (R-D-C) scalability, spatiotemporal motion vector prediction, wavelet video coding.**

I. Introduction

With the recent deployment of several multimedia mobile devices that have constrained computational resources as well as reconfigurable processing capabilities, it is getting especially important to achieve complexity scalability for multimedia coding. In such systems, the computational capabilities of the devices vary not just in terms of the power of the underlying processing, but also at run-time, for instance with the depletion of battery resources. This requires that the complexity of the multimedia algorithms be scaled gracefully to meet these constraints. Current video coding standards identify different profiles corresponding to varying computational complexity, however these profiles correspond specifically to the decoding complexity, and are fairly coarse in terms of their supported scalability. In this paper, we focus specifically on encoding complexity and show how different encoding parameters and configurations can be selected to scale the complexity gracefully, while optimizing rate-distortion (R-D) tradeoffs. This problem of scaling complexity gracefully has not been examined sufficiently in the past, and most current solutions have either a high quality high complexity codec or a low complexity low quality coding codec.

We present a framework to model and analyze video encoder complexity, especially in terms of the number of computations incurred during the motion estimation (ME) process.[1] We develop closed-form expressions for the number of computations required by ME under different coding modes and structures. With our analysis, the complexity associated with these different coding options can be predicted accurately independent of the content characteristics and, therefore, it can be used to determine the best coding options for optimized rate-distortion-complexity (R-D-C) tradeoffs.

While the complexity can be predicted across sequences with different content characteristics, the corresponding impact of scaling complexity on the decoded video quality can vary significantly. In this paper we further evaluate the impact of these different encoder configurations (with different complexities) on the decoded video quality for a variety of video sequences with widely differing content characteristics. Our complexity analysis can be combined with models for the decoded video quality (specific to different types of content) to determine the optimal R-D-C tradeoffs at run-time.

In this paper, we specifically consider motion-compensated temporal filtering (MCTF) based wavelet video coding schemes as they provide significant flexibility in terms of the coding modes, parameters and spatiotemporal decomposition structures, however our analysis is not limited to these schemes, and can be extended to different coding schemes and systems. Motion compensated wavelet video coding schemes use multiresolution temporal and spatial decompositions to remove redundancy, and achieve spatiotemporal scalability. MCTF was introduced by Ohm [1] and later improved by Choi and Woods [2]. There has been a growing interest in motion-compensated (MC)-wavelet schemes due to their significantly improved R-D performance that is achieved while also providing spatiotemporal signal-to-noise ratio (SNR) scalability. Recently, results have been presented in [6] showing that the R-D performance of MC-wavelet schemes is comparable with the latest predictive coding standards like H.264, despite the MC-wavelet coder being embedded and the H.264 coder being optimized for each individual decoded bit rate.

Many extensions have been proposed to increase the coding efficiency and the visual quality of MCTF schemes. Among

Manuscript received May 19, 2003; revised January 20, 2005. This paper was recommended by Associate Editor A. Puri.

D. S. Turaga is with the Wireless Communications and Networking, Philips Research USA, Briarcliff Manor, NY 10510 USA. He is also with the IBM T.J. Watson Research Center, Hawthorne, NY 10532 USA (e-mail: turaga@us.ibm.com).

M. van der Schaar is with the Wireless Communications and Networking, Philips Research USA, Briarcliff Manor, NY 10510 USA. He is also with the Department of Electrical and Computer Engineering, University of California Davis, Davis, CA 95616-5294 USA (e-mail: mvanderschaar@ece.ucdavis.edu).

B. Pesquet-Popescu is with the Signal and Image Processing Department, Télécom Paris, 75634 Paris Cedex 13, France (e-mail: pesquet@tsi.enst.fr).

Digital Object Identifier 10.1109/TCSVT.2005.852399

[1] The ME process can be a significant component of the encoder complexity.

these are lifting-based MCTF schemes proposed by Pesquet-Popescu and Bottreau [3] and by Secker and Taubman [7], [10], as well as unconstrained MCTF (UMCTF) introduced by Turaga and van der Schaar [8]. The UMCTF framework allows flexible and efficient temporal filtering by combining the best features of motion compensation, used in predictive coding, with MCTF. UMCTF involves designing temporal filters appropriately to enable this flexibility, while improving coding efficiency.

There has been a significant amount of work on designing novel ME algorithms for reducing video encoding complexity, and several algorithms that use gradient-descent like approaches, stochastic optimization, hierarchical strategies, genetic optimization etc. have been developed. As opposed to this, we consider the scaling of ME complexity through the selection of the appropriate encoding structure (spatiotemporal decomposition), ME parameters and spatiotemporal prediction. In this paper we perform our analysis using the full search ME algorithm, to retain the optimality of the search, however our analysis can easily be extended to account for other search strategies to reduce encoding complexity further.

The ME complexity in UMCTF based MC-wavelet coding can be adapted by changing the spatiotemporal decomposition structure as follows.

- By selecting the UMCTF controlling parameters - the temporal decomposition structure, the GOF size, the temporal filter taps and lengths, etc.—the *number of MEs* that need to be performed for each block can be reduced.
- Using adaptive spatiotemporal decomposition, the *number of blocks* for which ME is performed can be decreased.
- Subsequently, given a certain spatiotemporal decomposition, the temporal correlations existing across the different temporal decomposition levels can be exploited to further reduce the ME complexity. For instance, by temporally predicting MV across temporal levels and using adaptive search ranges across the temporal levels, the ME complexity for each block, i.e., *number of mean absolute difference (MAD) comparisons*, can be decreased.

We examine the combination of these "macro" and "micro" complexity controls, to achieve the appropriate R-D-C tradeoffs. This paper is organized as follows. We first introduce the UMCTF notation and framework, and its specific features. We then derive expressions for the ME complexity under different coding parameters and decomposition structures in Sections III and IV. We present results verifying our analysis, and highlighting the variation in decoded video quality across different video sequences, in Section V and finally conclude in Section VI.

II. Unconstrained Motion Compensated Temporal Filtering (UMCTF)

A. Notation

We consider a group of frames (GOF) containing N frames that will be filtered together. The temporal multiresolution analysis is performed over $D \in \mathbf{N}$ decomposition levels (by convention, $D = 0$ corresponds to the original frames) and we denote by N_d the number of frames at level $d \in [0, D]$ in the approximation subband. Let A_i^d be the approximation frames[2] at level $d \in [0, D]$, where $0 \leq i \leq N_d - 1$.

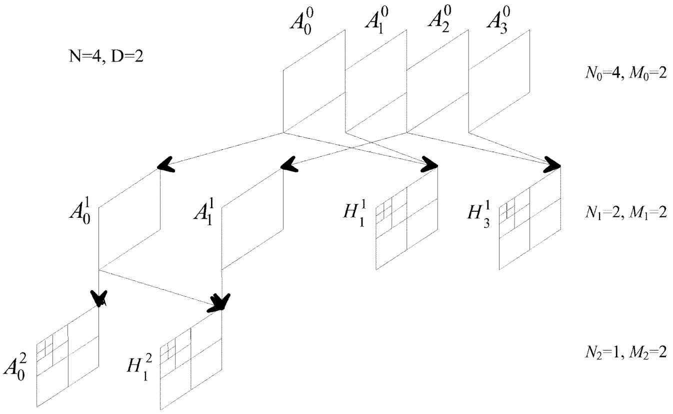


Fig. 1. Illustration of used UMCTF notation.

The subsampling factor can be different according to the resolution level, and we denote by M_d this decimation coefficient at level $d \in [0, D]$ (note that this also represents the gap between successive A frames at level d). We have, therefore, for $d > 0 : A_i^d = A_{M_d i}^{d-1}, i \in \{0, \ldots, N_d - 1\}$.

The motion vector (MV) connecting frames k and l at level $d \in [0, D]$ is denoted by $(v_{y,k\to l}^d, v_{x,k\to l}^d)$. The temporally high-pass filtered frames H_i^d at level $d \in [0, D]$ are obtained by motion-compensated filtering as follows:

$$H_i^d(n,m) = A_i^{d-1}(n,m) - \sum_{j \in S_i^d} f_j^d(n,m) A_{i-j}^{d-1}\left(n - \nu_{y,i-j\to i}^d, m - \nu_{x,i-j\to i}^d\right)$$

where index i belongs to $\{1, \ldots, M_d - 1, M_d + 1, \ldots, 2M_d - 1, 2M_d + 1, \ldots, N_{d-1} - 1\}$ (in other words, we skip frames with indexes multiple of M_d, which are the approximation frames); $A_{i-j}^{d-1}(n - \nu_{y,i-j\to i}^d, m - \nu_{x,i-j\to i}^d)$ represents the motion-compensated $(i - j)$th frame; this may possibly include a spatial interpolation, in case of a fractional-pel ME; f_j^d are the coefficients of the temporal high-pass filter to create H_i^d frames; and S_i^d is the support of the temporal filter, taking into account perfect reconstruction constraints and also the following conditions: $j \neq 0, i - N_{d-1} + 1 \leq j \leq i$. We denote by R_p^d (respectively R_f^d) the maximum number of reference frames allowed from the past (respectively, from the future). This notation is illustrated in Fig. 1.

B. UMCTF Framework

The peformance of UMCTF based coding schemes is determined by a set of "controlling parameters" as shown in Table I.

For instance by selecting the filter coefficients appropriately, we can introduce multiple reference frames and bidirectional prediction, like in H.264, in the MC-wavelet framework. We can adaptively change the number of reference frames, the relative

[2] In this paper, we do not consider any low-pass filtering, instead we pass the approximation frames unfiltered to be filtered later at future decomposition levels.

TABLE I
ADAPTATION PARAMETERS FOR UMCTF

Controlling Parameter	Adaptation Result
N	Changes GOF size
D	Limits the number of temporal decomposition levels
M^d	Enables flexible temporal scalability; allow different decodable frame rates
R_p^d	Varies the number of reference frames used from the past; can be different at different levels
R_f^d	Varies the number of reference frames used from the future; can be different at different levels
f_i^d	Changes the relative importance between reference and current frames, selects between available reference frames, can be different at different levels

importance attached to each reference frame, the extent of bidirectional filtering etc. Therefore, with this filter choice, the efficient compensation strategies of conventional predictive coding can be obtained by UMCTF, while preserving the advantages of conventional MCTF.

Similarly, we know that the coding gains through temporal filtering are likely to be smaller at higher decomposition levels, as the frames get farther apart. In such cases we may reduce the filter lengths. Sometimes, we might even stop the temporal decomposition beyond a certain level, i.e., reduce D. This decreases the overhead of transmitting different filter choices, MVs etc. that provide little or no gain. Simultaneously, the complexity is also reduced at these higher levels (lower frame rates).

C. UMCTF Complexity Analysis

In this section we derive the number of MEs performed per block of each frame for one GOF with multiple reference frames and bidirectional filtering.

For this analysis, we use N frames with positions $\{0, 1, \ldots, N-1\}$ in the GOF, and D decomposition levels. For the sake of simplicity, we set $M_d = M, R_p^d = N_{d-1} - 1$ and $R_f^d = 1$ for all levels d. More precisely, in high-pass temporal filtering (see Section II–A), we use from the past all possible frames before the current one and from the future the next approximation frame in the GOF. We assume $N \bmod (M)^D = 0$. Obviously, frame 0 does not use any other frame as reference. For all other frames we can derive the number of frames that they use as reference. This corresponds to the number of MEs that need to be performed for each block of the frame. In order to derive this number we consider frames that are encoded at different levels of the temporal decompositions. For instance, all high-pass filtered frames at level $d(d = 0, 1 \ldots D)$ were originally located at positions $k_{d,i}(M)^{d-1}$ with the factor $k_{d,i}$ such that $k_{d,i} \bmod M \neq 0$, and the index $i = 1, 2 \ldots ((N/(M)^{d-1}) - (N/(M))^d)$. In order to illustrate this, consider a decomposition with $N = 27, M = 3$ and $D = 3$. Frames that are encoded at level $d = 2$ come from original positions 3, 6, 12, 15, 21 and 24 corresponding to $k_{d,i} = 1, 2, 4, 5, 7, 8$ and $i = 1, 2, 3, 4, 5, 6$.

It can be determined that a frame in this position $k_{d,i}(M)^{d-1}$, uses $(k_{d,i} + 1)$ frames as reference. Indeed, $k_{d,i}$ is the number of frames before this frame (including the first frame) at the current level. Since we allow multiple reference frames from the past, all these are allowable reference frames. Besides this, we also have bidirectional filtering, so we need to add in one future frame. We also need to account for the case when we have no future frames for bidirectional filtering. This is true for the last few frames at each level, i.e., frames with $k_{d,i} > (N/(M))^{d-1} - M$. For each block in these frames, we can only perform $k_{d,i}$ MEs. We show two examples to illustrate this in Fig. 2.

In Fig. 2 we show two decompositions, one with $N = 9, M = 3, D = 2$ and one with $N = 8, M = 2, D = 3$. For both schemes, under each frame, in parentheses, we write the original positions of the frames that it may use as reference. We count these reference frames and include the total number enclosed in a circle, on the frame. This corresponds to the number of MEs that need to be performed for each block in the current frame.

Hence, if we assume that the number of blocks per frame is B the total number of MEs that need to be performed to encode a GOF of N frames using these UMCTF settings is

$$\#\mathrm{ME} = B \sum_{d=1}^{D} \left[\sum_{k_{d,i} < \frac{N}{(M)^{d-1}} - M} (k_{d,i} + 1) + \sum_{k_{d,i} > \frac{N}{(M)^{d-1}} - M} k_{d,i} \right]. \quad (1)$$

We show in Appendix A that this sum is equal to

$$\#\mathrm{ME} = B\zeta(D, N, M)$$

where

$$\xi(D, N, M) = \frac{N^2}{2} \frac{M}{M+1} (1 - M^{-2D}) + N(1 - M^{-D}) - D(M-1). \quad (2)$$

As an example, using different UMCTF parameters as shown in Fig. 2, we find $34B$ or $25B$ MEs (corresponding to the left and right UMCTF structures) per GOF.

The above result is derived assuming that we set $R_p^d = (N/(M^{d-1})) - 1$. In practice, we can limit the number of reference frames from the past to a smaller number R. In such a case each block in a frame at position $k_{d,i}(M)^{d-1}$ would require $\min(R+1, (k_{d,i}+1))$ MEs if $k_{d,i} \leq (N/((M)^{d-1})) - M$ and $\min(R, k_{d,i})$ otherwise.

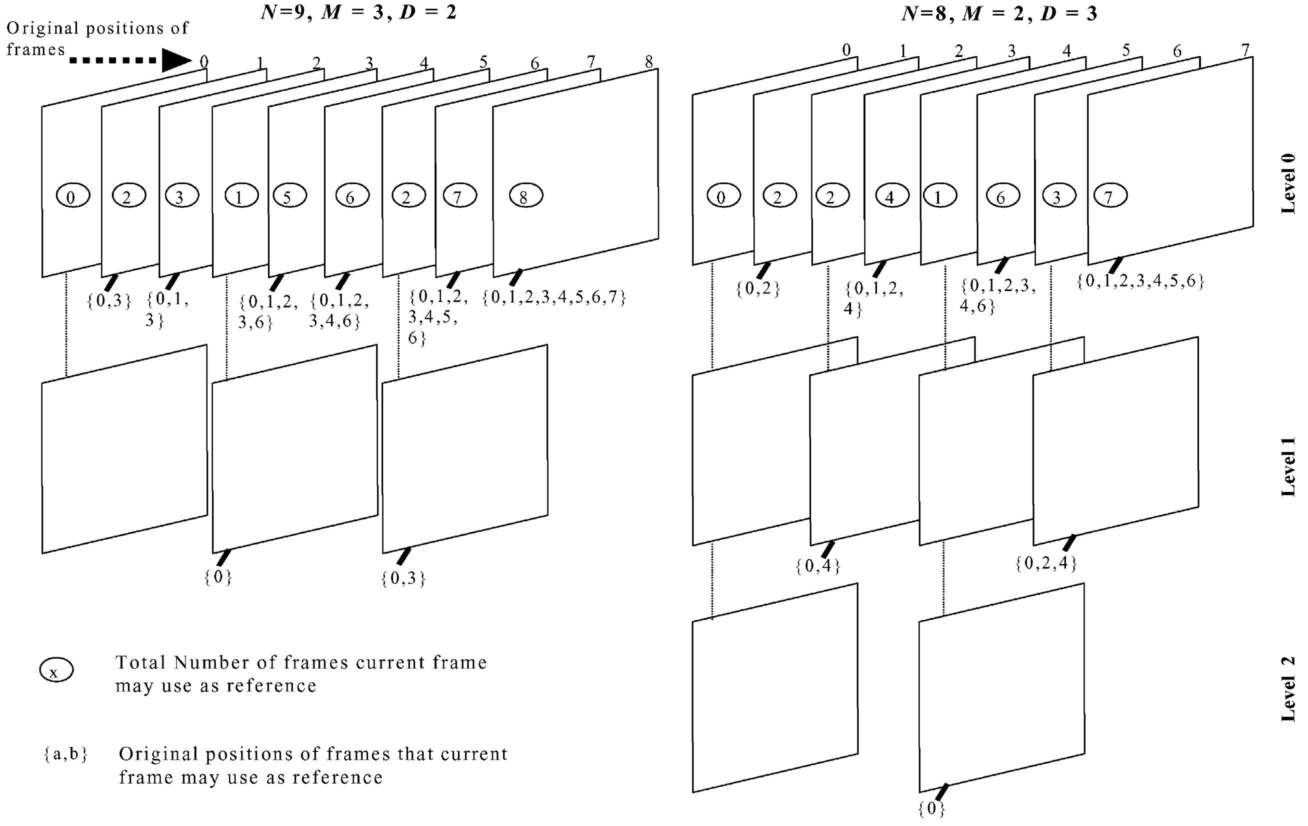


Fig. 2. Number of reference frames illustrated in two particular cases. (a) $N = 9, M = 3, D = 2$. (b) $N = 8, M = 2, D = 3$.

III. Spatiotemporal Decomposition Order

In conventional MC-wavelet schemes the spatial transform is applied after MCTF, thus it does not affect the ME complexity. Moreover, ME is typically performed on full resolution frames at all temporal levels, even though some of the spatial detail subbands are lost in the adaptation. In order to avoid this useless computational load, we can perform MCTF after performing some levels of the spatial transform, and thus we can significantly vary the ME complexity because this directly controls the *number of blocks* for which ME needs to be performed.

A. Example of Different Spatiotemporal Decomposition Order

To illustrate this tradeoff, we consider in Fig. 3, two schemes with different spatiotemporal decomposition order. Each of these schemes has $N = 4, D = 2$ and $M = 2$.

In the scheme on the left, two levels of temporal decomposition are performed first, followed by three levels of spatial decomposition. In this case ME is performed at the full spatial resolution for all the frames. Alternately, another spatiotemporal decomposition order is shown on the right. Here, while the first level of temporal decomposition is performed as on the left, the second level of the temporal decomposition is performed after one level of spatial decomposition. Hence, ME for frame 2 is performed at half the spatial resolution, while ME for frames 1 and 3 is performed at the full spatial resolution.

With a different spatiotemporal decomposition order, the ME is performed for a different number of blocks (if the block size for ME is fixed). For instance, if a frame consists of B blocks, then the scheme on the left requires ME for 3 frames with B blocks each, i.e., a total of $3 \times B$ blocks. In contrast, the scheme on the right requires ME for two frames with B blocks each and one frame with $B/4$ blocks (due to ME being performed at half the spatial resolution) i.e., a total of $9B/4$ blocks. Clearly, the decomposition scheme on the right requires a smaller number of ME and, hence, is less complex. We may thus adaptively select a different decomposition order to reduce the ME complexity.

We define D_h as the number of temporal levels that use half the spatial resolution and D_f as the number of temporal decomposition levels that use the full spatial resolution and $D_f + D_h = D$. Once we use half the spatial resolution at a temporal level, we have to use half (or smaller) the spatial resolution for all following coarser temporal levels.[3] In Fig. 3, the scheme on the left has $D_f = 2$ and $D_h = 0$, while the scheme on the right has $D_f = 1$ and $D_h = 1$.

B. Complexity Tradeoffs Using Adaptive Spatiotemporal Decomposition Order

When we use an adaptive spatiotemporal decomposition order, as discussed in the previous section, we no longer have the same number of blocks for each frame. Since frames at coarser temporal levels than level D_h are at half the spatial

[3] In general it is possible to perform multiple levels of spatial decomposition before MCTF, however in this paper we consider only the case with one level of spatial decomposition before MCTF. Equation (3) may be generalized to consider these different cases.

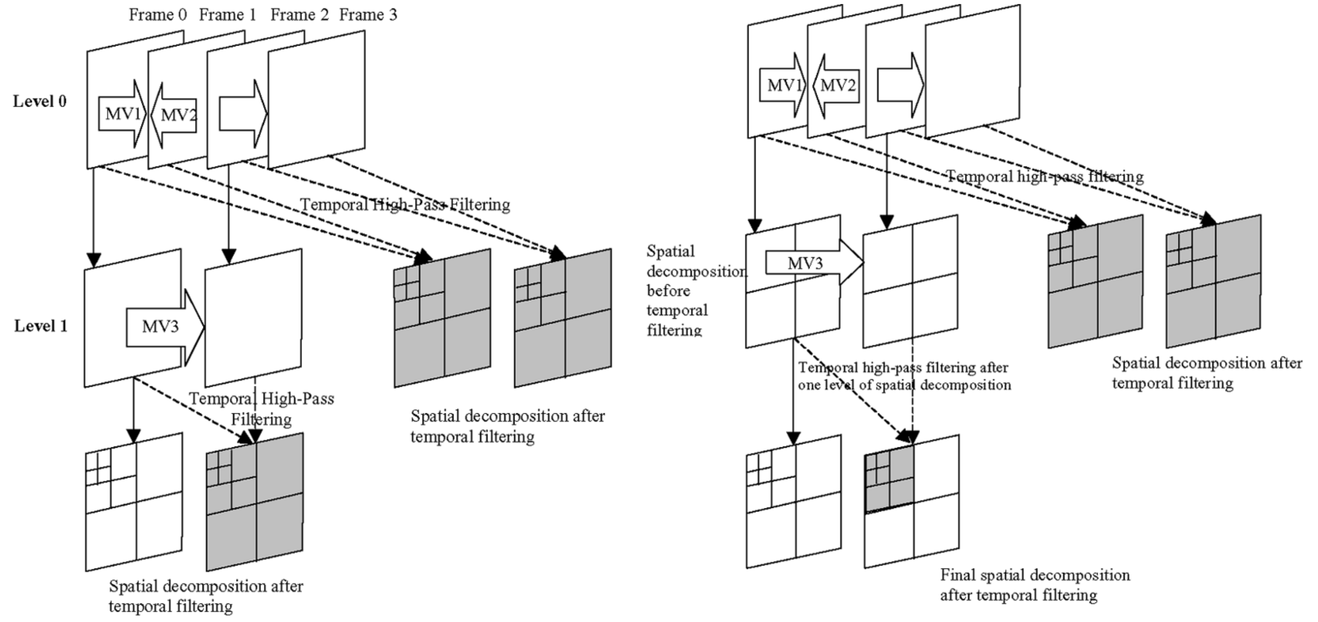

Frame 0
Frame 1
Frame 2
Frame 3
Level 0
MV1
MV2
Temporal High-Pass Filtering
Level 1
MV3
Temporal High-Pass Filtering
Spatial decomposition after temporal filtering
Spatial decomposition after temporal filtering
MV1
MV2
Temporal high-pass filtering
Spatial decomposition before temporal filtering
MV3
Temporal high-pass filtering after one level of spatial decomposition
Spatial decomposition after temporal filtering
Final spatial decomposition after temporal filtering

Fig. 3. Different spatiotemporal decomposition schemes.

resolution, while others are at full resolution we may modify (1), given D_h. Hence

$$\#\mathrm{ME} = B\sum_{d=1}^{D_f}\left(\sum_{k_{d,i}<\frac{N}{(M)^{d-1}}-M}(k_{d,i}+1) + \sum_{k_{d,i}>\frac{N}{(M)^{d-1}}-M} k_{d,i}\right) + \frac{B}{4}\sum_{d=D_f+1}^{D}\left(\sum_{k_{d,i}<\frac{N}{(M)^{d-1}}-M}(k_{d,i}+1) \times \sum_{k_{d,i}>\frac{N}{(M)^{d-1}}-M} k_{d,i}\right). \tag{3}$$

Proceeding similarly to the calculations in Appendix A, we find

$$\#\mathrm{ME} = \frac{3B}{4}\zeta(D_f,N,M) + \frac{B}{4}\zeta(D,N,M)$$

where the function ζ is defined by (2).

As $\zeta(D_f,N,M)$ decays when D_f decreases, if we increase D_h, the total number of ME blocks decreases, thereby reducing the ME complexity. The gain in complexity is given by

$$\frac{4}{1+3\frac{\zeta(D_f,N,M)}{\zeta(D,N,M)}}.$$

Note that this gain ranges from 1 to 4, but a gain up to 4^J could be similarly obtained by carrying out the spatial decomposition up to level J.

Considering the decomposition shown in Fig. 2, for the scheme with $N=8, D=3, M_d=2, R_p^d=N_{d-1}-1, R_f^d=1$ by setting D_h to 0, 1, 2 and 3, we need to perform $25B, 97B/4$ and $82B/4$ and $25B/4$ MEs per GOF. Thus, changing D_h can affect the ME complexity significantly.

IV. Prediction Across Temporal Decomposition Levels

Since the MC-wavelet coding schemes employ a multiresolution temporal decomposition, strong correlations exist between MVs at different temporal decomposition levels. By predicting MVs across different levels, we can decrease the search range and the number of MAD comparisons, thereby directly reducing ME complexity. We have introduced different prediction schemes in [14] and describe two of them in the following subsections. In order to simplify the description of these schemes, we show two levels of UMCTF decomposition along with the corresponding MVs in Fig. 4.

We label three MVs as MV1, MV2, and MV3 for ease of description.

A. Bottom-Up Prediction and Coding

In this scheme, we use MVs at temporal level $d+1$ to predict MVs at temporal level d and so on. Using our example in Fig. 4 this may be written as follows.

1) Estimate MV3.
2) Code MV3.
3) Predict MV1 and MV2 using MV3. Estimate refinement for MV1 and MV2 (or no refinement).

The prediction is used as the search center during estimation for MV1 and MV2. We show this in Fig. 5.

After prediction, we can vary the search range and reduce the complexity without sacrificing the quality of the match significantly. We include an analysis of the complexity with varying search ranges in Section IV-C.

The bottom-up prediction scheme produces temporally hierarchical MVs, like the temporally hierarchical decomposed

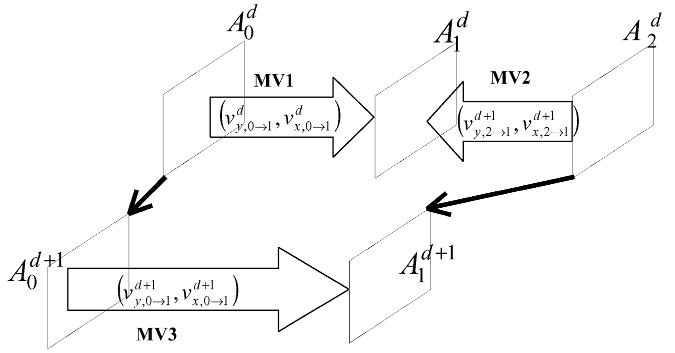

A_0^d
A_1^d
A_2^d
MV1
$(v^d_{y,0\to1}, v^d_{x,0\to1})$
MV2
$(v^{d+1}_{y,2\to1}, v^{d+1}_{x,2\to1})$
A_0^{d+1}
$(v^{d+1}_{y,0\to1}, v^{d+1}_{x,0\to1})$
MV3
A_1^{d+1}

Fig. 4. Two levels of temporal decomposition.

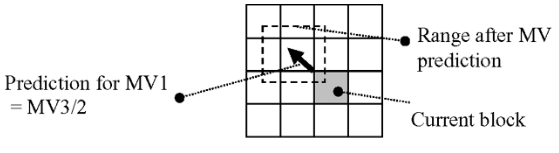


Fig. 5. MV prediction during estimation of MV3.

frames, that may be used progressively at different levels of the temporal decomposition scheme. So MV3 can be used to recompose Level 1 without having to decode MV2 and MV1. This is required to support temporal scalability. Also, this hierarchy of MVs may be coded using unequal error protection schemes to produce more robust bitstreams.

B. Top-Down Prediction and Coding

In this scheme, we use MVs at temporal level d to predict MVs at temporal level $d+1$, for $d \in \{0, \ldots, D-1\}$ and so on. In terms of simple procedural steps using our example in Fig. 4, this may be written as follows.

1) Estimate MV1 and MV2
2) Code MV1 and MV2.
3) Predict MV3 using MV1 and MV2. Code the refinement for MV3 (or no refinement).

One advantage of this scheme is the increased accuracy of MV predictions compared to the Bottom-up case. This is because MV1 and MV2 are likely to be accurately estimated (due to the small distance between the current and reference frames), thereby improving the prediction for MV3. The disadvantage is that all MVs need to be decoded before temporal reconstruction, even at low temporal resolutions. So MV1 and MV2 need to be decoded before MV3 can be decoded, and Level d can be recomposed. This is an unfavorable situation for temporal scalability.

Many other temporal prediction schemes may be defined for MV coding in the MCTF framework. Such schemes include hybrid prediction, i.e., using top-down prediction and bottom-up coding, or using MVs from multiple levels as predictors etc. One such hybrid prediction scheme has been introduced by Secker and Taubman [7].

C. Complexity Analysis of Variable Search Range Selection

As a measure of ME complexity we count the number of computations incurred during ME. We consider each addition to the MAD during ME, as one computation, i.e., we count

$$\mathrm{MAD} \leftarrow \mathrm{MAD} + \left|A_i^d\left(y - v^d_{y,i\to j}, x - v^d_{x,i\to j}\right) - A_j^d(y,x)\right|$$

as one computation. Frames i and j are the reference and current frames respectively, with $(v^d_{y,i\to j}, v^d_{x,i\to j})$ being the MV connecting blocks in the two frames. The number of MAD computations captures the effect of adaptive decomposition scheme choice as well as the effect of prediction and range selection. The global number of MAD operations required to estimate the MV $(v^d_{y,i\to j}, v^d_{x,i\to j})$ is directly proportional to the number of block comparisons performed during the search algorithm.

Let the search range after prediction be chosen as $\pm S$ in each direction, for a frame of size $Y_F \times X_F$ with blocks of size $Y_B \times X_B$ (we assume that $Y_F \bmod Y_B = 0$ and $X_F \bmod X_B = 0$). As we do not examine regions outside the frame boundaries, for a block (k,l) at the position[4] $(x_{k,l}, y_{k,l})$, the left-most position is $\max(0, x_{k,l} - S)$, therefore, we can search exactly $\min(S, x_{k,l})$ pixels to the left. Similarly, the right-most position of the block is $\min(X_F - 1, x_{k,l} + S + X_B - 1)$, and we can search exactly $\min(S, X_F - x_{k,l} - X_B)$ pixels to the right. Similar expressions may be written for the number of pixels below and pixels above. Hence, the total number of comparisons $C_{y_{i,j},x_{i,j}}$ for one block may be written as follows:

$$\begin{aligned}C_{y_{i,j},x_{i,j}}&(Y_F, X_F, Y_B, X_B, S)\\ &= [\min(S, X_F - x_{k,l} - X_B) + 1 + \min(S, x_{k,l})]\\ &\quad \times [\min(S, Y_F - y_{k,l} - Y_B) + 1 + \min(S, y_{k,l})] X_B Y_B.\end{aligned}$$

Hence, the number of comparisons for each frame, C_F (with B blocks) may be written as

$$\begin{aligned}C_F&(Y_F, X_F, Y_B, X_B, S)\\ &= \sum_{k=0}^{\frac{Y_F}{Y_B}-1} \sum_{l=0}^{\frac{X_F}{X_B}-1} C_{y_{k,l},x_{k,l}}(Y_F, X_F, Y_B, X_B, S).\end{aligned}$$

According to the prediction strategies across temporal level described in Sections IV-A and IV-B, a reduced search range S_d can be used, possibly different at each temporal resolution level, leading to an additional gain in complexity. Using the same block size at all spatial resolutions, we may combine the above expression with the results in Appendix A to obtain the total number of block comparisons for a GOF as follows:

$$\begin{aligned}\#\mathrm{comp.} = &\sum_{d=1}^{D_f} (M-1)\left(\frac{N^2}{2M^{2d-1}} + \frac{N}{M^d} - 1\right)\\ &\times C_F(Y_F, X_F, Y_B, X_B, S_d)\\ &+ \sum_{d=D_f+1}^{D} (M-1)\left(\frac{N^2}{2M^{2d-1}} + \frac{N}{M^d} - 1\right)\\ &\times C_F\left(\frac{Y_F}{2}, \frac{X_F}{2}, Y_B, X_B, S_d\right). \qquad (4)\end{aligned}$$

[4] The position may be the top-left point of the searched displaced block.

TABLE II
R-D-C RESULTS FOR DIFFERENT UMCTF PARAMETER SETTINGS

Sequence	UMCTF Ex 1					UMCTF Ex 2				
	Comp ($\times 10^8$)	MV Bits	PSNR (300)	PSNR (500)	PSNR (1000)	Comp ($\times 10^8$)	MV Bits	PSNR (300)	PSNR (500)	PSNR (1000)
Foreman	73.04	1403230	31.43	33.49	36.57	33.82	1091608	32.04	34.00	36.73
Coastguard	92.41	1158682	28.23	29.72	31.94	44.01	863874	27.71	29.25	31.62
Football	75.46	2037244	-	26.40	29.21	36.21	1433202	-	26.73	29.82
Mobile	68.94	1372456	-	25.18	27.89	32.64	1062448	-	24.85	27.37

For example, consider the MCTF parameters set to $N = 8, D = 3, M_d = 2, R_p^d = N_{d-1}-1, R_f^d = 1$, for a CIF (352 × 288) with $D_h = 0$. When applying a bottom-up approach where $S_2 = 64$ and $S_1 = S_0 = 4(X_B = Y_B = 16)$, by using (4) we can predict a reduction in complexity by a factor of about 22, compared against using a constant search range of 64.

V. EXPERIMENTAL SETUP AND RESULTS

In our experiments we use the Foreman, Coastguard, Football, and Mobile sequences at CIF (352 × 288) at 30 f/s. We use fixed block size ME with the block size selected as 16 × 16, and use full search at full pixel resolution to determine the MVs. We use the codec developed by Hsiang and Woods [15] for the entropy coding of the MC-wavelet data. As a measure of rate, we use the number of bits needed to code the MVs. As a measure of distortion, we use the average PSNR (in decibels) across the decoded video sequence. Finally, as a measure of complexity, we use the number of comparisons as defined in (4). This section is organized as follows. We first present results showing the effect of varying the UMCTF controlling parameter settings. We also present results in conjunction with adaptive spatiotemporal decomposition order. We then present some results with MV prediction and adaptive search range selection.

A. UMCTF Controlling Parameter Variation

We compare the ME complexity for different selections of the UMCTF parameters. In particular we consider two decomposition schemes. In both cases we perform five levels of spatial decomposition.

Example 1 (Ex 1): $N = 16, D = 4, M_d = 2, R_p^d = 1, R_f^d = 1$. We set $\sum_j f_j^d(n, m) = 1$, where either two coefficients are nonzero in the summation and they are equal to $1/2$ (bidirectional filtering), or only one is nonzero and it is equal to 1 (forward,/backward filtering).

Example 2 (Ex 2): $N = 9, D = 2, M_d = 3, R_p^d = 1, R_f^d = 1$. As in the previous example, a forward, backward or bidirectional filtering, using only previous and future A frame as reference is possible.

The decision between forward, backward or bidirectional filtering is made adaptively for every block based on which of these provides the smallest MAD. We use a search range of ± 64 during ME.[5] We present rate, distortion and complexity results in Table II. The results are reported at three different bit rates, 30, 500, and 1000 kb/s, and these are indicated on the columns.

The number of frames for which bidirectional ME is performed for Ex 2, is smaller than for Ex 1, leading to reduced ME complexity; smaller by a factor of ~ 2. This reduction in complexity can be predicted from (1). From the equation, we expect the reduction in complexity to be $13/6$, as observed experimentally. For sequences with temporally correlated motion, a larger N and D improve the R-D performance, however for sequences with temporally uncorrelated motion they actually degrade the R-D performance. This is observed in Table II, where Ex 2 results are ~ 0.5 dB better for the Foreman and Football sequences and ~ 0.5 dB worse for the Coastguard and Mobile sequences than Ex 1 results. Hence, by smart selection of UMCTF parameters we can actually reduce the complexity while improving the R-D performance. These results are consistent across the different decoded bit rates. Ex 2 requires fewer bits for MV coding, and these savings translate to improved performance at lower bit rates.

B. Adaptive Spatiotemporal Decomposition Order

We now present results that measure the effect of adaptive spatiotemporal decomposition order. For the UMCTF parameter setting Ex 1, in Section V-A, we select five decomposition orders, and examine their performance. These five schemes correspond to $D_h = 4$ (Levels 0, 1, 2 and 3), $D_h = 3$ (Levels 1, 2 and 3), $D_h = 2$ (Levels 2 and 3), $D_h = 1$ (Level 3), and $D_h = 0$. Similarly for UMCTF Ex 2 we consider three orders $D_h = 2$ (Levels 0 and 1), $D_h = 1$ (Level 1) and $D_h = 0$.

$D_h = 4$ for Ex 1 and $D_h = 2$ for Ex 2 correspond to using ME at half the spatial resolution for all temporal levels, while $D_h = 0$ corresponds to using ME at the full spatial resolution for all the temporal levels. We show the spatial resolution at each temporal decomposition level for these schemes in Fig. 6.

The results on complexity, rate and distortion are presented in Table III. The PSNR results are presented at 300 kb/s for Foreman and Coastguard and 1000 kb/s for Football and Mobile.

[5] In this range, the search center location is predicted from the MVs of the spatial neighbors of the block using MV prediction schemes similar to those part of MPEG4/H.263. Since the search center is not the same for all the frames, we observe a variation in the number of computations across sequences. However, the relative ratios between the number of computations for each sequence, for different parameter choices, can be predicted very accurately for all sequences.

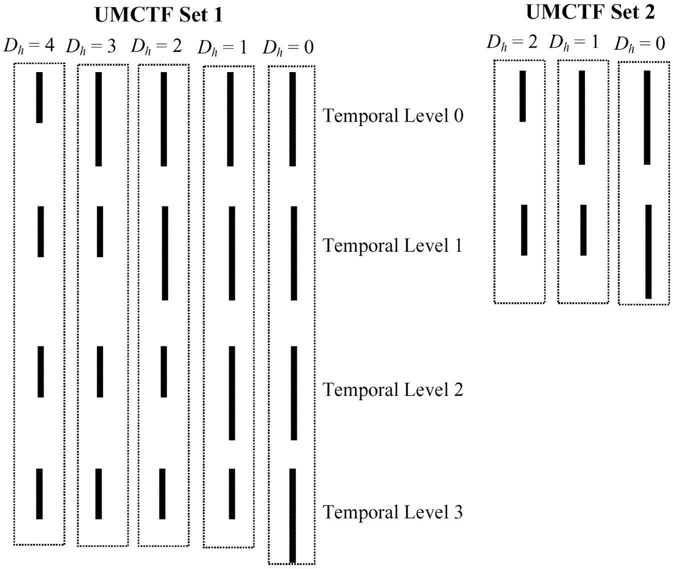


Fig. 6. Spatial resolution at different temporal levels.

TABLE III
R-D-C RESULTS WITH ADAPTIVE SPATIO-TEMPORAL DECOMPOSITION ORDER

Sequence	D_h	UMCTF Ex 1			UMCTF Ex 2		
		Comp. ($\times 10^8$)	MV Bits	PSNR (dB)	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)
Foreman (300 Kbps)	4	11.43	359738	29.43	-	-	-
	3	47.13	772758	30.29	-	-	-
	2	59.32	1004570	31.04	6.11	297728	30.34
	1	67.61	1178324	31.45	28.31	945864	31.29
	0	73.04	1403230	31.43	33.82	1091608	32.04
Coastguard (300 Kbps)	4	14.23	320166	26.33	-	-	-
	3	49.84	588302	27.23	-	-	-
	2	73.21	785421	28.07	6.74	254232	26.39
	1	84.58	1000312	28.28	36.27	701444	27.15
	0	92.41	1158682	28.23	44.01	863874	27.71
Football (1000 Kbps)	4	13.77	589340	26.26	-	-	-
	3	47.28	1193426	27.74	-	-	-
	2	65.91	1632084	28.27	7.93	387318	27.74
	1	71.34	1883426	28.73	30.92	1239326	29.01
	0	75.46	2037244	29.21	36.21	1433202	29.82
Mobile (1000 Kbps)	4	10.97	387464	23.92	-	-	-
	3	43.85	743928	25.16	-	-	-
	2	60.29	1143788	26.68	6.04	289846	24.09
	1	65.73	1278664	27.64	26.43	897462	26.26
	0	68.94	1372456	27.89	32.64	1062448	27.37

The first trend that we can observe is that as D_h increases, the PSNR drops. This is to be expected, because with increasing D_h the amount of temporal redundancy removed is smaller. However, comparing D_h equal to 1 and 0, for the Coastguard sequence, we see that sometimes PSNR increases with D_h. In this case, although the temporal filtering is less efficient at level 3 for $D_h = 1$, the savings in MV bits are sufficient to overcome this small difference. By comparing the different rows for UMCTF Ex 1, we observe that by changing the spatiotemporal decomposition order we can reduce complexity by a factor of ~6, however, this comes at almost 2-dB loss in quality. Again, this can be predicted by (4). From the equation when D_h varies between 4 and 0, the complexity reduces by a factor of 5.5, similar to what is observed. From the table we also observe that the complexity can be reduced by a factor of 1.5–2 while sacrificing less than 1 dB of quality. Interestingly, by comparing across Ex 1 and Ex 2 we see that by sacrificing ~1 dB of PSNR, we can reduce the complexity by a factor of ~11 for the Foreman sequence, and a factor of ~2.5 for the Coastguard, Mobile, and Football sequences.

C. Temporal MV Prediction During ME

We first illustrate the effects of using temporal MV prediction for estimation and coding on the ME complexity. For these results we use UMCTF parameters as Ex 1. We present results for both the bottom-up and the top-down strategies.

We consider three different search ranges after prediction. These are: $\text{Range} = 0, \text{Range} = 4$, and $\text{Range} = 64$. In the $\text{Range} = 0$ case, we perform ME only at temporal level 0 (for top-down) and at temporal level 3 (for bottom-up) and propagate those MVs across all the temporal levels, without refinement. When the range is nonzero, we refine the prediction within that search range. This prediction and adaptive range choice controls the number of computations for each block during ME. We first show results for the bottom-up strategy in Table IV.

When we use a search range 0, i.e., do not use a refinement, we can save up to 75% of the bits needed to code MVs over when we use a search range of 64. This saving in MV bits may be used to code the texture, however the lack of refinement of MVs leads to poor matches. Importantly, we can reduce the complexity by a factor of ~15 while sacrificing ~1 dB in PSNR.

We also present results for the top-down prediction strategy in Table V.

The degradation in quality for the top-down schemes is less significant than for the bottom-up strategies, however the ME complexity also continues to remain high. We can achieve a reduction in complexity by a factor of 1.8–2 while sacrificing less than 0.4 dB in PSNR. However, the top-down strategy does not support temporal scalability.

There are other interesting extensions that can be examined in the future, such as varying the search pyramidally across the different temporal levels to tradeoff the accuracy of prediction and the complexity.

D. Combined UMCTF Settings, Spatiotemporal Decomposition Order, and MV Prediction

In this section we present results to demonstrate the combined effect of selecting the UMCTF settings, the spatiotemporal decomposition order, and using adaptive search ranges in conjunction with MV prediction. Since we use a fixed block size for the ME and filtering, we need to extend the MV prediction scheme to consider cases when we predict across temporal levels at different spatial resolutions. This happens at most once per GOF, i.e., when we predict across the boundary levels. For example, when $D_h = 2$, this happens when we use MVs from Level 2 to predict MVs from Level 1. In such cases each MV from the coarser resolution is used to predict four forward and four backward MVs at the finer resolution. We use the Bottom-up strategy and consider three different search ranges after prediction: $\text{Range} = 0, \text{Range} = 4$, and $\text{Range} = 64$.

TABLE IV
R-D-C RESULTS FOR BOTTOM-UP PREDICTION OF MVs

Sequence	Range = 0			Range = 4			Range = 64		
	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)
Foreman (300 Kbps)	3.63	332656	29.35	4.62	996864	30.39	67.14	1245728	31.58
Coastguard (300 Kbps)	5.07	332360	26.37	5.99	955800	27.50	88.10	1139464	28.33
Football (1000 Kbps)	4.01	572826	27.05	6.37	1743024	28.19	73.51	1996008	29.17
Mobile (1000 Kbps)	3.98	297410	26.45	5.64	823306	27.85	65.43	1193204	27.98

TABLE V
R-D-C RESULTS FOR TOP-DOWN PREDICTION OF MVs

Sequence	Range = 0			Range = 4			Range = 64		
	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)	Comp. ($\times 10^8$)	MV Bits	PSNR (dB)
Foreman (300 Kbps)	35.45	661272	30.21	36.83	942952	31.21	64.58	1019880	31.55
Coastguard (300 Kbps)	42.85	464544	27.71	43.84	663592	28.16	86.79	764000	28.34
Football (1000 Kbps)	32.86	1242692	28.12	39.27	1641726	28.54	75.46	1847244	29.31
Mobile (1000 Kbps)	35.93	781144	27.30	37.84	1028002	27.87	63.81	1162592	27.95

TABLE VI
COMBINED R-D-C RESULTS FOR UMCTF EX 1

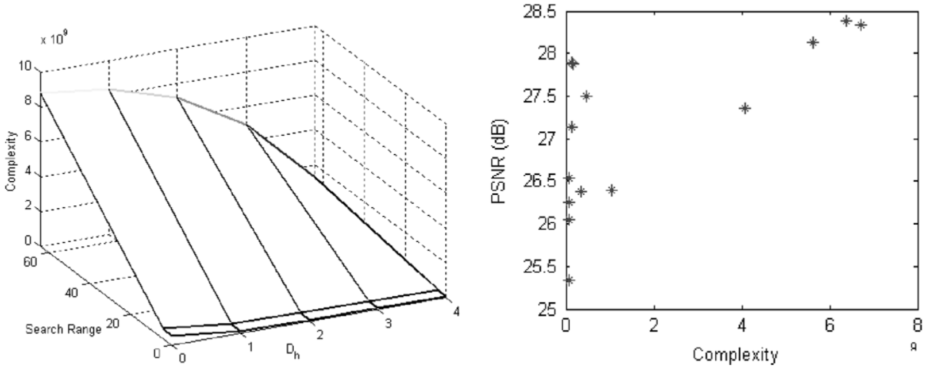
Sequence	D_h	Range = 0			Range = 4			Range = 64		
		Comp. ($\times 10^8$)	Bits	PSNR (dB)	Comp. ($\times 10^8$)	Bits	PSNR (dB)	Comp. ($\times 10^8$)	Bits	PSNR (dB)
Foreman (300 Kbps)	4	0.61	180728	27.69	0.80	315344	28.98	10.51	344648	29.57
	3	0.62	227584	28.67	1.25	626784	29.81	40.83	732864	30.40
	2	0.63	258688	29.27	1.48	824416	30.34	56.27	996768	31.06
	1	0.63	274240	29.46	1.59	918528	30.58	63.92	1149216	31.48
	0	3.63	332656	29.35	4.62	996864	30.39	67.14	1245728	31.58
Coastguard (300 Kbps)	4	0.79	179712	25.35	0.96	291680	26.25	13.13	306272	26.40
	3	0.80	226584	26.06	1.31	550784	27.14	49.76	579120	27.36
	2	0.81	257688	26.54	1.53	679816	27.90	72.00	754592	28.12
	1	0.81	273240	26.54	1.66	809208	27.87	83.63	991768	28.38
	0	5.07	332360	26.37	5.99	955800	27.50	88.10	1139464	28.33
Football (1000 Kbps)	4	0.69	257274	24.71	0.99	513502	25.43	13.77	543724	26.09
	3	0.70	349112	25.61	1.46	1032088	26.54	47.28	1137952	27.41
	2	0.70	427054	26.09	1.58	1303004	27.20	62.83	1548096	28.14
	1	0.71	513002	26.62	1.73	1596504	27.84	70.04	1820872	28.68
	0	4.01	572826	27.05	6.37	1743024	28.19	73.51	1996008	29.17
Mobile (1000 Kbps)	4	0.72	174374	22.97	0.95	329086	24.25	8.78	357464	23.96
	3	0.72	216992	24.18	1.34	609072	25.55	41.02	673928	25.75
	2	0.72	241852	25.20	1.51	717512	26.44	57.09	973822	26.85
	1	0.72	274760	26.05	1.59	782092	27.10	64.38	1083492	27.70
	0	3.98	297410	26.45	5.64	823306	27.85	65.43	1193204	27.98

We first present complexity-R-D results for UMCTF Ex 1 results for the sequences in Table VI.

By comparing different columns for each row we observe that we can obtain savings in complexity of factors of typically 50–100 times for these different sequences. As before, we can predict this using (4). Such a reduction in ME complexity is significant, especially for devices with limited computational power, however it comes at a cost of ~2 dB in PSNR.

This combination of spatiotemporal decomposition order and adaptive search ranges provides us a large and flexible set of operating points to choose from, based on the decoder requirements and constraints. For instance, by comparing across these different rows and columns we can see that by sacrificing 0.6–1 dB[6] (over the best achieved PSNR) in quality, we can achieve significant savings in complexity, between 40–50 times. We show the complexity versus D_h and search range surface and also the distortion-complexity points for our results for the Coastguard sequence in Fig. 7.

[6] The precise drop in PSNR is sequence dependent and is typically larger for videos with rapid motion, and lower for sequences with limited motion.



(a)

(b)

Fig. 7. (a) Complexity surface and (b) distortion-complexity points for Coastguard.

TABLE VII
PSNR (decibels) AND COMPLEXITY WITH ME TURNED OFF

Sequence	D_h	PSNR (dB)	Complexity ($\times 10^6$)
Foreman (300 Kbps)	4	25.09	0.66
	3	25.77	1.79
	2	26.60	2.33
	1	27.29	2.56
	0	27.74	2.64
Coastguard (300 Kbps)	4	23.59	0.66
	3	24.16	1.79
	2	24.69	2.33
	1	25.01	2.56
	0	25.12	2.64
Football (1000 Kbps)	4	24.16	0.66
	3	25.08	1.79
	2	25.72	2.33
	1	26.10	2.56
	0	26.16	2.64
Mobile (1000 Kbps)	4	21.63	0.66
	3	21.77	1.79
	2	22.45	2.33
	1	22.47	2.56
	0	22.49	2.64

By optimizing the distortion-computation cost the desired decomposition scheme with the appropriate search range can be selected.

Similarly, we can show that for UMCTF Ex 2 we can still achieve reductions by factors of 20–30 with a corresponding the drop in PSNR of 0.6–0.8 dB. The complexity reduction obtained by adaptive search range selection is smaller in this case than for UMCTF Ex 1 due to the smaller number of temporal decomposition levels.

E. Results With ME Off

As a lower bound on ME complexity, we consider the ME turned off (all MVs set to zero) scenario. The only ME complexity comes from deciding between forward, backward and bidirectional MV selection (since one MAD needs to be computed per block for each of these cases). We present results with the UMCTF parameters chosen as in Ex 1, and with an adaptive spatiotemporal decomposition order in Table VII.

This scheme has very low complexity, however by comparing these columns against those in Table VI we see that the corresponding PSNR drop is very large (2.5–4 dB), depending on the sequence characteristics.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we present a comprehensive analysis of the encoder complexity in terms of the number of computations required during ME. We derive closed-form expressions for the complexity under different encoder coding modes, parameter selection and spatiotemporal decomposition structures. With our analysis, the complexity associated with these different coding options can be predicted accurately independent of the content characteristics and, therefore, it can be used to determine the best coding options for optimized R-D-C tradeoffs. We use this analysis to show how different encoding parameters and configurations can be selected to scale the complexity gracefully, while optimizing R-D performance.

We implement these different coding structures and options in a UMCTF based MC-wavelet encoding scheme. We adaptively select different UMCTF parameter settings, spatiotemporal decomposition order, and MV prediction with variable search ranges, to obtain a wide range of R-D-C operating points, and show how the complexity associated with these different options can be predicted using our analysis.

We have shown that we may obtain reductions in complexity by factors of up to 50, over the full complexity implementation, with penalties of less than 0.6–1 dB in PSNR, by adaptively changing the spatiotemporal decomposition order and structure, by using MV prediction across temporal resolutions, and by changing the search range after prediction. Furthermore, we also show that by changing the temporal decomposition structure, we can improve the PSNR by ~0.5 dB while reducing the ME complexity by a factor of ~2. Turning off ME can also result in very low complexity, however, that leads to a significant loss in PSNR (~2.5–4 dB), which is undesirable for most applications.

We present a large set of operating points corresponding to different distortion-complexity costs and a R-D-C cost can be optimized to select the appropriate decomposition structure and coding parameters. While the complexity can be accurately predicted, across different sequences, by our analysis, the R-D performance depends heavily on the sequence content characteristics. There is some work on developing R-D models for video under different coding schemes [17], [18], however these models need to be further refined to improve their accuracy. Our complexity prediction can be combined with such models to obtain complete R-D-C models and thereby used to select the optimal encoding parameters and structures on the fly. Some preliminary work on developing combined R-D-C models for video has been presented in [19]. Additional work on combining some of these R-D-C models with on-the-fly adaptation of reconfigurable processor and memory architectures for optimal power utilization in mobile devices has been presented in [20]. We are also investigating different optimization techniques for the pruning of the parameter space and the selection of the optimal operating points. Among these, we are considering standard Lagrangian optimization techniques, and classification based learning approaches [16]. Finally, we are extending our work to include other techniques to reduce complexity, such as using different ME strategies, adaptive subpixel accuracy for ME, other coding schemes etc.

Appendix A

The sum in (1) can also be written as

$$\sum_{d=1}^{D}\left(\sum_{k_{d,i}} k_{d,i} + \sum_{k_{d,i}>N/M^{d-1}-M} 1\right).$$

The first term in this expression is

$$\begin{aligned}\sum_{k_{d,i}} k_{d,i} &= \sum_{k=0}^{\frac{N}{M^{d-1}}-1} k - \sum_{l=0}^{\frac{N}{M^{d}}-1} lM \\ &= \frac{\left(\frac{N}{M^{d-1}}-1\right)\frac{N}{M^{d-1}}}{2} - M\frac{\left(\frac{N}{M^{d}}-1\right)\frac{N}{M^{d}}}{2}.\end{aligned}$$

After some straightforward calculations, this leads to

$$\sum_{k_{d,i}} k_{d,i} = \frac{N^2}{2M^{2d-1}}(M-1).$$

The second term of the sum is

$$\begin{aligned}\sum_{k_{d,i}>N/M^{d-1}-M} 1 &= \frac{N}{M^{d-1}} - \frac{N}{M^{d}} - (M-1) \\ &= (M-1)\left(\frac{N}{M^{d}}-1\right).\end{aligned}$$

By summing over d the above two terms, we get

$$\begin{aligned}&\sum_{d=1}^{D}(M-1)\left(\frac{N^2}{2M^{2d-1}} + \frac{N}{M^{d}} - 1\right) \\ &\quad = (M-1)\left[\frac{N^2}{2}\frac{M}{M^2-1}(1-M^{-2D}) \right.\\ &\qquad \left. + N\frac{1-M^{-D}}{M-1} - D\right]\end{aligned}$$

which finally leads to the expression of ζ in (2). As expected, $D \mapsto \zeta(D,N,M)$ is a strictly increasing function, as it is the sum of D positive terms.

References

[1] J. R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 559–589, Sep. 1994.

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

[3] B. Pesquet-Popescu and V. Bottreau, "Three-dimensional lifting schemes for motion compensated video compression," in *Proc. ICASSP*, Salt Lake City, UT, May 2001, pp. 1793–1796.

[4] Y. Zhan, M. Picard, B. Pesquet-Popescu, and H. Heijmans, "Long temporal filters in lifting schemes for scalable video coding," presented at the MPEG Contribution M8680, Klagenfurt, Germany, Jul. 2002.

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

[6] K. Hanke, "R-D performance of fully scalable MC-EZBC," presented at the MPEG Contribution M9000, Oct. 2002.

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

[8] C. Tillier and B. Pesquet-Popescu, "3D, 3-band, 3-tap temporal lifting for scalable video coding," in *Proc. IEEE ICIP*, Barcelona, Spain, Sep. 2003, pp. 779–782.

[9] M. van der Schaar and D. S. Turaga, "Unconstrained motion compensated temporal filtering framework for wavelet video coding," in *Proc. ICASSP*, May 2003, pp. 81–84.

[10] V. Bottreau, M. Bénetiére, B. Felts, and B. Pesquet-Popescu, "A fully scalable 3-D subband video codec," in *Proc. IEEE Int. Conf. Image Processing*, Thessaloniki, Greece, Oct. 2001, pp. 1017–1020.

[11] A. Secker and D. Taubman, "Highly scalable video compression using a lifting-based 3-D wavelet transform with deformable mesh motion compensation," in *Proc. ICIP*, Rochester, NY, Sep. 2002, pp. 749–752.

[12] D. S. Turaga, M. van der Schaar, and B. Pesquet-Popescu, "Differential motion vector coding in the MCTF framework," presented at the MPEG Contribution M9035, Shanghai, China, Oct. 2002.

[13] J. R. Ohm, "Complexity and delay analysis of MCTF interframe wavelet structures," presented at the MPEG Contribution M8520, Klagenfurt, Germany, Jul. 2002.

[14] D. S. Turaga, M. van der Schaar, and B. Pesquet-Popescu, "Differential motion vector coding for scalable coding," in *Proc. IVCP*, Jan. 2003, pp. 87–97.

[15] S.-T. Hsiang and J. W. Woods, "Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank," *Signal Process.: Image Comm.*, vol. 16, pp. 705–724, 2001.

[16] D. S. Turaga and T. Chen, "Classification based mode decisions for video over networks," *IEEE Trans. Multimedia, Special Issue on Multimedia over IP*, vol. 3, no. 1, pp. 41–52, Mar. 2001.

[17] M. Wang and M. van der Schaar, "Rate-distortion modeling for wavelet video coders," in *Proc. IEEE ICASSP*, 2005, pp. 53–56.

[18] T. Rusert, K. Hanke, and J. Ohm, "Transition filtering and optimization quantization in interframe wavelet video coding," in *Proc. SPIE VCIP*, vol. 5150, 2003, pp. 682–693.

[19] M. van der Schaar and Y. Andreopoulos, "Rate-distortion-complexity modeling for network and receiver aware adaptation," *IEEE Trans. Multimedia, Special Issue on MPEG-21*, vol. 7, no. 3, pp. 471–479, Jun. 2005.

[20] G. Landge, M. van der Schaar, and V. Akella, "Complexity metric driven energy optimization framework for implementing MPEG-21 scalable decoders," in *Proc. SPIE VCIP*, 2005, pp. 1141–1144.

Deepak S. Turaga received the B.Tech. degree in electrical engineering in 1997 from the Indian Institute of Technology, Bombay, and the M.S. and Ph.D. degrees in electrical and computer engineering in 1999 and 2001, respectively, from Carnegie Mellon University, Pittsburgh, PA.

He is currently a Research Staff Member in the Media Delivery Architectures Department, IBM T.J. Watson Research Center, Hawthorne, NY. His research interests lie primarily in multimedia coding and streaming, and computer vision applications. In these areas, he has published several journal and conference papers and one book chapter. He has also filed over fifteen invention disclosures, and has participated actively in MPEG standardization activities.

Dr. Turaga is an Associate Editor of the IEEE Transactions on Multimedia.

Mihaela van der Schaar (SM'04) received the M.Sc. and Ph.D. degrees in electrical engineering from Eindhoven University of Technology, Eindhoven, The Netherlands.

She is currently an Assistant Professor in the Electrical and Computer Engineering Department, University of California, Davis. Between 1996 and June 2003, she was a Senior Member of Research Staff at Philips Research, both in The Netherlands and the U.S., where she led a team of researchers working on scalable video coding, networking, and streaming algorithms and architectures. From January to September 2003, she was also an Adjunct Assistant Professor at Columbia University, New York. Since 1999, she has been an active participant to the MPEG-4 standard, for which she received an ISO recognition award. She is currently chairing the MPEG Ad-Hoc group on Scalable Video Coding, and is also Co-Chairing the Ad-Hoc group on Multimedia Test-bed. She has coauthored more than 90 book chapters, conference and journal papers in this field and holds 11 patents and several more pending.

Dr. van der Schaar is an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, was anAssociate Editor of the *SPIE Electronic Imaging Journal*, and a Guest Editor of the *EURASIP* Special Issue on Multimedia over IP and Wireless Networks. She was elected Member of the Technical Committee on Multimedia Signal Processing of the IEEE Signal Processing Society. She has also chaired and organized many conference sessions in this area and was the General Chair of the Picture Coding Symposium 2004. In 2004, she received the NSF Career Award.

Beatrice Pesquet-Popescu received the engineering degree in telecommunications from the "Politehnica" Institute in Bucharest in 1995 and the Ph.D. thesis from the Ecole Normale Supérieure de Cachan in 1998. In 1998 she was a Research and Teaching Assistant at Université Paris XI and in 1999 she joined Philips Research France, where she worked during two years as a research scientist in scalable video coding. Since Oct. 2000 she is an Associate Professor in multimedia at the Ecole Nationale Supérieure des Télécommunications (ENST). Her current research interests are in scalable and robust video coding, adaptive wavelets and multimedia applications. EURASIP gave her a "Best Student Paper Award" in the IEEE Signal Processing Workshop on Higher-Order Statistics in 1997, and in 1998 she received a "Young Investigator Award" granted by the French Physical Society. She holds 20 patents in wavelet-based video coding. Beatrice Pesquet-Popescu is co-Guest Editor for a special issue of the EURASIP Journal on Applied Signal Processing on "Video Analysis and Coding for Robust Transmission."