Generalized Rate-Distortion Optimization for Motion-Compensated Video Coders

Yan Yang and Sheila S. Hemami, *Member, IEEE*

***Abstract*—This paper addresses jointly rate-distortion optimal selection of coding parameters in a general motion-compensated video coder. The general coder uses variable-block-size motion estimation and multimode residual coding. This is essentially the optimal bit-allocation problem for an individual frame at a given rate constraint. This paper not only gives the general formulation and solution using the Lagrange multiplier method and dynamic programming, but also demonstrates how the general theory can be adapted and applied to both an MPEG-like coder and a motion-compensated wavelet coder. Simulations demonstrate that both proposed coders outperform MPEG (TM5) by 0.7–1.3 dB at a variety of bit rates, with the gain provided by both better motion estimation and the joint-parameter optimization. The technique is applicable to MPEG-compliant coders with fixed block-size motion estimation and provides a gain of 0.5–0.7 dB over TM5. The optimization approach can also be applied to distortion-constrained coding, and therefore allows a fine tuning of either the rate or distortion to follow any desired profile.**

***Index Terms*—Mode selection, motion compensation, MPEG, rate-distortion optimization, video coding, wavelet video coding.**

I. Introduction

A TYPICAL motion-compensated predictive video-coding system consists of motion estimation and motion compensation (MEMC) and transform coding (TC) modules. MEMC provides temporal redundancy reduction between adjacent or closely spaced frames. For both simplicity and efficiency, MEMC is usually performed on blocks rather than on individual pixels; the frame being coded is divided into either fixed- or variable-size blocks and then the motion vectors (MVs) for each block are estimated using block matching. Compared to using fixed-block-size motion estimation, variable-block-size motion estimation (VBSME) is more general and can better adapt to motion discontinuities such as moving edges, while fixed block size is simply a special case of VBSME. Current standards such as MPEG-1 [1], MPEG-2 [2], and H.261 [3] employ fixed-block-size motion estimation, while H.263 [4] allows VBSME. After MEMC, the displaced frame difference (DFD) is formed as the residual of the current and the predicted frame. Transform coding further exploits spatial redundancy across the DFD. Current standards [1]–[4] use the discrete cosine transform (DCT) along with variable-length coding, such as run-length and Huffman coding. As an alternative, the DFD can be coded using a wavelet transform (WT) followed by arithmetic coding, as proposed for the MPEG-4 standard [6], [7].

To achieve content adaptability, each block from the VBSME structure is labeled as one of several modes (mode selection) and encoded accordingly. In an MPEG-like coder, a given macroblock can be intra-frame coded, inter-frame coded using motion-compensated prediction, or simply replicated from the previously decoded frame. Quantization and coding are performed differently for each block according to its mode. In a motion-compensated wavelet coder (MCWC), each block is first labeled as one of the modes. Different regions can thus be formed by grouping blocks of the same mode. By first allocating bits among the regions, coding is performed region by region instead of block by block.

Regardless of the specifics of MEMC, TC, or mode selection, the rate-distortion (R-D) performance of the video coder is dependent on the complete set of parameters from both the MEMC and TC modules. These parameters can be but are not limited to the VBSME structure, the MVs, the mode selection for each block, and either the quantizer step sizes for each block (in an MPEG-like coder) or the region rate allocation (in an MCWC). Current implementations of the standards treat MEMC and TC separately and the parameters are usually selected based on distortion only. For example, the MVs are found using MSE or MAE as the matching criterion; a buffer-constraint-based rate control, which is independent of MEMC, is used to select the quantization scaling factor (MQUANT) which scales the DCT coefficient quantization matrix; and the mode selection for each block is based on distortion only. Since the ultimate goal of a video coder is to minimize the distortion given a rate constraint or to minimize the rate given a distortion constraint, R-D optimized decision making is essential to guarantee good coding performance [8]–[17]. For example, one parameter set might result in a higher distortion but a much lower rate and thus overall better R-D performance than another parameter set. Moreover, the choice of one of these parameters will influence the others in the final coding performance, and thus it is desirable to jointly optimize them instead of individually or sequentially selecting them.

Generally speaking, a complete optimal bit allocation algorithm for a motion-compensated predictive video coding system includes both frame-level bit allocation to meet a rate constraint over a group of frames (GOP) and within-frame bit allocation with a rate constraint for each frame. The former case of frame-level bit allocation is addressed in [18]–[21]. This paper only considers the latter case, i.e., optimal bit allocation for an individual frame at a given rate constraint. Early research on the

Manuscript received November 23, 1998; revised January 24, 2000. This paper was recommended by Associate Editor R. Lanchini.

Y. Yang was with Cornell University, Ithaca, NY 14853 USA. She is nnow with Aware, Inc., Bedford, MA 01730 USA (e-mail: yyang@aware.com).

S. S. Hemami is with the School of Electrical Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail: hemami@ee.cornell.edu).

Publisher Item Identifier S 1051-8215(00)07560-1.

joint R-D optimization focused on the optimal bit allocation between the MV selection and the DFD coding [8]–[10], which attempted to solve the problem from an *ad hoc* approach or to use only an entropy estimation for the DFD coding. More recent work is entirely for MPEG-like coders and can be grouped into two categories: those that perform an optimization of the TC coding parameters given a fixed set of MVs, and those that include MEMC in the optimization. The Lagrange multiplier method with dynamic programming is the most commonly used approach.

In the first category, the overhead for the selection of MQUANT is considered in [11] to achieve the R-D optimization. Motion estimation and mode selection are only distortion based. Joint MV and mode selection are not considered. In [12], mode selection that optimizes [for a given group of blocks (GOB)] the overall performance in the R-D sense is considered. The selection is made assuming that MQUANT is fixed for a GOB, and the MVs are selected independently of the mode or MQUANT selection. This work is extended in [13] where MQUANT and mode selection are jointly optimized with a given set of MVs. A near optimal m-best search is used. A heuristic approach for mode selection is proposed in [14], where a spatial-masking-activity weighted quantizer scale is used as the distortion measure. The mode selection minimizes the overall rate subject to uniform distortion over the picture, again for a given set of MVs. The optimal quadtree for VBSME along with the best quantizer selection in the R-D sense is considered in [15]. But each block is independently optimized: MVs are computed using the MAE criterion and mode selection is not included in the joint decision.

In the second category, a joint-optimization approach is proposed in [16] to achieve optimal bit allocation among a segmentation description, MVs, and the DFD. This parameter selection along with the coding mode decision are made based on the fixed MQUANT and DCT only; no generalized formulation for different TC is discussed. In [17], R-D optimal motion estimation is considered to achieve the optimal tradeoff between the MV coding and DFD coding. A fast practical approach reduces the computational complexity, but it addresses neither the optimal mode nor quantizer selection.

This paper proposes a generalized R-D optimized approach for the joint selection of coding parameters in motion-compensated predictive video coders; namely the quadtree structure in the case of the VBSME, and the MV, the mode and the DFD coding parameter selection associated with each tree node. The framework is developed in the context of VBSME for generalization purposes. In particular, VBSME using quadtree splits on superblocks is considered. Such an implementation allows compatibility with the forthcoming MPEG-4, and has also been commonly proposed as a practical implementation for VBSME. Fixed-block-size motion estimation is simply a special case of VBSME with a fixed quadtree pattern for each superblock. VBSME allows a tradeoff between the bits allocated to MVs and the bits allocated to the DFD, and is therefore well suited as an MEMC model for the joint parameter optimization.

This optimization problem is essentially an optimal bit-allocation problem which can be solved in the framework of R-D theory. It is shown that the generalized R-D optimization problem can be converted to a parametric decision problem where different sets of parameters which affect the R-D function can be chosen jointly. The problem can be generally formulated using the Lagrange multiplier method and solved using DP, though a fast algorithm is required to reduce the computational complexity. To demonstrate both the practical application of the joint optimization as well as both objective and subjective results, the optimization is applied to two common types of motion-compensated predictive video coders, namely, VBSME with DCT DFD coding and VBSME with wavelet transform DFD coding. Experimental results demonstrate that jointly optimal parameter selection in these coders exceeds the coding performance of TM5 [5] by 0.7–1.3 dB.

The paper is organized as follows. Section II formulates the general problem using the Lagrange multiplier method, and then provides a theoretical solution for the global optimization using DP. Sections III and IV detail the general formulation and solution for the MPEG-like coder and the MCWC, respectively. Section V addresses the practical implementation issues and proposes a near-optimal greedy approach. Section VI presents experimental results and Section VII concludes the paper.

II. General Problem Formulation

The VBSME is represented by a quadtree. Each block in a quadtree is called a *tree node*. A block is called a *parent node* if it is further branched into four nodes, which are called *child nodes*. A block is called a *leaf node* if it is not further subdivided. The tree can be represented by a series of bits that indicate termination by a leaf with a "0" and parent nodes with a "1." A three-level quadtree with a superblock structure instead of a complete tree is used here. A frame is segmented into contiguous 32×32 superblocks, each of which can be further quadtree split into 16×16 and 8×8 blocks. The superblock structure instead of a full tree is used since 32×32, 16×16 and 8×8 block sizes give a good tradeoff between prediction and overhead information.

Let T be a particular tree structure for a frame which belongs to the set of all possible quadtree structures $\mathbf{T}$. Let $N(T)$ denote the number of tree nodes in T. Assume there are K parameters which impact the rate and distortion for a given tree node. For example, in an MPEG-like coder, each block can have a MV, a mode label, and an MQUANT as coding parameters. Each block also has a rate and distortion associated with those parameters. Denote each of the parameters by p_i, where $i = 1, \ldots, K$. Let each p_i take on values from the set $\mathbf{p}_i = \{p_{ij}: j = 1, \ldots, L_i\}$, with the restriction that each set $\mathbf{p}_i$ is finite. L_i indicates the number of all the possible choices of each p_i. Denote a particular collection of these parameters by $\wp = (p_1, \ldots, p_K) \in \mathbf{P} = \mathbf{p}_1 \times \ldots \times \mathbf{p}_K$. The number of possible choices of $\wp$, denoted as $M_\wp$, is therefore $\prod_{i=1}^{K} L_i$.

A parameter set assigned to the nodes in T is given by a $N(T)$-tuple, $P = (\wp_1, \ldots, \wp_{N(T)})$, where each $\wp$ corresponds to the coding parameter set for each block in the tree. The optimization problem is then to jointly determine the quadtree T and parameter set P associated with all the tree nodes in T such

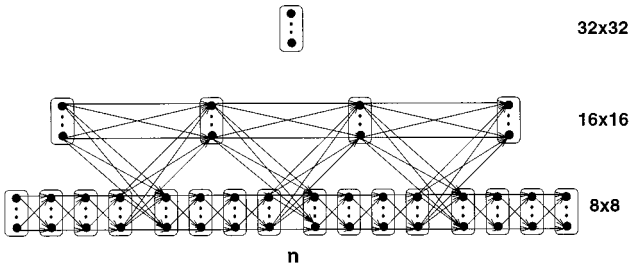


Fig. 1. Multilevel trellis for the nth superblock.

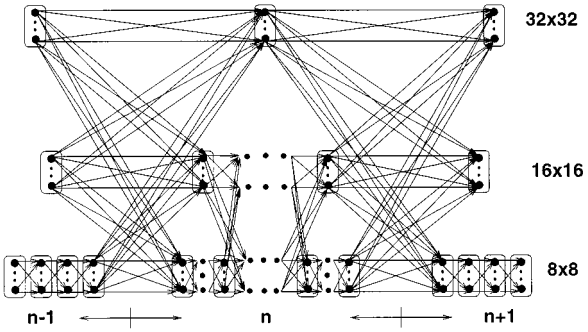


Fig. 2. Trellis representation for three consecutive superblocks.

that the total distortion D is minimized subject to the rate constraint R_{budget}

$$\min_{\mathbf{T}} \min_{\mathbf{P}} D(T, P) \quad \text{s.t.} \quad R(T, P) \le R_{\text{budget}} \tag{1}$$

where $D(T, P)$ and $R(T, P)$ represent the total distortion and rate, respectively, resulting from a particular combination of T and P. The following subsections reformulate the problem using the Lagrange multiplier method and give the theoretical solution using dynamic programming (DP).

A. Lagrange Formulation

By introducing the Lagrange cost

$$J(T, P) = D(T, P) + \lambda \cdot R(T, P) \tag{2}$$

the hard constrained optimization problem of (1) can be converted to an unconstrained problem using the Lagrange multiplier method and becomes

$$\min_{\mathbf{T}} \min_{\mathbf{P}} J(T, P) = \min_{\mathbf{T}} \min_{\mathbf{P}} [D(T, P) + \lambda \cdot R(T, P)]. \tag{3}$$

It can be shown [18],[25] that the solution (T^*, P^*) to (3) is also a solution to (1) for the particular case of $R(T^*, P^*) = R_{\text{budget}}$. Once λ^* is found such that (T^*, P^*) satisfies (3) and leads to $R(T^*, P^*) = R_{\text{budget}}$, then (T^*, P^*) is also the optimal solution for (1). When λ sweeps from zero to infinity, the solution to (3) traces out the convex hull of the rate distortion curve. Since $R(T, P)$ is monotonically inversely proportional to λ, a bisection or a fast convex search can be used to find λ^*.

The distortion of the reconstructed frame is assumed to be the sum of the distortion of individual nodes. Common distortion measures such as MSE and MAE fall into this class. Let $(b_1, \ldots, b_{N(T)})$ be the tree nodes (blocks) in a certain scanning order for a given tree T. Then the total rate for the frame is the sum of the rates for all the nodes, and (3) can be written as

$$\min_{\mathbf{T}} \min_{\mathbf{P}} J(T, P) = \min_{\mathbf{T}} \min_{\mathbf{P}} \sum_{i=1}^{N(T)} J(b_i, P). \tag{4}$$

The solution to (3) is unwieldy due to the rate and distortion dependencies manifested in the $D(T, P)$ and $R(T, P)$ terms. Without further assumptions, the resulting distortion and rate associated with a particular node are inextricably coupled to the chosen parameters for every other node in the tree. The global minimization of the cost function requires an exhaustive search over all

$$(T, P) \in \mathbf{T} \times \overbrace{\mathbf{P} \times \ldots \times \mathbf{P}}^{N(T)}.$$

The computational requirements are impractical for most applications. However, for typical video-coding systems, constraints are often imposed that can simplify the optimization problem.

For example, the simplest case assumes that both the rate and distortion for a given node are impacted only by the current block and its respective operational parameters. Thus, (4) can be further simplified to

$$\min_{\mathbf{T}} \min_{\mathbf{P}} \sum_{i=1}^{N(T)} J(b_i, P) = \min_{\mathbf{T}} \sum_{i=1}^{N(T)} \min_{\wp_i} J(b_i, \wp_i) \tag{5}$$

and thus can be easily minimized by independently selecting the best parameters for individual nodes in a quadtree. In most of the coding standards [1]–[4], block-to-block dependency exists such that the rate term for a given macroblock is dependent not only on the current block, but also on the adjacent blocks. For example, the DPCM coding of MVs and the DC values are dependent on the neighboring blocks. Thus, the structural constraint for the above simplification is too restrictive and leads to poor R-D performance. Since the complexity grows exponentially as the degree of dependency increases, and only 1-D dependency is present in current standard implementations, 1-D dependency is assumed in this paper; that is, the total influence on rate and distortion for any particular block is limited to that from the immediately preceding block in some scanning order. The Lagrange cost for the current block can then be written as

$$J(b_i, P) = J(b_i, \wp_i, \wp_{i-1}). \tag{6}$$

Thus, (4) becomes

$$\min_{\mathbf{T}} \min_{\mathbf{P}} \sum_{i=1}^{N(T)} J(b_i, P) = \min_{\mathbf{T}} \min_{\mathbf{P}} \sum_{i=1}^{N(T)} J(b_i, \wp_i, \wp_{i-1}). \tag{7}$$

In the following, deterministic DP is used to find the solution to this 1-D dependency optimization problem.

B. DP Optimization

This constrained optimization over finite discrete sets can be solved using DP [26]. In the above problem, once the parameter set $\wp_n$ for the nth block is known, the distortion and the DFD

coding rate of the block depend entirely on $\wp_n$. The 1-D dependency is introduced only by the differential coding of the $\wp_n$. It can be shown that a DP recursion formula can be established and a trellis can be constructed where states of the trellis represent all the possible choices of $\wp_n$ [16]. By associating each arc between trellis states with the Lagrange cost, the deterministic finite-state optimization problem is converted to the shortest path finding problem. The special case of forward DP, known as the Viterbi Algorithm (VA), can be used to find the optimal solution. Using the VA, only one incoming path (the minimal cost path) is kept for each trellis node at each stage. The accumulated cost and the path are recorded. The shortest path can be found at the end of the trellis along with the optimal trellis states which are back traced.

The intuitive approach to solve the minimization problem (3) by the VA is to first grow the trellis for a given tree T and find the shortest path for T. Then the optimal solution is found by performing the minimization for all trees within the possible quadtree set $\mathbf{T}$. For a given tree, the ith tree node in the scanning order represents the ith stage in the trellis. At each stage, there are $M_\wp$ trellis states, each representing a set of control parameters $\wp$ for the corresponding block.

Since the superblock structure is used in this paper, an alternative approach is adopted which forms a multilevel trellis by incorporating the possible subtree structures into the trellis [16]. This is illustrated in Figs. 1 and 2. In both figures, each rectangle indicates the parameter set $\mathbf{P}$ for the corresponding block. The black circles inside the rectangle are the trellis nodes where each node represents one particular choice of $\wp$ (for example, a combination of MV, mode, and quantization step size in the MPEG-like coder) which is used to encode the block. Each node also contains the resulting distortion and rate. The distortion and DFD coding rate are dependent only on the current $\wp$ while the overhead rate for coding the $\wp$ depends on that of the immediately preceding trellis node too.

Fig. 1 shows the multilevel trellis for an individual superblock. It consists of trellis nodes of all possible 3 level quadtree decomposition from 32×32 to 16×16 and 8×8 blocks. The permissible transitions from one trellis node to another are restricted by the possible quadtree decompositions. For example, there are no transitions from the 32×32 level to any of the 16×16 or the 8×8 levels in the same superblock. Thus the optimal path is forced to select only valid quadtree structures.

Fig. 2 shows the trellis structure for three consecutive superblocks in a scanning order. A sample raster scanning order is shown in Fig. 3(a), where each superblock row is scanned from the left to right and top to bottom. The subtrees for each superblock are scanned in a top-left to bottom-right order, as illustrated for an example subtree in Fig. 3(b). Each tree node can have $M_\wp$ trellis states. Other scanning orders, such as the Hilbert scan [16], can also be used. Because of the limited scope of this paper, the optimality of the scanning order is not included. Since the transition is from left to right, the figure shows only the states associated with the rightmost 16×16 block and its four 8×8 children of the $n-1$th superblock and those associated with the leftmost 16×16 block along with its four 8×8 children for the $n+1$th superblocks. For the nth superblock, only the leftmost and the rightmost 16×16 blocks along with their leftmost and rightmost two children are shown to demonstrate the transitions. As illustrated, the transitions between the trellis nodes of two superblocks include all the possible transitions from level to level.

Once the trellis is formed, the VA is applied to find the shortest path for a given λ. The final optimal solution is achieved by iterating through λ to satisfy the rate constraint.

From the above problem formulation, note that the fine tuning of rate or distortion is accomplished via a single parameter λ, with the desirable outcome that for a given rate constraint, the distortion is minimized. This is in contrast to the *ad hoc* strategies in the standard implementations which typically scale a single parameter such as the quantizer step size to control the instantaneous rate, but cannot guarantee any type of optimal R-D performance.

The dual problem, which minimizes the rate for given distortion constrained, can be stated as

$$\min_T \min_P R(T, P) \quad \text{s.t.} \quad D(T, P) \leq D_{\text{budget}}. \tag{8}$$

The same technique can be used to solve this problem. This type of optimization is hence suitable for applications which require either constant bit rate or constant distortion.

The Lagrange formulation and DP solution for the block-based parameter selections define a general dependent optimal bit-allocation problem. In any implementation, the actual coding parameter set depends on the TC type. The rate and distortion for each block can be either exactly calculated by actual coding or estimated by a model-based approach. In an MPEG-like coder, exact calculation of the rate and distortion is possible, since the block-based transform and variable-length coding with a look-up table are used for the TC. In an MCWC, the rate and distortion can only be estimated using statistical assumptions about input distributions and quantizer characteristics. In the following two sections, this general formulation is applied to both an MPEG-like coder and an MCWC by addressing the definition of the parameters and the calculation of the rate and distortion.

III. Application to MPEG-Like Coders

This section applies the general framework to MPEG-like coders, which includes the current standards (MPEG-1, MPEG-2, H.261 and H.263). To illustrate the application to MPEG-like coders, TM5 is first introduced. The above general formulation and solution is then applied using the identical DCT and MV coding to TM5. Emphases are on the choice of the coding parameters and the rate and distortion calculation in order to construct the trellis.

A. Introduction to TM5 Coding

In MPEG, a frame is divided into coding units called macroblocks that may contain a single 16×16 luminance block and two 8×8 chrominance components. A motion vector for each macroblock is determined by block matching using MAE. The MV is differentially coded using a Huffman table. Macroblock coding is performed in a *multimode* fashion: a macroblock

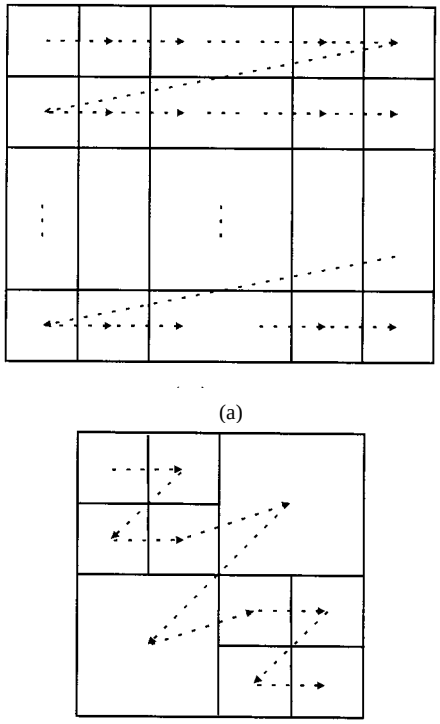


Fig. 3. Sample scanning order for (a) 32×32 superblock and (b) subtree inside a superblock.

can be intra-frame coded (*intra* mode), inter-frame coded using motion-compensated prediction (*inter* mode), or simply replicated from the previously decoded frame (*skip* mode). The MSE is used as the criterion for mode choice. For example, the intra-/inter-decision is determined by a comparison of the MSE of the macroblock pixels against the MSE of the DFD. The mode selection is signaled by both *macroblock_type* and *coded_block_pattern* which are also variable-length coded and included with the quantized coefficients. The transform coder performs a DCT, quantization and Run-Length Huffman coding on 8×8 blocks for each macroblock. Different quantization matrices and coding tables are used depending on the macroblock mode. The actual quantization step sizes for each macroblock are given by the mode-specific quantization matrix scaled by an integer called MQUANT, which is chosen using a buffer-constraint-based rate-control scheme.

B. Application of the General Framework

In TM5, MV determination and mode selection are distortion-based only, while MQUANT is selected using a buffer-constrained rate-control scheme which does not consider distortion. Moreover, the MV, coding mode, and MQUANT are selected independently without considering the tradeoff between the overhead rate (rate required to transmit MVs, mode and MQUANT information) and DFD rate. The overhead rate consists of both MV's information and header information which includes the description of the mode and the MQUANT. Generally, the R-D characteristics of a macroblock depend on those of the other macroblocks because of the dependency of the parameter choices between this macroblock and the other macroblocks. For example, a mode selection of *skip* is dependent on the value of the MQUANT. Hence, it is desirable to perform a joint optimization for all the parameters of the block according to the operating rate. To describe the application of the general framework for the MPEG-like coder, only forward motion estimation is considered (P frames only). A similar approach can be applied to bidirectionally predicted frames (B-frames) if necessary.

1) Coding Parameter Choice: Three modes $\{$intra (I), inter (P), skip (S)$\}$ are used. Intra mode is identical to that in TM5. Inter mode is used for blocks which are predicted and the DFD is coded. Skip mode is used for blocks which are noncoded, which is slightly different than TM5's skip mode. This more general skip mode includes all the blocks without DCT coefficients, for both zero and nonzero MVs. The variable-length codes $\{11, 10, 0\}$ are used to code the modes $\{$I, P, S$\}$, respectively.

Using the notation of the general formulation in the previous section, the control parameter set which influences the rate and distortion for a tree node is $\{$MV, mode, MQUANT$\}$. Specifically, let p_1 represent the MV , which is chosen from the set $\mathbf{p}_1 = \{-16, -15.5, -15, \ldots, 15.5\}$. Let p_2 represent the mode choice, with $p_2 \in \mathbf{p}_2 = \{I, P, S\}$. MQUANT is represented by p_3 which can be chosen from the set $\mathbf{p}_3 = \{1, 2, \ldots, 31\}$. For a given tree, the scanning order of all the blocks in the tree is found by integrating both the superblock scanning [Fig. 3(a)] and the subtree scanning inside the superblock [Fig. 3(b)]. Let $b_1, \ldots, b_N$ indicate all the blocks in the scanning order. The set of states for each block is $(\{I\} \times \mathbf{p_3}) \cup (\mathbf{p_1} \times \{P\} \times \mathbf{p_3}) \cup (\mathbf{p}_1 \times \{S\})$. The DCT, inverse DCT, quantization and run-length Huffman coding are performed in exactly the same manner as in TM5.

2) R-D Calculation: There are three 1-D dependencies in the rate calculation. First, the MVs are 1-D differentially coded in the scanning order. Secondly, in the case of intra coding, if the previous block is intra coded, then the DC value of the currently coded intra block is differentially coded with respect to the previous DC value. Thirdly, the coding of MQUANT is dependent on the previous MQUANT. One bit indicates whether or not the current MQUANT is the same as the previous MQUANT. If not, then more bits (in this case five) are used to code the current MQUANT.

The rate calculation for the different coding modes is now described in detail. Four terms that appear in the rate calculation are:

1) R_T: bits for quadtree structure, fixed-length coded;
2) R_{DFD}: bits for DFD coding, variable-length coded;
3) R_{header}: bits for header information, variable-length coded;
4) R_{MV}: bits for MV coding, variable-length coded;

Intra Mode: The rate equation for intra-mode blocks is

$$R(b_i, b_{i-1}) = R_T(b_i) + R_{\mathrm{DFD}}(b_i, b_{i-1}) + R_{\mathrm{header}}(b_i, b_{i-1}). \tag{9}$$

For R_T, only 1 bit is needed for the quadtree block at the superblock (lowest) level to indicate whether or not it is split. For each of the next two levels, four more bits are needed to indicate whether or not each of the four child nodes is split. This

quadtree coding cost has to be distributed to the trellis nodes of the child blocks. How the cost is split among the child nodes is arbitrary, since every path has to either go through all of the child nodes or none of them. The path will pick up the quadtree coding cost no matter how it is distributed among the four children.

$R_{\mathrm{DFD}}(b_i, b_{i-1})$ indicates the bits for image block coding (notice that the notation is R_{DFD}, but it is actual image block coding instead of DFD coding). The rate is determined by actually performing the DCT, quantization and run-length Huffman coding which is attainable for an MPEG-like coder. This rate is also a function of previous block since the DC value might be coded differentially from the previous DC value if the previous node is also intra coded. An alternative way to calculate R_{DFD} is to use a model-based approach [22]–[24] for fast approximation. These approaches estimate the R-D performance by using parametric functions obtained either experimentally or using a certain mathematical distribution such as the generalized Gaussian or Laplacian distribution. For the purpose of the demonstration of the general framework, real coding is performed in this paper.

For $R_{\mathrm{header}}(b_i, b_{i-1})$, coding MQUANT requires either 1 bit or 6 bits depending on whether or not the current MQUANT is the same as the previous MQUANT. One or two bits are needed for coding the mode depending on whether the block is skipped or nonskipped.

Inter Mode: The rate equation for inter-mode blocks is

$$R(b_i, b_{i-1}) = R_T(b_i) + R_{\mathrm{DFD}}(b_i) + R_{\mathrm{header}}(b_i, b_{i-1}) + R_{\mathrm{MV}}(b_i, b_{i-1}). \quad (10)$$

The calculation of R_T and R_{header} are the same as those of the intra mode. R_{DFD} depends only on the current block since there is no differential coding of the DC value. The new term $R_{\mathrm{MV}}(b_i, b_{i-1})$ indicates the bits for MVs, which are 1-D DPCM coded. The predictor is either the previous block MV or zero, depending on whether the previous block is intra coded or not. The exact rate for R_{MV} is calculated using the variable-length table defined in TM5.

Skip Mode: The rate equation for skip-mode blocks is

$$R(b_i, b_{i-1}) = R_T(b_i) + R_{\mathrm{header}}(b_i) + R_{\mathrm{MV}}(b_i, b_{i-1}). \quad (11)$$

There is no R_{DFD} term since no DCT coefficients are coded. Without MQUANT, R_{header} depends only on the current block. The coding of the MV is different than that in the inter mode: 1 bit is sent first to indicate whether or not the MV is zero. For the nonzero case, the MV is then differentially coded.

For intra- or inter-mode blocks, the distortion is calculated as the sum of the squared error between the coefficients before and after quantization. For skip-mode blocks, the distortion is the sum of the squared DFD. For blocks larger than 8×8, the distortion and rate are obtained as the sum of the distortion and rate of each 8×8 child block.

Once the rate and distortion for each trellis node are found, the VA can be applied to find the optimal state sequence associated with $\{b_1, \ldots, b_N\}$.

IV. Application to MCWCs

As an alternative approach to the DCT, wavelet compression techniques have been used in both still image and video coding [6], [7], [29]. The MCWC introduced in [6], [7], uses a region-based embedded wavelet coding approach. By first classifying all the 8×8 or 16×16 blocks in each frame as motion failure or nonfailure mode, coding is performed only on the motion-failure region, which is formed by all the blocks with failure mode. The classification is performed using a variance-based approach together with *ad hoc* thresholds, and no R-D optimization is performed for either the mode selection or motion estimation. Similar to the MPEG-like coder, it is important to consider the joint optimization of the mode selection and motion estimation to achieve the overall optimal R-D performance. In this section, the general framework in the context of the VBSME and the region-based embedded MCWC is applied. The region-based embedded wavelet DFD coding is first introduced. The application is then described by addressing both the coding parameter choices and the R-D calculation.

A. Region-Based Embedded Wavelet DFD Coding

In this application, a region-based embedded wavelet coding technique [6], [7] codes the DFD. This coder consists of a wavelet decomposition followed by embedded wavelet coding of selected regions. Different regions are formed by grouping blocks of the same mode. Prior to the coding, a full-resolution region map is formed by setting all the blocks in that region to the region label. The multiresolution region map is then formed by down-sampling the full resolution map at each decomposition level in order to adapt to the multiresolution nature of the wavelet transform. Fig. 4 shows an example of a map for the two-region case where the block can either be of nonskipped mode or skipped mode. The DFD is first wavelet transformed using a (2, 6) biorthorgonal filter with a three-level decomposition. Short analysis filters are used to limit the area of spatial influence of one wavelet coefficient. The coefficients are then coded region by region according to the multiresolution map.

An embedded coding technique [29] is used to code the coefficients in a region. Similar to other embedded coding techniques, this approach employs successive quantization and codes the DFD bit planes to the exact bit. To achieve the optimal bit allocation within the frame, the coder optimizes the coefficient coding order by selecting the coefficient with the maximum estimated R-D slope. To avoid explicitly transmitting the coding order, the expected R-D slope is calculated using context modeling, which can be obtained from the previously coded results. Because of near-optimal coefficient coding order, this coder demonstrates outstanding R-D performance. To maximize performance, the context is modified to adapt to the statistics in the DFD coding.

B. Application of the General Framework

The application of the general framework is coder specific, that is, coding parameters and the R-D calculation can be different with different coders. This section addresses the appli-

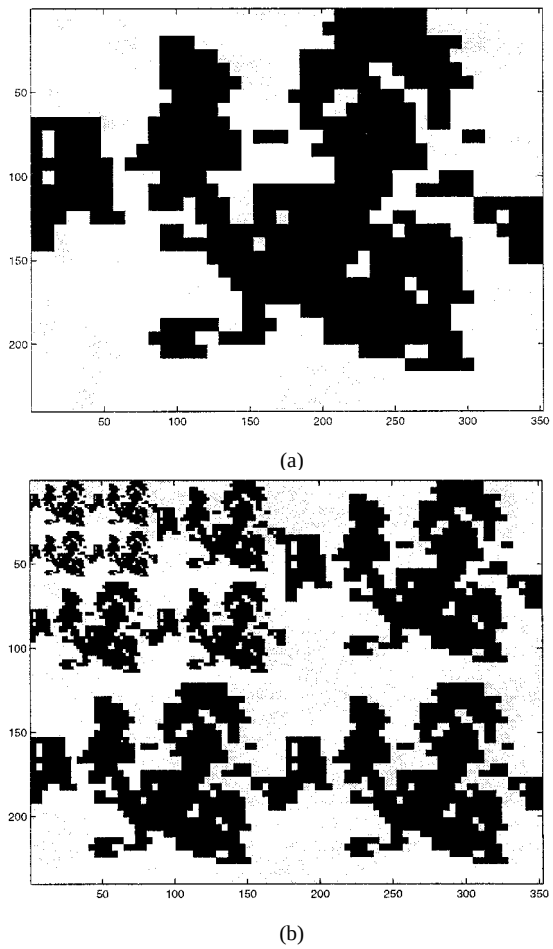


Fig. 4. Skipped and non-skipped (inter-coded) binary region map for wavelet coding. (a) Full-resolution binary map (black: skipped region, white: nonskipped (inter-coded) region). (b) Binary map modified for three-level wavelet decomposition.

cation of general framework to the MCWC by discussing the coding parameter choice and the R-D calculation.

1) Coding Parameter Choice: Since bit-plane coding is used, quantization is implicit instead of explicit. Hence the parameter set is reduced to {MV, mode}. Specifically, let p_1 indicate the MV, which can be chosen from the set $\mathbf{p}_1 = \{-16, -15.5, -15, \ldots, 15.5\}$. Let p_2 correspond to the mode choice. Two modes are used: inter mode and skip mode, so $\mathbf{p}_2 = \{I, S\}$. Skip mode indicates that the allocated rate to the DFD is zero. Inter mode indicates otherwise. The intra mode is omitted in this MCWC application. When the video is coded at 30 frames per second (fps), a small proportion of blocks are labeled as intra, and the number of bits allocated to the intra region is not high enough to fully exploit the efficiency of the embedded coder. However, intra mode should be included for low-frame-rate applications where a larger proportion of the frames cannot be successfully predicted.

2) R-D Calculation: The same four rate terms of R_T, R_{DFD}, R_{header} and R_{mV} as those introduced in the previous section are used. For inter mode, the calculation of the R_T and R_{MV} are the same as those discussed in the MPEG-like coder in section. However, the calculations of $R_{\mathrm{header}}(b_i)$, $R_{\mathrm{DFD}}(b_i)$ and $D(b_i)$ are different. Because there is no MQUANT coding, R_{header} is a function only of the current node. Since coding is performed on the region instead of block by block, it is intractable to calculate the $R_{\mathrm{DFD}}(b_i)$ and $D(b_i)$ exactly. Therefore, a model-based approach is adopted instead.

Since motion estimation is done in the pixel domain, the R-D estimation is performed directly in the pixel domain instead of the transform domain. To generalize, a weighted distortion measure is used. Assuming the source is i.i.d stationary Gaussian, the weighted distortion for the ith node is given by [28]

$$D(b_i) = w_{b_i}\sigma_{b_i}^2 2^{-2R_{\mathrm{DFD}}(b_i)} \tag{12}$$

where w_{b_i} and σ_{b_i} are the weight and the variance of the DFD associated with the corresponding block. R_{DFD} with a given λ can be found theoretically by

$$R_{\mathrm{DFD}}(b_i) = \begin{cases} \frac{1}{2}\log_2(2\ln 2)w_{b_i}\sigma_{b_i}^2/\lambda, & \text{if } \sigma_{b_i}^2 > \lambda \\ 0, & \text{otherwise} \end{cases}. \tag{13}$$

Distortion is found using (12) once the rate is calculated. For the skip mode, the same R-D calculation as the MPEG-like case applies.

Upon finding the rate and the distortion, the trellis can be constructed and the VA can be applied to find the shortest path and hence frame coding parameters.

V. Implementation

Although the VA can be used to find the globally optimal solution, the large number of possible states in each stage poses prohibitively high computational and memory requirements. This section addresses the complexity reduction first and then summarizes the optimization process.

A. Complexity Reduction

The complexity of the exhaustive search is determined by the number of all the possible paths, which is $O((\sum_{i=1}^{M_{\mathrm{tree}}} M_{\wp}^{N_i})^{M_{32\times 32}})$, where M_{tree} is the number of possible subtree structures for a three-level quadtree of a 32×32 superblock. $M_{\mathrm{tree}} = 1 + 2^{4^1} = 17$. N_i denote the number of tree nodes of the ith subtree. $M_{\wp}$ is the number of possible trellis states for each tree node. $M_{32\times 32}$ is the number of 32×32 blocks in each frame. Using the VA can reduce the complexity to $O((\sum_{i=1}^{M_{\mathrm{tree}}} M_{\wp}^2 N_i)^2 M_{32\times 32})$. This complexity is $O(10^{23})$ for the MPEG-like coder and $O(10^{19})$ for the MCWC. This is only the number of the path searches and does not yet include the cost generation. The complexity is further reduced through trellis state reduction and a suboptimal greedy solution.

1) Trellis-State Reduction: Since the R-D calculation is required for each trellis node and a large fraction of the nodes are far from R-D optimal, it is important to reduce the number of the trellis nodes in order to reduce the computational burden.

a) *Fast Search for MVs*:

For the MPEG-like coder, MV selection with the minimal Lagrange cost is very expensive because of the R-D generation through actual coding. Most of the candidate MVs result in exceedingly large distortion or rates, and

are thus far from optimal. A two-step search is used to reduce the number of states. In the first step, the full-pel MSE-optimal MV by the logarithmic search and the zero MV are used as the candidate MVs. The best combination of the MV and the MQUANT (*mv, mqaunt*) which gives the least Lagrange cost is found. In the second step, eight neighboring half pel positions for the MV and five choices of MQUANT ($\mathrm{mquant}-2, \ldots, \mathrm{mquant}+2$) are used for the search of the final best combination. For a search window of $(-16, 15)$ in both x and y directions, the number of states for inter mode is reduced from $63 \times 63 \times 31 = 123\,039$ for an exhaustive search to $2 \times 31 + 8 \times 5 = 101$. Thus, the complexity of the VA is reduced to $O(10^{14})$ for the MPEG-like coder.

For the MCWC, the cost generation is much less expensive using a model-based approach. Logarithmic fast search of the MV is performed using the Lagrange cost. For a search window of $(-16, 15)$ in both x and y directions, the number of the MV states is reduced to $\log_2 32 \times \log_2 32 + 8 = 33$ where the complexity is reduced to $O(10^{13})$.

b) *Reduction of Possible Subtree Structures*:

Each superblock has 17 possible quadtree structures, and quadtrees with more splits have more tree nodes (blocks). Thus it is desirable to avoid growing the full tree as much as possible. For background regions in the frame, large blocks can be sufficient for motion estimation, and since large blocks reduce the rate for both the block header and the MVs, using a low-level tree is very likely to be R-D optimal. Thus, a preprocessing algorithm [31] is used to obtain an initial superblock quadtree structure where superblocks with small DFD are not split further. Depending on the motion characteristics of the sequence, this can reduce the search complexity by another factor of 4–100.

2) Suboptimal Greedy Search: A greedy approach is used for a suboptimal performance at a fraction of the complexity. This approach keeps only the lowest cost branch thus far at all stages in the trellis, i.e., the selection of the optimal state at the current stage is forced based on the optimal state at the previous stage. This is demonstrated in Fig. 5, where a three-stage trellis with three states at each stage is used for illustration. This approach can reduce the complexity to $O(10^6)$ for the MPEG-like coder and $O(10^5)$ for the MCWC.

B. Optimization Procedure Summary

The optimization is performed by first preprocessing the frame to obtain the initial superblock structures. Then the optimal solution is found with Algorithm II by iteratively invoking Algorithm I. The optimization procedure is summarized as follows.

1) *Preprocessing*:
   1) Obtain the direct difference between the current frame and the reference frame.
   2) Obtain the binary significance map by thresholding the direct difference and removing the isolated significant pixels.
   3) Split the current frame into 32×32 superblocks.
   4) For each superblock, recursively quadtree split the block until the block contains no significant pixels or the block size reaches 8×8.

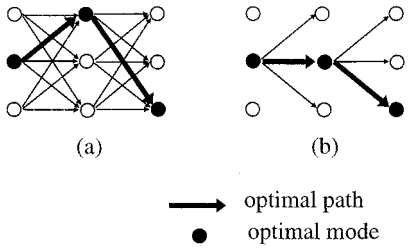


Fig. 5. Optimal and fast search. (a) Full search. (b) Greedy fast search.

• Algorithm I [for fixed λ, find (T, P)]:

1. For the first superblock in the scanning order, use the following steps (a. to c.) to find the optimal quadtree structure and the associated parameter set for all the subtree nodes.

a. Given the resulting subtree structure from the preprocessing, find all the possible subtrees by only allowing a merge of children nodes into parent nodes.

b. For all the blocks in each possible subtree, according to the scanning order search through all the possible states to find the best state (the parameter set which gives the minimum Lagrange cost) associated with the current node based on the best state of the previous node.

c. Find the best subtree with its associated parameter set which has the minimum total Lagrange cost.

2. Repeat step 1 for all the superblocks in the scanning order. The cost generation is based on the best subtree and the associated parameter set for the previous superblock.

• Algorithm II (iterative process):

1. Find initial λ_1 and λ_2 such that the resulting $R(T_1, P_1)$ and $R(T_2)$ from Algorithm I satisfy: $R(T_1, P_1) \leq R_{\mathrm{budget}} \leq R(T_2, P_2)$. If either of the equalities hold, the problem is solved.

2. Otherwise, let $\lambda_3 = \sqrt{\lambda_1 \lambda_2}$. Perform Algorithm I with λ_3 and obtain $R(T_3, P_3)$. If$(1 - \delta_1)R_{\mathrm{budget}} <= R(T_3, P_3) <= (1 + \delta_2)R_{\mathrm{budget}}$, then the problem is olved. Else if $R(T_3, P_3) > (1 + \delta_2)R_{\mathrm{budget}}$, let $\lambda_2 = \lambda_3$. Otherwise let $\lambda_1 = \lambda_3$. Repeat step 2.

VI. Experiments and Results

The coding algorithms were evaluated using *Football*, *Table Tennis* and *Flower Garden* video sequences [SIF (352×240), 30 frames/second, luminance only]. The performance of the RD-optimal algorithms for both the MPEG-like coder and the MCWC were compared with TM5. The TM5 encoding used a 15-frame GOP with coding pattern IBBPBBPBBPBBPBB (I: Intra frame; B: Bidirectionally predicted inter frame; P: Forward predicted inter frame). For TM5, this pattern has better performance than the P-only pattern. Both optimized approaches also used a 15-frame GOP, with every 15th frame intra coded, and only forward prediction in

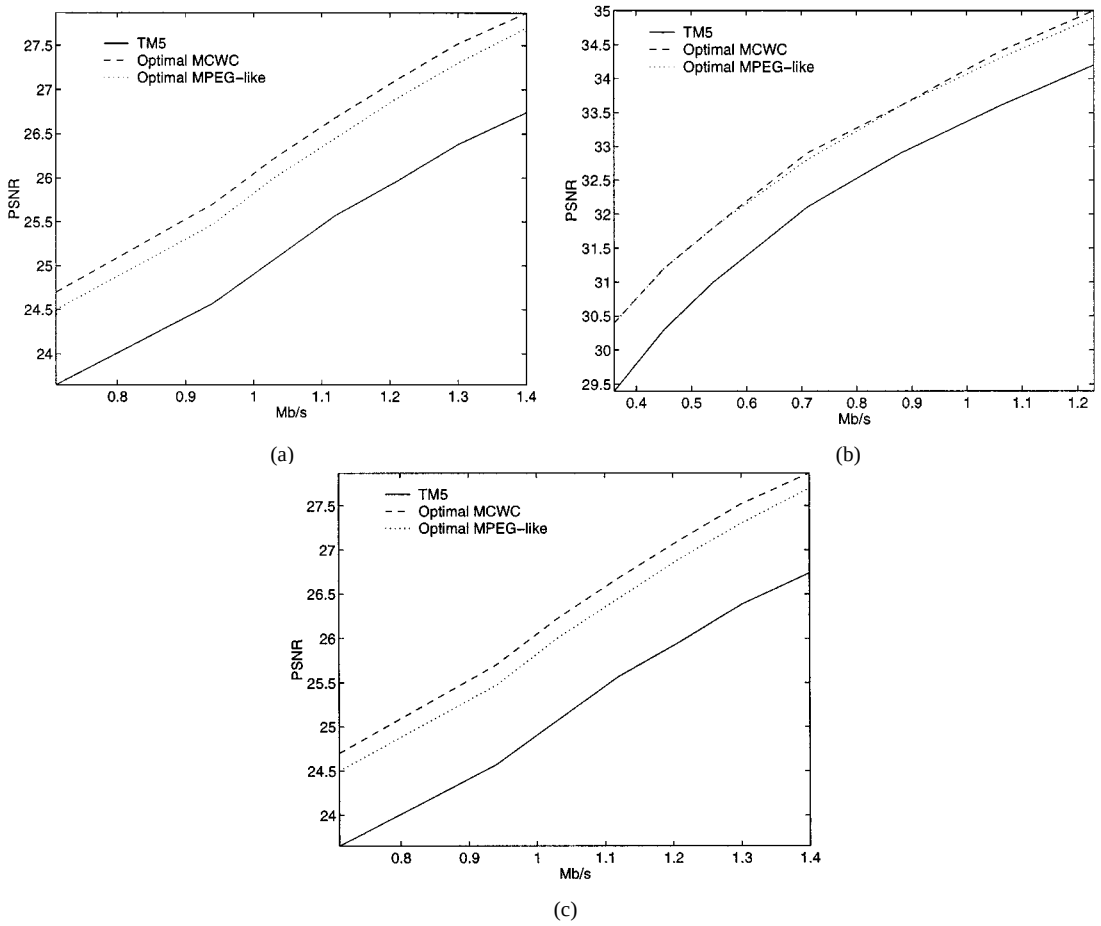


Fig. 6. PSNR versus rate. (a) *Football*. (b) *Table tennis*. (c) *Flower garden*.

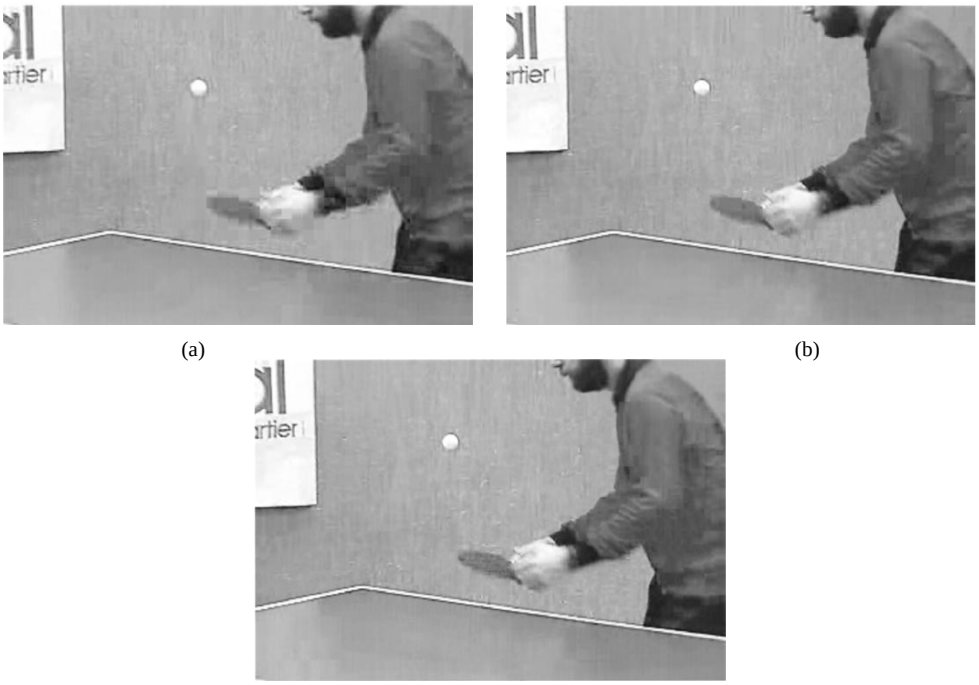
the motion estimation. B-frame adaptation is possible but is not implemented in this paper. For a fair comparison with TM5, all coders used the same intra frames from the TM5 coder at the corresponding total bit rate. Since no frame-level rate control was used in the R-D optimal coders, the inter frame bit rate was calculated as the average inter frame bit rate from the TM5 coder. For the MCWC, the exact average rate can be reached due to the embedded coding. For the MPEG-like coder, the mismatch rate is controlled using $\delta_1 = 1\%$ and $\delta_2 = 0.1\%$.

Both the R-D optimized coders result in significantly improved R-D performance and visual quality over TM5. Fig. 6 plots the PSNR vs. rate for the three sequences. For all the bit rates and all the sequences, both optimized coders achieved 0.7–1.3-dB gain over TM5. The individual frame PSNR gain can be as high as 2.2 dB. Since the same intra frames were used, the resulting gain is entirely due to better inter-frame coding. Subjectively, both approaches greatly reduce blocking effects within frames and mosquito effects around moving edges. Two frames which are coded as P frames in TM5 are used for demonstration. Fig. 7 shows the coding results for frame 47 in *Table Tennis*. The blocking effects around the paddle and the arm are significantly reduced in the MPEG-like coder and are nearly imperceptible in the MCWC. Fig. 8 shows the results for frame 29 in *Football* with similar visual results.

The improved results are due to a combination of the better motion estimation by using the variable block size and the optimal bit allocation by the jointly optimal selection of coding parameters.

If fixed block size is used in the motion estimation and mode classification, the optimized MPEG-like coder is completely MPEG compatible. This is achieved by using a fixed quadtree pattern in the optimal MPEG-like coder with each superblock consisting of four 16×16 nodes, and initializing the MV predictor to zero for the first macroblock of each row. An average of 0.5–0.7-dB PSNR gain over TM5 can be achieved.

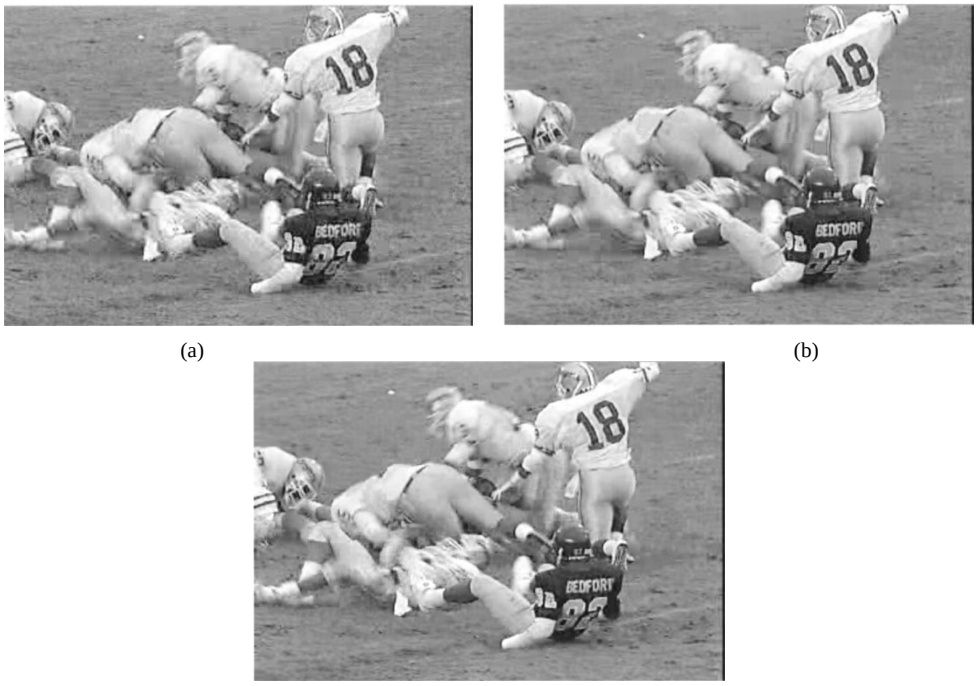
The VBSME provides an adaptive motion field with more bits allocated to MVs than in fixed-block- size ME, providing more accurate prediction and therefore requiring fewer DFD coding bits. By jointly determining the quadtree structure, the MV and the mode along with DFD coding parameters associated with each node in the tree, both optimized coders are able to achieve 0.7–1.3-dB gain over TM5. Compared to the fixed block size, the VBSME not only achieves additional 0.2–0.6-dB gain, but also produces improved visual quality especially around moving edges. Fig. 9 demonstrates the MV bit usage, the PSNR and the total bits for 90 frames (20th–109th frame) in *football* coded at 0.92 Mbits/s. The MV bit usage is shown in Fig. 9(a). The average bits for MV are 4318 and 7360 bits/frame for the optimal MPEG-like coder and the optimal MCWC, respectively, which are higher than 3146 for the TM5. But as can be observed from Fig. 9(b), both optimal coders



(a) (b)

(c)

Fig. 7. Comparison of the coded 47th frame in *Table tennis* coded using: (a) TM5, 12 143 bits, PSNR = 28.3 dB; (b) the optimal MPEG-like coder, 9771 bits, PSNR = 30.3 dB; and (c) the optimal MCWC, 9783 bits, PSNR = 30.5 dB.

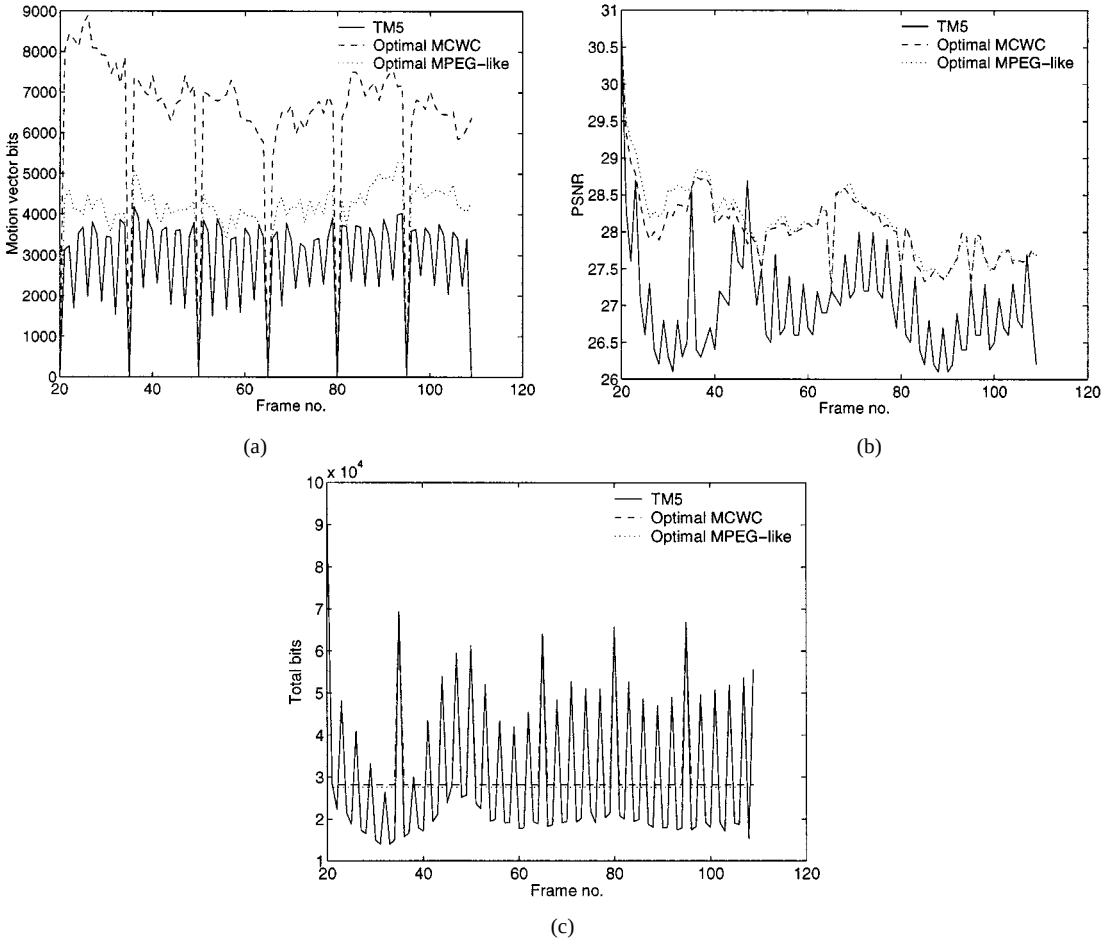


(a) (b)

(c)

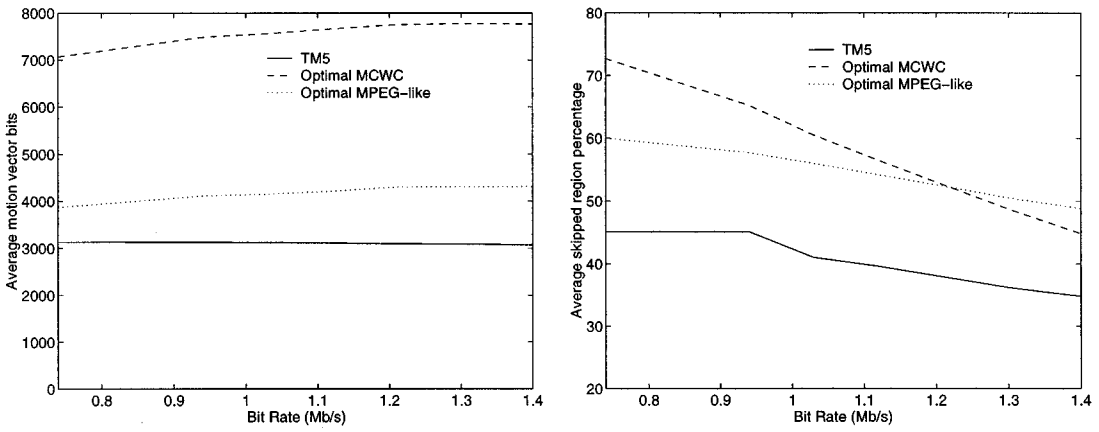
Fig. 8. Comparison of the coded 29th frame in *Football* using: (a) TM5, 33 170 bits, PSNR = 26.8 dB; (b) the optimal MPEG-like coder, 27 603 bits, PSNR = 28.3 dB; and (c) the optimal MCWC, 27 693 bits, PSNR = 28.1 dB.

result much higher PSNR (average of 1.2-dB gain for the optimal MPEG-like coder and 1.1 dB for the optimal MCWC). The average MV bit usages for different bit rates are demonstrated in Fig. 10(a). The optimized approaches used 30%–130% more bits for MVs than TM5, yet they achieved higher coding performance because of the better tradeoff between the MV bits and DFD coding bits. The average percentages of skipped region are also compared in Fig. 10(b) for three coders. It can be observed that higher bit rates result in lower regions that are uncoded or skipped. Both optimized approaches have more skipped regions than TM5 because of the better motion prediction and better quality of the previously coded frame. Despite the fact that skipped regions are noncoded in both R-D optimal coders,



(a) (b) (c)

Fig. 9. Comparison of coding results for TM5, optimal MPEG-like coder and optimal MCWC approach for *Football* at 0.92 Mb/s: (a) bits allocated to MVs; (b) PSNR; (c) total rate.



(a) (b)

Fig. 10. Comparison of coding results for *Football* as a function of rate using TM5, the optimal MPEG-like coder, and the optimal MCWC. (a) Average number of bits allocated to MVs. (b) Average percentage of each frame coded in "skip" mode.

the resulting video achieved higher quality both subjectively and objectively than that of TM5. This demonstrates that joint optimization is more efficient in allocating the bits across the frame.

Fig. 11 shows an example of the resulting quadtree, MVs, and coding modes for the 29th frame in *Football* coded using 27 693 bits using the MCWC. Notice that small blocks are used around the moving areas to provide better motion prediction.

The R-D optimized approaches are robust with respect to the quality of the intra frames. As shown in Fig. 9(b), the inter-frame PSNR remains relatively consistent even when the coded intra-frame quality varies. This observation justifies the flat rate for

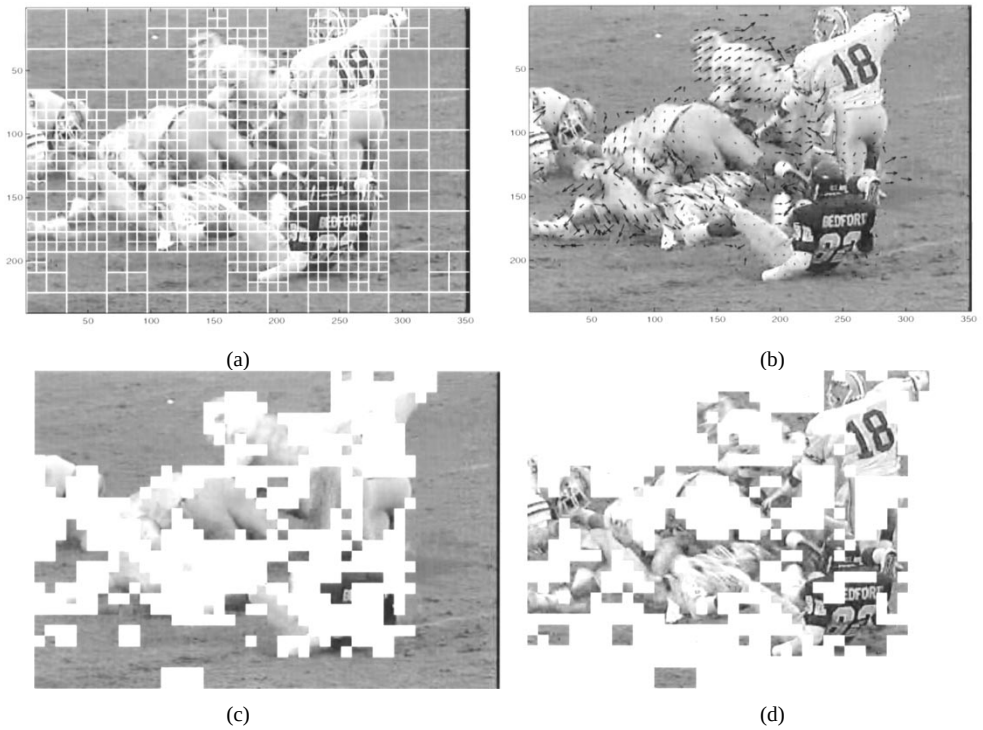


Fig. 11. Coding results for frame 29 of *Football* (coded using 27 693 bits) using the optimal MCWC coder. (a) Quadtree structure. (b) MVs. (c) Skip-mode blocks. (d) Inter-mode blocks.

the inter-frame coding, which greatly simplifies the frame-rate allocation.

Several simplifications and assumptions are considered to further reduce the computation complexity, and comparisons are made with the generalized algorithm. One way to simplify the Lagrange cost generation is to optimize the MV separately with mode (and MQUANT for the MPEG-like coder). The optimal quadtree and associated MVs are first found using a multilevel trellis. Each state in the multilevel trellis represents an MV choice. The rate and distortion of the node are the MV rate R_{MV} and the motion prediction error, respectively. Once the quadtree along with the MVs are found, other parameters such as mode or quantization step size for each node can be found by minimizing the Lagrange cost with the given MV using a single level trellis. Experiments on three sequences show about 0.1–0.2-dB degradation compared to the joint optimization with as much as 0.4-dB decrease for an individual frame.

The computational cost can also be greatly reduced if (5) is assumed, i.e., the rate and distortion for a given node are independent of each other. The optimal solution satisfies the equal slope condition. A DP solution is not necessary in this case since the parameters can be determined independently. Experiments were performed to find the optimal quadtree along with the block coding parameters; the final coding process codes the parameters (MV, MQUANT, DC coefficient) differentially as the general algorithm does. Results demonstrate an average of 0.1–0.3-dB loss with higher losses for lower bit rates. This suggests that the reduction of the overhead bit rate is more crucial for lower bit rates than for higher bit rates.

In order to reduce the computation cost of iterative search for λ, a simplification can be made using a constant λ through the GOP, where an iterative search is performed only once to find the λ of the first inter frame in a GOP. This is based on the observation that the optimal λ is quite consistent for a relatively stationary sequence. This approach can be directly applied to the wavelet coder but not to the MPEG-like coder since the rate constraint usually cannot be satisfied. Experiments with the three sequences using the wavelet coder show very little performance degradation, averaging less than 0.1 dB.

Simulations were performed on a 266-MHz Pentium II PC, with no special speed optimization of the code. The MPEG-like coder requires 3-4 min/frame, with most of the computation going toward the cost generation process where the actual coding is performed to generate real rate and distortion data. For the MCWC, the optimization process requires 40 s to 1 min/frame using the model-based R-D estimation. The embedded DFD coding requires 2–3 min/frame depending on the rate. This limited speed is because of the update of the context model and the R-D slopes after every coefficient is coded. The probability estimation table of the QM-coder could be used instead to speed up the process.

VII. Conclusion

This paper addresses the jointly R-D optimal selection of the coding parameters in a motion-compensated video coder. The general problem is formulated using the Lagrange multiplier method and solved using DP. The general framework is applied to both an MPEG-like coder and to an MCWC. The proposed optimization approaches can fine tune the rate or distortion and thus follow any bit rate or distortion profile.

Simulation results demonstrated that R-D optimized coders provide significant gains both objectively and subjectively

over TM5. An optimal MPEG compliant coder, that is, using the fixed block size as in TM5 but with joint optimization of motion estimation, mode selection, and quantization, can achieve 0.5–0.7-dB gain over TM5. Using VBSME gives additional 0.2–0.6-dB gain. Moreover, VBSME allows the adaptive motion field resolution which is able to give better motion prediction than the fixed block size, thus improving the visual quality, especially along the moving edges.

The R-D calculation is dependent on the specific coder. In the case of the model-based R-D estimation, more accurate estimation gives better bit allocation and thus better coding performance. Although the exponential model under a Gaussian assumption gives reasonably good results in experiments with the MCWC, it does not take into consideration the transform coding gain, and thus tends to overestimate the distortion for a given rate. A more accurate R-D model for estimation is currently an ongoing research topic.

The optimized approaches are developed in the framework of block-based motion estimation and mode classification. The mode selection is general in the sense that the mode choices can be but are not limited to intra, inter, or skipped. Also, block-based object or perceptual classification can be incorporated into the optimization process. This can be achieved by performing the classification on each frame beforehand and then incorporating it into the optimization process using weighted distortion to achieve perceptual bit allocation [27]. The block-mode selection which is performed during the motion estimation and coding process can be different from the classification. Object or perceptual classification can be used for the optimal perceptual bit allocation which affects the R-D estimation through the weighted distortion measure. The classification does not need to be transmitted to the decoder, whereas the block-mode selection has to be transmitted for the decoding process.

The complexity of the algorithm is prohibitive for real-time applications, but it can serve as a benchmark for evaluating practical solutions in terms of the tradeoffs between the desired performance and the required computational complexity; also, there are many scenarios where time is not critical, such as video storage onto CD-ROM or video library where the encoding process is performed only once and off-line. In that case, the additional complexity can be worth the resulting higher quality.

REFERENCES

[1] *Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbits/s*, ISO/IEC 11 172, Oct. 1993.

[2] *oding of Moving Pictures and Associated Audio*, ISO/IEC 13 818, Nov. 1995.

[3] *Video Codec for Audiovisual Services at $p \times 64$ kbits*, ITU-T Recommendation H.261, Mar. 1993.

[4] *Video Codec for Low Bitrate Communication*, ITU-T Recommendation H.263, May 1996.

[5] *Coded Representation of Picture and Audio Information—MPEG-2 Test Model 5*, ISO-IEC AVC-491, Apr. 1993.

[6] R. Talluri, K. Oehler, T. Bannon, J. D. Courney, A. Das, and J. Liao, "A robust, scalable, object-based video compression technique for very low bit-rate coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 221–233, Feb. 1997.

[7] K. Oehler, "Region-based wavelet compression for very low bitrate video coding," in *Proc. Int. Conf. Image Processing*, vol. 2, 1996, pp. 573–576.

[8] F. Moscheni, F. Dufaux, and H. Nicolas, "Entropy criterion for optimal bit allocation between motion and prediction error information," *Proc. SPIE Conf. Visual Communications and Image Processing*, vol. 2094, pp. 235–242, 1993.

[9] B. Girod, "Rate-constrained motion estimation," in *Proc. SPIE Visual Communications and Image Processing Conf.*, vol. 238, 1994, pp. 1026–1034.

[10] G. J. Sullivan and R. L. Baker, "Rate-distortion optimized motion compensation for video compression using fixed or variable block size blocks," in *Proc. 1991 IEEE Global Telecommunications Conf.*, vol. 1, 1991, pp. 85–90.

[11] A. Ortega and K. Ramchandran, "Forward-adaptive quantization with optimal overhead cost for image and video coding with applications to MPEG video coders," *Proc. SPIE Digital Video Compression*, vol. 2419, pp. 129–138, Feb. 1995.

[12] T. Wiegand, M. Lightstone, D. Mukherjee, T. G. Campbell, and S. K. Mitra, "Rate-distortion optimized mode selection for very low bit rate video coding and the emerging H.263 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 182–190, Apr. 1996.

[13] D. Mukherjee and S. K. Mitra, "Combined mode selection and macroblock step adaptation for H.263 video encoder," in *Proc. Int. Conf. Image Processing*, vol. 2, Aug. 1997, pp. 37–40.

[14] H. Sun, W. Kwok, M. Chien, and C. H. J. Ju, "MPEG coding performance improvement by jointly optimizing coding mode decisions and rate control," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 449–458, Apr. 1997.

[15] J. Lee, "Optimal quadtree for variable block size motion estimation," in *Proc. Int. Conf. Image Processing*, vol. 3, Oct. 1995, pp. 480–483.

[16] G. M. Schuster and A. K. Katsaggelos, "A video compression scheme with optimal bit allocation among segmentation, motion, and residual error," *IEEE Trans. Image Processing*, vol. 6, pp. 1487–1501, Nov. 1997.

[17] M. C. Chen and A. N. Wilson Jr., "Rate-distortion optimal motion estimation algorithms for motion-compensated transform video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 147–158, Apr. 1998.

[18] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Trans. Image Processing*, vol. 3, pp. 533–545, Sept. 1994.

[19] G. Keesman, I. Shah, and R. Klein-Gunnewiek, "Bit-rate control for MPEG encoders," *Signal Process.: Image Commun.*, vol. 6, pp. 545–560, Feb. 1995.

[20] P. Cheng, J. Li, and C.-C. J. Kuo, "Rate control for an embedded wavelet video coder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 696–701, Aug. 1997.

[21] L. Jin and A. Ortega, "Bit-rate control using piecewise approximated rate-distortion characteristics," *IEEE Trans. Circuits Syst. Video Techol.*, vol. 8, pp. 446–459, Aug. 1998.

[22] H. M. Hang and J. J. Chen, "Source model for transform video coder and its application—Part I: Fundamental theory," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 287–298, Apr. 1997.

[23] S. Y. Hu, M. C. Chen, and A. N. Willson Jr., "A fast rate-distortion optimization algorithm for motion compensated video coding," in *Proc. 1997 IEEE Int. Symp. Circuits and Systems*, vol. 2, June 1997, pp. 1349–1352.

[24] W. Chung, F. Kossentini, and M. J. T. Smith, "An efficient motion estimation technique based on a rate-distortion criterion," in *Proc. 1996 IEEE ICASSP*, vol. 4, May 1996, pp. 1926–1929.

[25] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Accoust, Speech, Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.

[26] D. P. Bertsekas, *Dynamic Programming: Deterministic and Stochastic Models*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[27] Y. Yang and S. S. Hemami, "Rate-distortion-based combined motion estimation and segmentation," in *Proc. Int. Conf. Image Processing*, vol. 3, 1998, pp. 920–924.

[28] N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[29] J. Li and S. Lei, "An embedded still image coder with rate-distortion optimization," in *Proc. Visual Commun. and Image Processing*, vol. 3309, San Jose, CA, Jan. 1998, pp. 36–48.

[30] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York: Van Nostrand, 1993.

[31] Y. Yang and S. S. Hemami, "Rate-constrained variable block size motion estimation and perceptual coding," in *Proc. Int. Conf. Image Processing*, vol. 1, 1997, pp. 81–84.

Yan Yang received the B.S. degree from Tsinghua University, Beijing, China, in 1993, the M.S. degree from Tufts University, Medford, MA, in 1995, and the Ph.D. degree from Cornell University, Ithaca, NY, in 1999, all in electrical engineering.

She was a summer intern at Sarnoff Corporation, Princeton, NJ, in 1997. Since 1999, she has been with Aware Inc., Bedford, MA. Her current research interests include broadband communications, and video and image compression and transmission.

Dr. Yang is a member of Eta Kappa Nu.

Sheila S. Hemami (S'89–M'95) received the B.S. degree (*summa cum laude*) in electrical engineering from The University of Michigan, Ann Arbor, in 1990, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1992 and 1994, respectively. At Stanford University, she held a National Science Foundation Graduate Fellowship.

She was with Hewlett-Packard Laboratories, Palo Alto, CA, during 1994. In 1995, she joined the faculty of the School of Electrical Engineering, Cornell University, Ithaca, NY, where she is currently an Assistant Professor and the Kodak Term Professor of Electrical Engineering.

Dr. Hemami received a National Science Foundation Early Career Development Award in 1997 and has received numerous teaching awards.