Efficient Cost Measures for Motion Estimation at Low Bit Rates

Dzung T. Hoang

Consumer A/V/D Business Unit
Sony Semiconductor Company of America
3300 Zanker Road, MS SJ3C3
San Jose, CA 95134
dzung@ssa-de.sel.sony.com

Philip M. Long

ISCS Department
National University of Singapore
Singapore 119260
Republic of Singapore
plong@iscs.nus.edu.sg

Jeffrey Scott Vitter

Dept. of Computer Science
Duke University, Box 90129
Durham, NC 27708–0129
jsv@cs.duke.edu

Abstract— **We present and compare methods for choosing motion vectors for block-based motion-compensated video coding. The primary focus is on videophone and videoconferencing applications, where low bit rates are necessary, where motion is usually limited, and where the amount of computation is also limited. In a typical block-based motion-compensated video coding system, motion vectors are transmitted along with a lossy encoding of the residuals. As the bit rate decreases, the proportion required to transmit the motion vectors increases. We provide experimental evidence that choosing motion vectors explicitly to minimize rate (including motion vector coding), subject to implicit constraints on distortion, yields better rate-distortion tradeoffs than minimizing some measure of prediction error. Minimizing a combination of rate and distortion yields further improvements. Although these explicit-minimization schemes are computationally intensive, they provide invaluable insight which we use to develop practical algorithms. We show that minimizing a simple heuristic function of the prediction error and the motion vector codelength results in rate-distortion performance comparable to explicit-minimization schemes while being computationally feasible. Experimental results are provided for coders that operate within the H.261 standard.**

I. Introduction

Hybrid video coding that combines block-matching motion compensation (BMMC) with transform coding of the residual is a popular scheme for video compression, adopted by international standards such as H.261 [1], [2], H.263 [3], and the MPEG standards [4], [5], [6]. Motion compensation is a technique that exploits the typically strong correlation between successive frames of a video sequence by coding *motion vectors* that tell the decoder where to look on the previous frame for predictions of the intensity of each pixel in the current frame. With BMMC, the current frame is divided into blocks (usually 8×8 or 16×16) whose pixels are assigned the same motion vector $\vec{v}$. The residual from motion compensation is then coded with a lossy transform coder, such as the 2D-DCT, followed by a variable-length entropy coder.

In previous work on BMMC, motion vectors are typically chosen to minimize prediction error, and much of the emphasis has been on speeding up the motion search [7], [8], [9], [10]. However, for low bit-rate applications, such as videophone and videoconferencing, the coding of motion vectors takes up a significant portion of the bandwidth, as evidenced with a coding experiment summarized in Figure 1. This observation has previously been made in [11]. In this paper, we investigate cost measures that take into account the effects of the choice of motion vector on rate and distortion. We first develop and present computationally intensive coders that attempt explicitly to optimize for rate and distortion. Insights gained from these implementations lead to the development of faster coders that minimize an efficiently computed heuristic function. Experiments show that using these measures yields substantially better rate-distortion performance than standard measures based solely upon prediction error.

Fig. 1. Distribution of bits for intraframe coding of the Miss America sequence at various bit rates with a standard $p \times 64$ coder.

We implemented and tested our motion estimation algorithms within the H.261 standard, also known informally as the $p \times 64$ standard. The $p \times 64$ standard is intended for applications like videophone and videoconferencing, where low bit rates are required, not much motion is present, and frames are to be transmitted essentially as they are generated. Our experimental results are for benchmark videos typical of the type for which the $p \times 64$ standard was intended: they consist of a "head-and-shoulders" view of a single speaker.

In the next section, we briefly describe an existing implementation of the $p \times 64$ standard that we use as a basis for comparison. We then show how to modify the base implementation, but remain within the $p \times 64$ standard, to choose motion vectors that more directly minimize rate and distortion. Experiments show that when transmitting two benchmark QCIF video sequences, `Miss America` and `Claire`, at 18 kbits/sec using rate control, choosing motion vectors explicitly to minimize rate improves average

PSNR by 0.87 dB and 0.47 dB respectively. In the $p \times 64$ standard, two binary coding decisions must be made from time to time.[1] In the base implementation, heuristics based upon prediction error are used to make these decisions. When bit minimization is also applied to make the coding decisions, the improvement in PSNR becomes a significant 1.93 dB for `Miss America` and 1.35 dB for `Claire`. If instead of minimizing the bit rate, we minimize a combination of rate and distortion, we observe improvements of 2.09 dB and 1.45 dB, respectively.

In Section IV, we describe coders that minimize a heuristic function of the prediction error and motion vector code-length. These heuristic coders give compression performance comparable to the explicit minimization coders while running much faster. Experimental results are presented in Sections III-D and IV-B.

Preliminary descriptions of this work can be found in [12], [13], [14], [15].

II. PVRG Implementation of H.261

As a basis for comparing the different motion estimation schemes proposed in this chapter, we use the $p \times 64$ coder supplied by the Portable Video Research Group (PVRG).[2] In the base PVRG implementation, a motion vector $\vec{v}$ is determined for each macroblock M using standard full-search block-matching. Only the luminance blocks are compared to the determine the best match, with the mean absolute difference (MAD) being used as the measure of prediction error. Decisions on how to code individual blocks are made according to Reference Model 8 [16].

III. Explicit Minimization Algorithms

In the PVRG coder, motion estimation is performed to minimize the MAD of the prediction error. A rationale for this is that minimizing the mean square error (MSE) of the motion-compensated prediction, which is approximated with the MAD, is equivalent to minimizing the variance of the 2D-DCT coefficients of the prediction error, which tends to result in more coefficients being quantized to zero. However, minimizing the variance of the DCT coefficients does not necessarily lead to a minimum-length encoding of the quantized coefficients, especially since the quantized coefficients are then Huffman and run-length coded. Furthermore, since coding decisions are typically made independently of motion estimation, the effect of motion estimation on rate is further made indirect.

In this section, we describe two algorithms that perform motion estimation explicitly to minimize rate and a third algorithm that minimizes a combination of rate and distortion. We then present results of experiments that compare these algorithms with the standard motion estimation algorithm used by the PVRG coder.

[1] These are 1) whether to use motion compensation and 2) whether to use the loop filter with motion compensation.

[2] As of the publication date, the source code for this implementation can be obtained via anonymous ftp from `havefun.stanford.edu`.

A. Algorithm M1

In Algorithm M1, motion estimation is performed explicitly to minimize (locally) the code-length of each macroblock. The decisions of whether to use motion compensation and whether to use the loop filter are made in the same way as in the PVRG implementation. We invoke the appropriate encoding subroutines for each choice of motion vector within the search area, picking the motion vector that results in the minimum code-length for the entire macroblock. The computed code-length includes the coding of the transform coefficients for the luminance blocks,[3] the motion vector, and all other side information. When choosing the motion vector to minimize the coding of the current macroblock, we use the fact that the motion vectors for previous macroblocks (in scan order) have been determined in order to compute the code-length. However, since the choice of a motion vector for the current macroblock affects the code-length of future macroblocks, this is a greedy minimization procedure which may not result in a globally minimal code-length.

B. Algorithm M2

Algorithm M2 differs from Algorithm M1 in that the decisions of whether to use motion compensation and the loop filter are also made to minimize rate: all three combinations of the decisions are tried, and the one resulting in the minimum code-length is used. Since M2 is able to make decisions on how to code each macroblock, it is able to take into account the coding of side information in minimizing the rate. For low bit rates, where the percentage of side information is significant compared to the coding of motion vectors and transform coefficients, we would expect M2 to be effective in reducing the code-length of side information.

C. Algorithm RD

With Algorithms M1 and M2, we minimize rate without regard to distortion and then choose the quantization step size to achieve the desired distortion level. This is not always the best policy. There may be cases where the choice of motion vector and coding decisions that minimize rate results in a relatively high distortion, whereas another choice would have a slightly higher rate but substantially lower distortion. In terms of rate-distortion tradeoff, the second choice may be better. Since the ultimate goal is better rate-distortion performance, we expect further improvements if we minimize a combination of rate and distortion. M1 and M2 call encoder routines in the minimization steps. By adding calls to decoder routines, we can compute the resulting distortion. We incorporate this idea into Algorithm RD.

Algorithm RD minimizes a linear combination of rate and distortion. Let $B(\vec{v}, \vec{c})$ denote the number of bits to code the current macroblock using motion vector $\vec{v}$ and

[3] The transform coding of the chrominance blocks could be included as well. However, we chose not to do so in order to make a fair comparison to the base PVRG coder. This is also the policy for the other coders described in this chapter.

coding decisions $\vec{c}$. Similarly, let $D(\vec{v},\vec{c})$ be the resulting mean squared error. RD minimizes the objective function:

$$C_{\mathrm{RD}}(\vec{v},\vec{c}) = B(\vec{v},\vec{c}) + \lambda D(\vec{v},\vec{c}). \quad (1)$$

If $B(\vec{v},\vec{c})$ and $D(\vec{v},\vec{c})$ for each block were independent of the choices of $\vec{v}$ and $\vec{c}$ for previously coded blocks, results of Shoham and Gersho [17] imply that an objective function of the form (1) would minimize distortion subject to a rate constraint. Since in $p \times 64$ a motion vector is coded with reference to a previously coded motion vector, there is some dependence at the macroblock level. Therefore, minimizing (1) locally for each block is globally suboptimal in the rate-distortion sense. With this caveat noted, we proceed as in [17].

In principle, we should choose λ based upon the theoretical rate-distortion curve for the input video. A good choice is to set λ to be equal to the negative of the slope of the line tangent to the distortion vs. rate curve at the desired operating point. This way we are optimizing in a direction perpendicular to the rate-distortion curve at the operating point. The rate-distortion curve can be estimated, for example, by preprocessing a portion of the input video. An online iterative search method could also be used [17]. In our experiments, we code the test sequence several times with different quantizer step sizes to estimate the rate-distortion function, and fix λ based upon the slope of the function at the desired rate. Our purpose is to explore the performance improvement offered by such an approach.

D. Experimental Results

For our experiments, we coded 49 frames of the `Miss America` sequence and 30 frames of the `Claire` sequence, both in QCIF format sampled at 10 frames/sec. These are "head and shoulders" sequences typical of the type found in videophone and videoconferencing applications. We present results here for coding at 18 kbits/sec using the rate controller outlined in Reference Model 8. The average PSNR for each coded frame is plotted for the `Miss America` and `Claire` sequences in Figure 2. The average PSNR for inter-coded frames are tabulated in Table I. For each sequence, all the coders used the same quantization step size for the initial intra-coded frame.

IV. Heuristic Algorithms

While Algorithms M1, M2, and RD generally exhibit better rate-distortion performance than the base PVRG coder, they are computationally expensive. The additional computation is in the explicit evaluation of the rate (and distortion in the case of RD). To reduce the computational complexity, we propose to minimize an efficiently computed model of rate and distortion. The idea is that the prediction error (MSE, MAD, or similar measure) can be used to estimate the rate and distortion for transform coding. This estimate is then combined with the motion vector codelength, which is readily available with a table lookup. We develop such a cost function below and use it in two heuristic coders H1 and H2 that are analogous to the explicit minimization coders M1 and M2. Both H1 and H2 choose motion vectors to minimize the cost function. However, H1 makes coding decisions using the same decision functions that the PVRG and M1 coders use, while H2 chooses the coding control that minimizes the coding rate given the estimated motion vectors. Since H2 has to try out three coding control choices, it will be about three times slower than H1. However, H2 gives us an indication of the performance that is achievable by improving the coding control. Also, H2 is easily parallelized, using duplicated hardware for example.

A. Heuristic Cost Function

Let $\vec{E}(\vec{v})$ denote a measure of the prediction error that results from using motion vector $\vec{v}$ to code the current macroblock. For example, the error measure could be defined as $\vec{E}(\vec{v}) = \langle \mathrm{MAD}(\vec{v}), \mathrm{DC}(\vec{v}) \rangle$, where $\mathrm{MAD}(\vec{v})$ is the mean absolute prediction error and $\mathrm{DC}(\vec{v})$ is the average prediction error. Suppose we have a model $H(\vec{E}(\vec{v}), Q)$ that gives us an estimate of the number of bits needed to code the motion compensation residual, where $\vec{E}(\vec{v})$ is defined above and Q is the quantization step size. We could then combine this estimate with $B(\vec{v})$, the number of bits to code the motion vector $\vec{v}$. The result is a cost function that we can use for motion estimation:

$$C_{\mathrm{H}}(\vec{v},Q) = H(\vec{E}(\vec{v}),Q) + B(\vec{v}). \quad (2)$$

As defined above, the function H provides an estimate of the number of bits needed to code the motion compensation residual with quantizer step size Q. As we will discuss later, it can also be used to estimate a combination of rate and distortion.

The choice of error measure $\vec{E}$ and heuristic function H are parameters to the motion estimation algorithm. In our investigations, we used MAD as the error measure, for computational reasons. We also looked into using the MSE, but this did not give any clear advantages over the MAD. It is also possible to define $\vec{E}$ to be a function of several variables. However, we report only on the use of MAD for $\vec{E}$ and denote $\vec{E}(\vec{v})$ by ξ for convenience, where the dependence upon $\vec{v}$ is implicit. We examined several choices for H and describe them below.

As mentioned above, we can use H to estimate the number of bits used to transform-code the prediction error. To get an idea of what function to use, we gathered experimental data on the relationship between the MAD and DCT coded bits per macroblock for a range of motion vectors. Fixing the quantization step size Q at various values, the data was generated by running the RD coder on two frames of the Miss America sequence and outputting the MAD and DCT coded bits per macroblock for each choice of motion vector. The results are histogrammed and shown as density plots in Figure 3.

These plots suggest the following forms for H:

$$H(\xi) = c_1\xi + c_2 \ , \quad (3)$$

$$H(\xi) = c_1 \log(\xi+1) + c_2, \quad (4)$$

$$H(\xi) = c_1 \log(\xi+1) + c_2\xi + c_3. \quad (5)$$

(a) Miss America (b) Claire

Fig. 2. Comparison of explicit-minimization motion estimation algorithms for coding the Miss America and Claire sequences at 18 kbits/sec.

Fig. 3. Density plots of DCT coding bits vs. MAD prediction error for first inter-coded frame of Miss America sequence at various levels of quantization.

The above forms assume a fixed Q. In general, H also depends upon Q; however, when using H to estimate the motion for a particular macroblock, Q is held constant to either a preset value or to a value determined by the rate control mechanism. We can treat the parameters c_i as functions of Q. Since there is a small number (31) of possible values for Q, we can perform curve fitting for each value of Q and store the parameters in a lookup table.

We can also model the reconstruction distortion as a function of prediction error. We use the RD coder to generate experimental data for distortion versus MAD, shown in Figure 4, and find a similar relationship as existed for bits versus MAD. Again, we can use (3)–(5) to model the distortion. As with the RD coder, we can consider jointly optimizing the heuristic estimates of rate and distortion with the following cost function:

$$C_{\mathrm{H}}(\vec{v}, Q) = B(\vec{v}) + H_{\mathrm{R}}(\xi, Q) + \lambda H_{\mathrm{D}}(\xi, Q), \qquad (6)$$

where H_{R} is the model for rate and H_{D} is the model for distortion.

If we use one of (3)–(5) for both H_{R} and H_{D}, the combined heuristic function, $H = H_{\mathrm{R}} + \lambda H_{\mathrm{D}}$, would have the same form as H_{R} and H_{D}. Therefore, we can interpret the heuristic as modeling a combined rate-distortion function. In this case, we can perform curve fitting once for the combined heuristic function by training on the statistic $R + \lambda D$, where R is the DCT bits for a macroblock and D is the reconstruction distortion for the macroblock. As with Algorithm RD, the parameter λ can be determined from the operational rate-distortion curve, for example.

B. Experimental Results

To test the H1 and H2 coders, we initially used the same test sequences and followed the procedures outlined in Section III-D and present results for coding at 18 kbits/sec using the buffer-feedback rate controller specified in RM8. In the next section, we verify these results with experiments on eight different test sequences.

B.1 Static Cost Function

Here, we present results using a static set of coefficients. To determine the coefficients for the heuristic functions, we performed linear least squares regression, fitting data generated by the RD coder to the $R + \lambda D$ statistic, as discussed earlier. A set of regression coefficients are stored in a lookup table, indexed by the quantizer step size Q. We

Fig. 4. Density plots of MSE reconstruction distortion vs. MAD prediction error for first inter-coded frame of Miss America sequence at various levels of quantization.

tested the different forms for the heuristic function given in (3)–(5). Comparative plots of the resulting PSNR are shown in Figures 5 and 6. The average PSNR for coding at 18 kbits/sec is tabulated in Table I. These results show that the heuristic coders perform comparably to the explicit minimization coders. In particular, the heuristic coders seem more robust than M1 and M2, most likely because the heuristic functions correlate well with both rate and distortion, whereas M1 and M2 only consider rate.

B.2 Adaptive Cost Function

The above results rely on pre-training the model parameters c_i for each value of Q for each video sequence. This is a tedious and time-consuming operation. Instead, we can use an adaptive on-line technique, such as the Widrow-Hoff learning rule [18], [19], to train the model parameters. (Despite its simplicity, the Widrow-Hoff rule has attractive theoretical properties [20], [21].) The training examples could be generated each time we encode a macroblock using motion compensation mode. However, we cannot possibly hope to train one model for each value of Q simply because there would not be enough training examples. We need a single model whose parameters are independent of Q. The curve fitting results from the pre-training trials show a strong correlation between the model parameters and Q^{-1}. This agrees well with previous work on rate-quantization modeling [22]. Therefore we propose the following form for the cost function:

$$H(\xi, Q) = c_1 \frac{\xi}{Q} + c_2. \tag{7}$$

This can be simplified as

$$H(\psi) = c_1 \psi + c_2, \tag{8}$$

where $\psi \equiv \xi/Q$. Since the simple linear model performed well with static cost functions, we do not consider more complex models here.

We conducted experiments using the Widrow-Hoff training rule on the Miss America and Claire sequences. As applied to the current context, the Widrow-Hoff rule is a technique for learning an objective function $f(\psi)$. With $H(\psi)$ as an estimate of $f(\psi)$, the Widrow-Hoff rule gives us a way to adapt the weights c_1 and c_2 of (8) when given ψ and the value of $f(\psi)$. For the experiments, we chose the objective function

$$f(\psi) = R(\psi) + \lambda D(\psi), \tag{9}$$

where $R(\psi)$ is the actual number of bits used to code the DCT coefficients and $D(\psi)$ is the resulting distortion, both of which can be evaluated by invoking encoder routines. Given an initial set of weights c_1 and c_2, a new set of weights c_1' and c_2' can be computed as:

$$c_1' = c_1 + \psi\eta \cdot \frac{f(\psi) - H(\psi)}{\psi^2 + 1}, \tag{10}$$

$$c_2' = c_2 + \eta \cdot \frac{f(\psi) - H(\psi)}{\psi^2 + 1}; \tag{11}$$

where η, the *learning rate*, is a parameter that determines how quickly the weights are adapted.

With the static cost function, we trained and evaluated the heuristic function based on the combined prediction error for the four luminance blocks that make up a macroblock. In order to gather more training examples for the adaptive heuristics, we evaluate and update the heuristic function once for each luminance block. This strategy increased the PSNR slightly at the expense of some extra computation.

In the experiments, the learning rate η was determined in a trial-and-error phase and fixed for both sequences. The parameter λ was also determined by trial-and-error and held constant for both test sequences. Comparative plots of the resulting PSNR are shown in Figures 7 and 8. The average PSNR for coding at 18 kbits/sec is tabulated in Table II. These results show that the adaptive heuristic coders perform comparably to and sometimes better than the static heuristic coders and the explicit minimization coders. Furthermore, the adaptive heuristic coders perform well on both sequences with the same initial parameter values.

As a comparison of visual quality, Frame 27 of the Miss America sequence is decoded and shown in Figure 9 for

(a) Miss America (b) Claire

Fig. 5. Comparison of H1 coder using static heuristic cost function with PVRG and M1 coders. Coding is performed with RM8 rate control at 18 kbits/sec. H1-A, H1-B, and H1-C use the heuristic functions (3), (4), and (5), respectively.

(a) Miss America (b) Claire

Fig. 6. Comparison of H2 coder using static heuristic cost function with PVRG and M2 coders. Coding is performed with RM8 rate control at 18 kbits/sec. H2-A, H2-B, and H2-C use the heuristic functions (3), (4), and (5), respectively.

Sequence	PVRG	M1	M2	RD	H1-A	H1-B	H1-C	H2-A	H2-B	H2-C
Miss America	34.58	35.44	36.51	36.67	35.60	35.72	35.58	36.63	36.77	36.68
Claire	32.77	33.24	34.12	34.22	33.68	33.50	33.60	34.47	34.36	34.39

TABLE I

RESULTS OF STATIC HEURISTIC COST FUNCTION. SHOWN IS AVERAGE PSNR (IN DB) OF INTER-CODED FRAMES FOR CODING TEST SEQUENCES AT 18 KBITS/SEC. H1-A (H2-A), H1-B (H2-B), AND H1-C (H2-C) USE THE HEURISTIC FUNCTIONS (3), (4), AND (5), RESPECTIVELY.

Video	PVRG	M1	M2	RD	H1-A	H1-WH	H2-A	H2-WH
Miss America	34.58	35.44	36.51	36.67	35.60	35.83	36.63	36.84
Claire	32.77	33.24	34.12	34.22	33.68	33.58	34.47	34.51

TABLE II

RESULTS OF ADAPTIVE HEURISTIC COST FUNCTION. SHOWN IS AVERAGE PSNR (IN DB) OF INTER-CODED FRAMES FOR CODING TEST SEQUENCES AT 18 KBITS/SEC. H1-A AND H2-A USE THE HEURISTIC FUNCTION (3) WITH STATIC PARAMETERS. H1-WH AND H2-WH USE ADAPTIVE PARAMETERS.

(a) Miss America (b) Claire

Fig. 7. Comparison of H1 coder using adaptive heuristic cost function with PVRG and M1 coders. Coding is performed with RM8 rate control at 18 kbits/sec. H1-A, H1-B, and H1-C use the heuristic functions (3), (4), and (5), respectively.

(a) Miss America (b) Claire

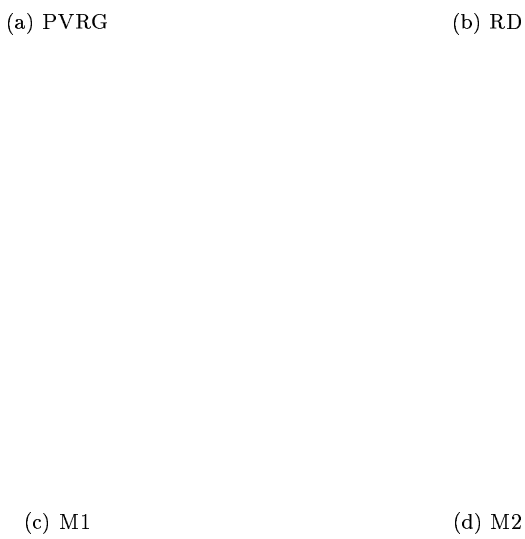
Fig. 8. Comparison of H2 coder using adaptive heuristic cost function with PVRG and M2 coders. Coding is performed with RM8 rate control at 18 kbits/sec. H2-A, H2-B, and H2-C use the heuristic functions (3), (4), and (5), respectively.

the PVRG and explicit-minimization coders and in Figure 10 for the heuristic coders. The motion vector field for the PVRG, RD, adaptive H1, and adaptive H2 coders are shown in Figure 11. Frame 27 was chosen because it is in a difficult scene with much head motion, resulting in more noticeable coding artifacts. The RD and adaptive heuristic coders give smoother motion fields than the reference PVRG coder, especially for the background region. Note also that for the former coders, no motion is indicated for the relatively uniform background *except* following macroblocks with detected foreground motion on the same row. Intuitively, this results in an economical encoding of the motion vectors, which are differentially encoded. Since the background is relatively uniform, coding motion in this area results in relatively small motion compensation residual.

C. Further Experiments

Here, we present results of further experiments to confirm the efficacy of the various motion estimation algorithms operating within the $p \times 64$ standard. We applied the various algorithms to code eight test video sequences without rate control, sweeping the quantization scale from 12 to 31 to determine the operational rate-distortion plots shown in Figure 12. Each test sequence consists of 50 frames in QCIF format coded at 10 frames/sec. The `Miss America` sequence in this test suite was obtained from a different source than the `Miss America` sequence used in the earlier experiments and has different rate-distortion characteristics.

The results show that the adaptive heuristic algorithms perform consistently well compared to the base PVRG and explicit-minimization implementations, though the level of improvement varies among sequences. The anomalies observed in coding the `Grandma` sequence at low rates with



(a) PVRG (b) RD

(c) M1 (d) M2

Fig. 9. Frame 27 of the Miss America sequence as encoded using the PVRG and explicit-minimization motion estimation algorithms. Only the luminance component is shown.

the PVRG and adaptive H1 coders, as evidenced by the steep slope and unevenness in the RD curve, seem to indicate a breakdown of the RM8 coding control heuristics, which were not optimized for operation at very low rates. This conclusion is supported by the lack of such anomalies when bit-minimization is used to perform coding control, as with the M2, H2, and RD coders.

The distributions of bits for coding the Miss America sequence with the H1 and H2 coders are plotted in Figure 13. Compared to Figure 1, these plots show that the H1 and H2 coders both reduce the percentage of bits used for coding motion vectors, while increasing the percentage of bits used to code the DCT coefficients. Furthermore, with the H2 coder, which applies bit-minimization to coding control, the number of bits used for coding side information is also reduced.

V. Related Work

In related work, Chung, Kossentini and Smith [23] consider rate-distortion optimizations for motion estimation in a hybrid video coder based upon subband coding and block-matching motion compensation. The input frames are first decomposed into subbands, which are divided into uniform rectangular blocks. For each block, a Lagrangian cost function is used to select between intraframe and interframe modes and to select between a small number of candidate motion vectors, which are coded with a lossy two-dimensional vector quantizer.

Independent of our work, rate-distortion optimization for motion estimation has been reported in [24]. The authors consider rate-distortion optimization in a dependent-coding environment where motion vectors are coded using DPCM techniques. After first constructing a dependency graph, the Viterbi dynamic programming algorithm is used to find a path that minimizes an additive Lagrangian cost function. Noting the computational complexity of this approach, the authors propose a reduced-complexity algorithm that considers only a small fraction of the possible states for each motion-compensated block. Even so, this reduced-complexity algorithm has a considerable processing and memory overhead associated with the dynamic programming algorithm, which is performed on top of traditional block matching. In comparison, our adaptive heuristic cost function requires minimal overhead

(a) H1-A (b) H2-A

(c) H1-WH (d) H2-WH

Fig. 10. Frame 27 of the Miss America sequence as encoded using the heuristic motion estimation algorithms. Only the luminance component is shown.

over block matching.

In [25], rate-distortion optimization is applied to the selection of coding control for low-bit-rate video coding under the H.263 standard, a newer standard than the H.261 standard that we consider here. A greedy optimization strategy is adopted to avoid the exponential complexity that a global optimization would entail. Limited dependencies between the coding control of neighboring blocks is considered and the coding control is computed using the Viterbi algorithm to minimize a Lagrangian cost function. Even with simplifying assumptions, the rate-distortion optimization is computationally complex and may not be suitable for real-time implementation, as the authors readily admit.

Ribas-Corbera and Neuhoff [26] describe a procedure for minimizing rate in a lossless motion-compensated video coder. They explore the allocation of bits between the coding of motion vectors and the coding of prediction error. They assume that the prediction error has a discrete Laplacian distribution and derive an expression for the total rate as a function of the number of bits allocated to code the motion vectors. It is not clear whether this work can be extended to lossy coding since distortion is not taken into account in the formulation.

A linear relationship between MAD and both rate and distortion has been independently observed in [27]. The authors mention the possibility of performing motion vector search to minimize the bit rate, but conclude that just minimizing MAD would have a similar effect.

VI. Discussion

We have demonstrated that, at low bit rates, choosing motion vectors to minimize an efficiently computed heuristic cost function gives substantially better rate-distortion performance than the conventional approach of minimizing prediction error. Furthermore, by adapting the heuristic function to the input sequence, we are able to achieve coding performance comparable to more computationally expensive coders that explicitly minimize rate or a combination of rate and distortion.

In the experiments, full-search block-matching was employed by all the coders. Our fast heuristic coders are also compatible with 2D logarithmic and many other reduced-search motion estimation techniques. Furthermore, since the heuristic cost function factors in the motion vector

(a) PVRG (b) RD

(c) H1-WH (d) H2-WH

Fig. 11. Estimated motion vectors for frame 27 of the Miss America sequence for the PVRG, RD, H1-WH, and H2-WH coders.

code-length, the cost function has a strong monotonic component and is better suited for the reduced-search techniques that assume monotonicity in the cost function.

We have considered only the simple case of using a fixed parameter λ to trade rate and distortion. An online adaptation of λ to track variations in the input sequence is certainly possible and would result in more robust coders. On the other hand, we observed that the behavior of these algorithms is quite robust with respect to moderate variations in λ, and that, for example, the best setting of λ for one test sequence worked well when used for the other. Thus, it seems that fixing λ is safe in practice. Still, since λ influences rate to some extent, it can be used in conjunction with the quantization step size in performing rate control. Automatic control of λ based upon buffer feedback as described in [28] is a possibility.

Although the methods presented here have been implemented within the H.261 standard, they should be applicable to any video coder that employs motion compensation in a low-bit-rate setting. In particular, the H.263 standard is similar enough to H.261 that it seems clear that these methods will work well with H.263. As a case in point, the bit-minimization strategy has been applied in [12] within a non-standard quadtree-based coder that chooses motion vectors to optimize a hierarchical encoding of the motion information within a block-matching framework with variable block sizes.

VII. Acknowledgements

Support for this work was provided in part by Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grants F49620–92–J–0515 and F49620–94–1–0217, and by Army Research Office grant DAAH04–93–G–0076. In addition, the first author was supported in part by a National Science Foundation Graduate Fellowship, and the third author was supported in part by an associate membership in CESDIS.

References

[1] CCITT, "Video codec for audiovisual services at $p \times 64$ kbit/s", Aug. 1990, Study Group XV—Report R 37.

[2] M. Liou, "Overview of the $p \times 64$ kbit/s video coding standard", *Communications of the ACM*, vol. 34, no. 4, pp. 60–63, Apr. 1991.

[3] ITU-T Study Group 15, "Draft recommendation H.263 (Video coding for narrow telecommunication channels)", April 26 1995, Document LBC-95-163.

[4] ISO, "Cd11172-2: Coding of moving pictures and associated

(a) Carphone

(b) Claire

(c) Foreman

(d) Grandma

(e) Miss America

(f) Mother and Daughter

(g) Suzie

(h) Trevor

Fig. 12. Performance of motion estimation algorithms on eight test sequences.

(a) H1 Coder (b) H2 Coder

Fig. 13. Distribution of bits for coding the Miss America sequence with the adaptive heuristic coder.

audio for digital storage media at up to about 1.5 mbits/s", Nov. 1991.

[5] D. J. LeGall, "MPEG: A video compression standard for multimedia applications", *Communications of the ACM*, vol. 34, no. 4, pp. 46–58, Apr. 1991.

[6] ISO-IEC/JTC1/SC29/WG11/N0802, "Generic coding of moving pictures and associated audio information: Video", Nov. 1994, MPEG Draft Recommendation ITU-T H.262, ISO/IEC 13818-2.

[7] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe coding", *IEEE Transactions on Communications*, vol. COM-29, no. 12, pp. 1799–1808, 1981.

[8] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion-compensated interframe coding for video conferencing", in *Proceedings IEEE National Telecommunication Conference*, Nov. 1981, vol. 4, pp. G5.3.1–G5.3.5.

[9] A. Puri, H.-M. Hang, and D. L. Schilling, "An efficient block-matching algorithm for motion compensated coding", in *Proceedings 1987 International Conference on Acoustics, Speech and Signal Processing*, 1987, pp. 25.4.1–25.4.4.

[10] R. Srinivasan and K. R. Rao, "Predictive coding based on efficient motion estimation", in *Proceedings International Conference on Communications*, 1988, vol. 1, pp. 521–526.

[11] H. Li, A. Lundmark, and R. Forchheimer, "Image sequence coding at very low bitrates: A review", *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 589–609, Sept. 1994.

[12] D. T. Hoang, P. M. Long, and J. S. Vitter, "Explicit bit-minimization for motion-compensated video coding", in *Proceedings 1994 Data Compression Conference*, Snowbird, UT, Mar. 1994, pp. 175–184, IEEE Computer Society Press.

[13] D. T. Hoang, P. M. Long, and J. S. Vitter, "Rate-distortion optimizations for motion estimation in low-bitrate video coding", Tech. Rep. CS-1995-16, Duke University, Dept. of Computer Science, 1995.

[14] D. T. Hoang, P. M. Long, and J. S. Vitter, "Rate-distortion optimizations for motion estimation in low-bit-rate video coding", in *Digital Video Compression: Algorithms and Technologies 1996*, V. Bhaskaran, F. Sijstermans, and S. Panchanathan, Eds., 1996, pp. 18–27, Proc. SPIE 2668.

[15] D. T. Hoang, P. M. Long, and J. S. Vitter, "Efficient cost measures for motion compensation at low bit rates", in *Proceedings 1996 Data Compression Conference*, Snowbird, Utah, Mar. 1996, pp. 102–111.

[16] CCITT, "Description of reference model 8 (RM8)", June 1989, Study Group XV—Document 525.

[17] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 9, pp. 1445–1453, Sept. 1988.

[18] B. Widrow and M. E. Hoff, "Adaptive switching circuits", in *1960 IRE WESCON Convention Record*, 1960, vol. 4, pp. 96–104.

[19] J. Hertz, A. Krogh, and R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, Redwood City, CA, 1991.

[20] B. Widrow and S. D. Stearns, *Adaptive signal processing*, Prentice-Hall, 1985.

[21] N. Cesa-Bianchi, P.M. Long, and M.K. Warmuth, "Worst-case quadratic loss bounds for prediction using linear functions an d gradient descent", *IEEE Transactions on Neural Networks*, vol. 7, no. 3, pp. 604–619, 1996.

[22] W. Ding and B. Liu, "Rate control of MPEG video coding and recording by rate-quantization modeling", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 1, pp. 12–20, Feb. 1996.

[23] W. C. Chung, F. Kossentini, and M. J. T. Smith, "A new approach to scalable video coding", in *Proceedings 1995 Data Compression Conference*, Snowbird, UT, Mar. 1995, pp. 381–390, IEEE Computer Society Press.

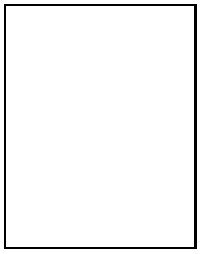
[24] M. C. Chen and Jr. A. N. Willson, "Rate-distortion optimal motion estimation algorithm for video coding", in *Proceedings 1996 International Conference on Acoustics, Speech and Signal Processing*, Atlanta, GA, May 1996, vol. 4, pp. 2096–2099.

[25] 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 Transactions on Circuits and Systems for Video Technology*, vol. 6, no. 2, pp. 182–190, Apr. 1996.

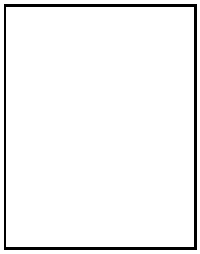
[26] J. Ribas-Corbera and D. L. Neuhoff, "Optimal bit allocations for lossless video coders: Motion vectors vs. difference frames", in *Proceedings ICIP'95*, 1995, vol. 3, pp. 180–183.

[27] J. L. Mitchell, W. B. Pennebaker, C. E. Fogg, and D. J. LeGall, Eds., *MPEG Video Compression Standard*, Chapman & Hall, New York, NY, 1997.

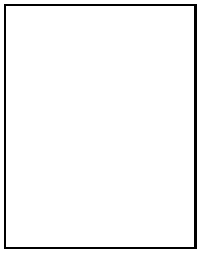
[28] J. Choi and D. Park, "A stable feedback control of the buffer state using the controlled lagrange multiplier method", *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 546–557, Sept. 1994.



Dzung T. Hoang received B.S. degrees (summa cum laude) in electrical engineering and computer science from Tulane University in 1990. He received the M.S. and Ph.D. degrees in computer science from Brown University in 1992 and 1997, respectively. He is currently a Senior Software Systems Engineer at Sony Semiconductor Company of America, San Jose, California. His research interests include video and image processing, with applications to motion estimation, rate control, and digital television. Dr. Hoang is a member of the IEEE, IEEE Computer Society, and the ACM.



Philip M. Long got a B.A. from Oberlin College in 1987, and a Ph.D. from the University of California at Santa Cruz in 1992. He was a postdoc at the Graz University of Technology during 1992-93, and at Duke University during 1993-95. He was at Research Triangle Institute during 1995, then returned to Duke to do a postdoc during 1995-6. He is currently a Lecturer at the National University of Singapore. His research interests are computational learning theory and data compression.



Jeffrey Scott Vitter is the Gilbert, Louis, and Edward Lehrman Professor and Chair of the Department of Computer Science at Duke University, where he joined the faculty in January 1993. He also serves as Co-Director of the Center for Geometric Computing. Previously he was Professor of Computer Science at Brown University. He received a Ph. D. in computer science from Stanford University in 1980 and a B.S. in mathematics with highest honors from the University of Notre Dame in 1977. Prof. Vitter is a Guggenheim Fellow, an ACM Fellow, an IEEE Fellow, an NSF Presidential Young Investigator, and an IBM Faculty Development Awardee. He is coauthor of the book *Design and Analysis of Coalesced Hashing* and is coholder of patents in the areas of external sorting, prediction, and approximate data structures. He serves or has served on the editorial boards of *Algorithmica*, *Communications of the ACM*, *IEEE Transactions on Computers*, *SIAM Journal on Computing*, and *Theory of Computing Systems*. His research interests include the design and mathematical analysis of algorithms, I/O efficiency and external memory algorithms, parallel computation, incremental (online) algorithms, computational geometry, data compression, data mining, machine learning, and order statistics.