

An Efficient Mode Selection Prior to the Actual Encoding for H.264/AVC Encoder

Manoranjan Paul, *Member, IEEE*, Michael R. Frater, *Member, IEEE*, and John F. Arnold, *Senior Member, IEEE*

Abstract—Many video compression algorithms require decisions to be made to select between different coding modes. In the case of H.264, this includes decisions about whether or not motion compensation is used, and the block size to be used for motion compensation. It has been proposed that constrained optimization techniques, such as the method of Lagrange multipliers, can be used to trade off between the quality of the compressed video and the bit rate generated. In this paper, we show that in many cases of practical interest, very similar results can be achieved with much simpler optimizations. Mode selection by simply minimizing the distortion with motion vectors and header information produces very similar performance to the full constrained optimization, while it reduces the mode selection and over all encoding time by 31% and 12%, respectively. The proposed approach can be applied together with fast motion search algorithms and the mode filtering algorithms for further speed up.

Index Terms—H.264 mode selection, motion estimation, teleconferencing, video coding.

I. INTRODUCTION

THE variable block size motion estimation and compensation in H.264/AVC is one of the most significant innovations when compared to previous video coding standards. The block sizes (popularly known as modes) are chosen in the range from 16×16 to 4×4 pixels. Choosing larger partition sizes (16×16 , 16×8 , 8×16) requires a smaller number of bits to encode the motion vectors and the type of partition, at the expense of containing a significant amount of energy in motion compensated residual error in areas of high detail. By contrast, choosing smaller partition sizes (8×4 , 4×8 , 4×4) may result in a lower number of bits for encoding the residual error at the expense of a larger number of bits to encode the motion vectors and the type of partitions. Thus the choice of mode has a crucial role on rate-distortion optimization.

The method of *Lagrange multipliers* (LM) has been proposed for mode selection to trade off between the quality of the compressed video and the bit rate generated [1]–[3]. By exploiting

the rate-distortion relationship in LM determination, these approaches perform better than traditional approaches that minimize only distortion or bit rates, but their large computational complexity limits their applications. Moreover, the same LM value does not provide the best results for all video sequences.

Some fast mode selection algorithms have been proposed in [4]–[11]. Most of the algorithms reduced computational time by sacrificing some quality or increasing bit rate. For example, Ahmad *et al.* [4] proposed a fast mode selection algorithm using motion vector cost and previous frame information. The experimental results showed that this approach reduced encoding time by around 32% while increasing bitstream size by around 18% without any degradation of image quality. The main disadvantages of this approach are the extra memory requirement to store the previous information and the increase in bitstream size due to scene changes. Yang *et al.* [5] proposed another fast mode selection algorithm which stopped motion searches early for some cases, thus skipping a large number of search points. Early termination of the search process reduced the encoding time by 15 ~ 25% without increasing of bitstream size for the same image quality throughout a wide bit rate range when applied to some standard CIF video sequences. This process would suffer significant performance degradation if there is a mistake in the early termination due to the presence of a local minimum in the performance surface. Moreover, there is less encoding time saving for high motion video sequences using full search motion estimation. The approaches [8]–[11] proposed innovative ideas to filter out a number of modes from motion estimation and compensation for an MB using either smoothness of the block compared to the reference blocks or the property of an all-zero coefficients block.

There are three ways to reduce the encoding computational time. One is reducing the number of search points in the motion estimation (all faster motion search algorithms together with [4]–[7]), another is reducing the candidate modes per *macroblock* (MB) (all algorithms in [8]–[11]), and the other is reducing the number of operations in the mode selection by redefining the optimization cost function. The first two types of algorithms try to reduce computational time by reducing the search points in motion estimation or filter out a number of modes per MB. Undoubtedly, reducing the number of search points and/or the number of modes also degrades the image quality and increases the size of the bit stream. On the other hand, there is no efficient algorithm so far to reduce the computational time in mode selection (i.e., by changing the so called optimization cost function) compared to the LM approach used in [1]–[3]. Any reduction of the mode selection process can be also applied on top of the above mentioned other two categories for further reduction of computational complexity. It is quite ob-

Manuscript received November 07, 2007; revised November 19, 2008. First published April 17, 2009; current version published May 15, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yo-Sung Ho.

M. Paul is with the Gippsland School of Information Technology, Monash University, Churchill, Vic 3842, Australia (e-mail: Manoranjan.Paul@infotech.monash.edu.au).

M. R. Frater and J. F. Arnold are with School of Information Technology and Electrical Engineering, University College, The University of New South Wales, Australian Defence Force Academy, Canberra, ACT-2600, Australia (e-mail: m.frater@adfa.edu.au; j.arnold@adfa.edu.au).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMM.2009.2017610

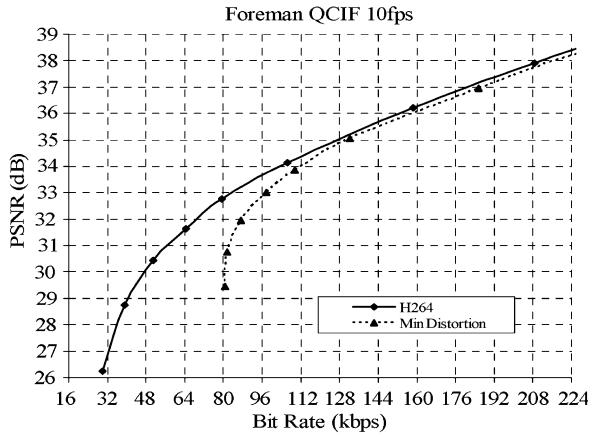


Fig. 1. Bit rates in kilobits per second (kbps) versus peak signal-to-noise ratio (PSNR) in dB curves using the LM and the only minimization of distortion for *Foreman* QCIF sequence where 10 frames per second is used.

vious that the performance of a video encoder depends on the quality of the reconstructed video at a range of bit rates, and the quality of the video depends upon the best choice of the encoder parameters. In addition to that, we can easily conclude that the best choice of encoder parameters could ensure not only the best video quality but also reduce the required bit rates. Thus, our motivation is that reconstructed video quality would be a dominant feature for selecting the encoding parameters especially for mode selection. If just the minimization of the distortion is considered for the decision of the coding tools, the achieved distortion is small but the required bit rate may be high. On the other hand, if just the bit rate is considered, the achieved bit rate is small but the distortion may be high [12]. In this paper, we propose a simplified LM approach, in which the mode selection criterion takes into account distortion together with amount of data required for motion vectors and variable block type. Experimental results confirm that this new technique reduces the computational complexity by at least 31% without degrading video quality or increasing bit rate and the overall gain would be more than 12% compared to the H.264.

This paper is organized as follows: Section II describes the mode selection approaches using LM and minimizing distortion. Section III describes the proposed method. Section IV analyzes the computational time of both approaches. Section V shows the experimental results and Section VI concludes the paper.

II. MODE SELECTION CRITERIA

Video sequences contain widely varying content and motion in different parts of each image, necessitating the selection between different coding options with varying rate-distortion efficiency. Mode selection is the most important coding option which controls the rate-distortion efficiency in the recent video coding standard H.264/AVC. During the encoding process, all coding modes of every MB are examined and the resulting bit rates and distortions are calculated. A decision can be made based on either one or both. Considering only one criterion for

mode selection reduces the computational time but does not provide optimal rate-distortion performance. If just the minimization of the distortion is considered for the choice of mode, the achieved distortion is small but the required bit rate may be high. On the other hand, if just the rate is considered the achieved rate is small but the distortion may be high [8]. Better rate-distortion performance is achieved by combining both.

In this method, the LM (λ) is first calculated with an empirical formula using the selected *quantization parameter* (QP) for every MB [3]

$$\lambda = 0.85 \times 2^{\frac{(QP-12)}{3}}. \quad (1)$$

During the motion estimation and encoding processes, all modes of every MB are examined and the resulting rates, $R(m_i)$, and the distortions, $D(m_i)$, are determined, where m_i is the i th ($i = 1 \dots 7$) mode. The Lagrangian cost function is defined for mode selection as

$$J^{LM}(m_i) = D(m_i) + \lambda \times R(m_i) \quad (2)$$

where $R(m_i)$ is the sum of the bits for mode m_i including the mode information, the motion vectors and the transformation coefficients, while $D(m_i)$ is measured as the *sum of square difference* (SSD) between the original MB and the corresponding reconstructed MB for mode m_i . The cost function (2) can also be expressed as

$$J^{LM} = D + \lambda \times (R_{MV} + R_H + R_{DCT}) \quad (3)$$

where R_{MV} , R_H , and R_{DCT} are the numbers of bits for motion vectors, block type, and DCT coefficients, respectively.

The mode m_n is selected as follows:

$$m_n = \arg \min_{\forall m_i} (J^{LM}(m_i)) | R(m_i) \leq R^T \quad (4)$$

where R^T is the target bit rate.

The LM is a function of QP and is used as a weighting factor between distortion and bit rate. By exploiting the rate-distortion relationship in LM determination, this approach performs better for many video sequences. But, due to its high computational requirement, some low/medium processing devices cannot afford it. Moreover, use of the same LM value does not provide the best results for all video sequences [1].

III. PROPOSED METHOD

We have noted previously that the minimization of distortion does not give satisfactory performance in the low bit rate range, but rivals the performance of the LM approach at higher bit rates (see Fig. 1). The same trend is also observed in other standard video sequences. The main reason behind this is the tendency to select relatively small blocks when minimizing only distortion, i.e., the number of modes per MB is high (as show in Fig. 2). Thus, our motivation is that we have to change the Lagrangian multiplier for minimization of distortion in such a way that an encoder takes large block type at low bit rates and small block type at high bit rates.

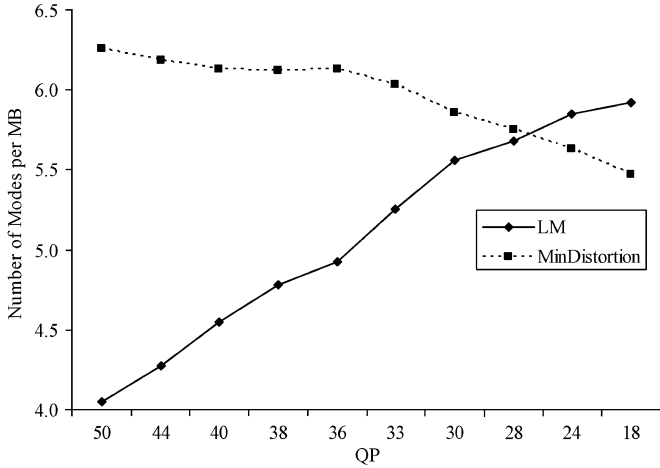


Fig. 2. Average number of modes per MB against QP is investigated for motion estimation and compensation using the LM and the only minimization of distortion.

We also note that a high proportion of the computational effort in the LM method is in calculating the numbers of bits required for the DCT coefficients. We therefore propose that mode selection be accomplished using the cost function

$$J^{Dist} = D + \lambda_m \times (R_{MV} + R_H) \quad (5)$$

which takes into account distortion and the bits use for the header comprising motion vectors and mode selection. R_{MV} and R_H are cheaper because the associated computational cost is lower.

The value of the Lagrangian multiplier, λ_m , could be chosen to be the same as the value of λ in the original LM method. In the following section we describe the detail procedure for generating the optimal value of λ_m using our proposed mode selection method.

To choose a Lagrangian multiplier with corresponding quantization parameter is a “chicken and egg” problem and requires huge number of iterations with enormous combinations of input parameters, e.g., frames rate, GOP length, picture type (e.g., I, P, B, etc. [15]), video coding standard profile, etc. To get the solution, we investigate the optimal value using the fundamental process used in [2]. At first, from the experimental result we try to get the nature of the values of LM with different QPs and then assume some values. Using those values we try to get the original value using different video sequence. It would be realistic if we can use all modes with all possible QPs (from 0 to 51) for a given λ . But this is very time consuming and we do not expect the new LM, λ_m , would be far different from the original one. Thus, we use a QP calculated from a given λ and consider $\pm v$ from the calculated QP to test the sensitivity to a variation of LM. The QP is first calculated from LM using a logarithmic function. To obtain a relationship between QP and λ_m , the minimization of the Lagrangian cost function in (5) is examined for the all modes using $QP \pm v$, which permits changing QP by a small amount to see the effect on the LM. More precisely, the macroblock mode decision is made by minimizing (5) over the set of macroblock modes. The modes are Intra modes,

skip mode, and other Inter modes with different QPs determined by $QP \pm v$. Note that, in our experiment, we use $v = 2$. For example, if the value of v and QP are 2 and 30, respectively, then we need to find corresponding total number of MBs for $QP = 28, 29, 30, 31, \text{ and } 32$. In this way, we calculated total number of MBs for QPs and then approximate the relationship of λ_m and QP for entire range of QPs by a function. The initial values of LM we assume in this experiment are $\{0.04, 0.06, 0.09, 0.15, 0.24, 0.38, 0.6, 1, 1.5, 2.5, 4, 6, 10, 15, 25, 40, 60, 100, 150, 250, 400, 600, 1000, 1500\}$.

Fig. 3 shows the relative frequency of chosen MBs with corresponding QPs for some values of λ_m . Each colour of curves represents the total number of MBs classified by $QP \pm 2$ where QP is calculated from a given LM, for example first curves (from left) in Fig. 3 represents an LM 0.04 which provides QPs = $\{1, 2, 3, 4, \text{ and } 5\}$ and their corresponding percentages of MBs classified. The y-axis of Fig. 3 shows the average percentage of MBs finally classified by different QPs for a given LM. The LM λ_m is varied over 24 values, producing 24 normalized histograms for the chosen quantization value QP. The λ_m values are chosen from left to right in Fig. 3. The QP value does not vary much for a given value of λ_m for different video sequences. Moreover, experimental results show that the gain when permitting the variation is rather small, indicating the justification of assuming a fixed value for various parameters. The results also indicate that the chosen values for individual video sequences are almost identical and they are only dependent on the value of QP. We can easily form another curve using average MB quantizer values QP with corresponding λ_m . Fig. 4 shows these curves. The bold line in Fig. 4 depicts the function

$$\lambda_m = 0.3 \times 2^{\frac{(QP-12)}{3}} \quad (6)$$

which is an approximation of relationship between λ_m and $QP \in \{0 \dots 51\}$ in H.264 using minimization of distortion as criterion of mode selection.

The choice of λ_m , in our algorithm provides very similar performance to the more complex approach. For simplicity we did not show all results generated using other video sequences in Fig. 4. When we plot the proposed LM, we use the average results of a number of standard video sequences. The normal trend is that video sequences having high motion require a high λ_m and video sequences having slow and smooth motion require a small λ_m . The experimental results show that for standard video sequences the value of λ_m varies from $\lambda_m = 0.25 \times 2^{(QP-12)/3}$ to $\lambda_m = 0.35 \times 2^{(QP-12)/3}$, thus our recommended value is $\lambda_m = 0.3 \times 2^{(QP-12)/3}$.

For the sake of completeness, block diagrams of mode selection using LM and minimization of distortion with header are given in Figs. 5 and 6, respectively. Thus, in the proposed algorithm, operations for bit rate calculations are totally omitted in the mode selection process but not in the final encoding using the selected mode. As a result, a significant number of operations are saved.

IV. COMPUTATIONAL COMPLEXITY OF VARIOUS APPROACHES

The computational complexity of above two approaches for mode selection and encoding is described in this section. Note

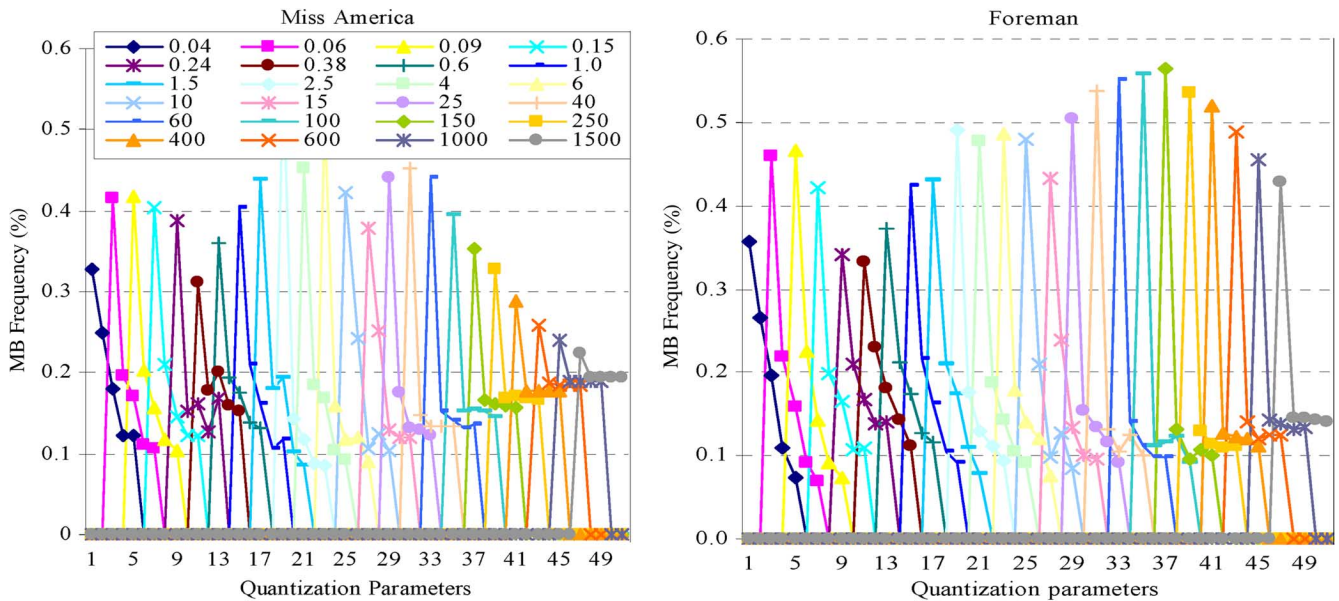


Fig. 3. Relative MB frequency versus QP for various values of the Lagrangian multiplier λ_m . The relative frequencies of MBs are gathered while coding 100 frames of QCIF video sequences *Miss America* and *Foreman* with frame rate 10 Hz.

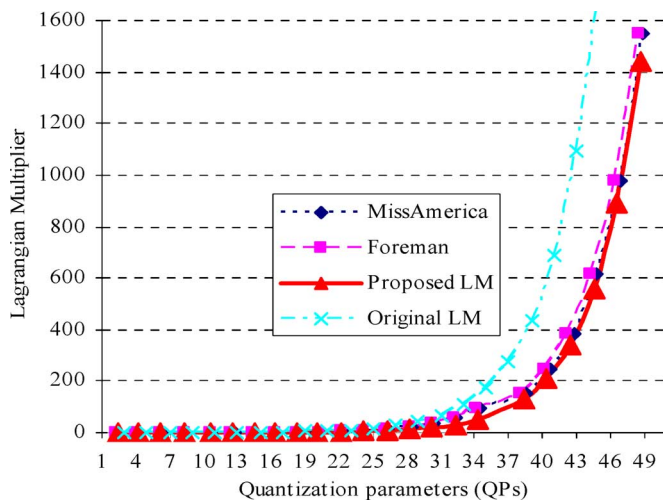


Fig. 4. Lagrangian multiplier with corresponding average MB QP frequency for a number of standard video sequences where as we only show two video sequences.

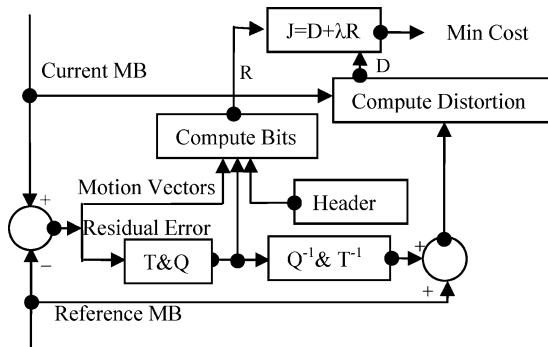


Fig. 5. H.264 mode selection using LM.

that for simplicity we use the “add-equivalent” operation which is equivalent to any addition, subtraction, comparison, or shift

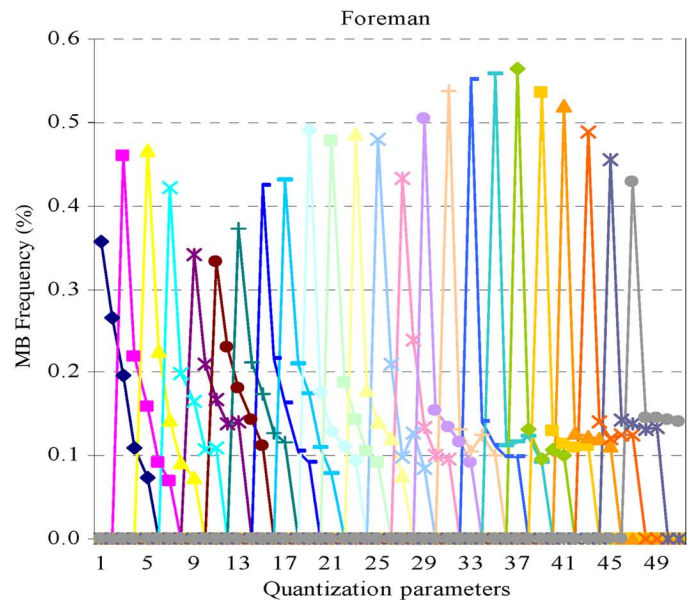


Fig. 6. H.264 mode selection using minimization of distortion with motion vector and header information.

operations and half of a multiplication/division operation. The H.264 video coding standard uses total seven Inter variable block size (such as 16×16 , 16×8 , 8×16 , 8×8 , 8×4 , 4×8 , and 4×4 pixels) modes for motion estimation and compensation. Normally first four modes are motion estimated and compensated, and a mode is selected using Lagrangian optimization. If 8×8 mode is selected among them, then the rest of the modes are also estimated and compensated, and one mode is selected using optimization. To compare the computational results average number of modes (which could be varied from 4 to 7) per MB is crucial. Let c be the average number of modes investigated for every MB. For each mode, the following calculations are required for each 16×16 macroblock:

- 1) residual error calculation: 256 (subtraction) add-equivalent operations;
- 2) DCT transformation: 1024 addition and 256 shift operations [13], i.e., 1280 add-equivalent operations;
- 3) quantization: 256 multiplications (for transformation scaling [13]) and 256 multiplications (for quantization adjustment), 256 addition (for “dead zone”) and 256 shift operations (assuming multiple bit shift is permitted) [14], i.e., 1536 add-equivalent operations;

TABLE I
COMPUTATIONAL COMPLEXITY COMPARISONS USING LM AND MINIMIZATION OF DISTORTION INCLUDING HEADER AND MOTION VECTOR INFORMATION

Sequences	Modes per MB using LM algorithm	Modes per MB using proposed algorithm	Reduced operations using proposed algorithm against LM algorithm in mode selection	Reduced operations using proposed algorithm against LM algorithm in overall encoding
Foreman	4.55	4.55	31%	12%
Football	5.30	4.79	38%	15%
Tennis	4.98	4.56	37%	15%
Coastguard	4.85	4.65	34%	14%
Miss America	4.29	4.27	32%	13%
Claire	4.30	4.22	32%	13%

- 4) bit rate calculation: 1) 256 comparisons and 255 addition operations for finding total number of zeros/nonzeros coefficients; 2) 256 comparisons and 48 additions operations for finding trailing ones; 3) 992 comparisons operations for finding VLC index for *Coeff-Token*; 4) 256 comparisons and 256 addition operations for coding sign bit of nonzero coefficients; 5) 64 comparisons and 64 addition operations for finding the appropriate *Coeff-Token* table and corresponding VLC code; 6) 256 comparison operations for finding the last nonzero coefficient; 7) 240 comparisons and 224 addition operations for finding total zeros before the last nonzero coefficient; 8) 224 comparisons and 16 additions operations for finding the VLC code for a given number of zeros before last nonzeros and total coefficients; 9) 512 comparisons and 256 addition operations for calculating *Suffix*, *Prefix*, and *ZeroBefore*; 10) 224 comparisons operations for *RunBefore*; and 11) 16 comparisons operations for adjusting the number of coefficients for the neighboring block [15], i.e., 4416 add-equivalent operations. Note that, we consider the worse case calculation here;
- 5) inverse quantization: same as step 3 [14];
- 6) inverse DCT transformation: same as step 2 [13];
- 7) distortion calculation: 256 multiplication and 256 subtractions for pixel comparison and 255 additions for error calculation, i.e., 1023 add-equivalent operations.

The rate-distortion control mechanism of H.264 using LM for mode selection and encoding requires all the above mentioned steps. On the other hand, the rate-distortion control mechanism of H.264 using minimization of only image distortion for each mode requires all steps except the fourth, while for encoding using the selected mode, it requires all steps. For final encoding we need to calculate only step four as all the other data is available from the mode selection process. Since an MB is encoded using either only first four modes or all the seven modes, we use experimental data (see Table I where we use $QP = 30$) for number modes per MB. The detail computational time comparison is shown in Table I where we find the value of c experimentally. The average number of modes depends on the video sequences. Normally a video sequence with high object motion has large value (e.g., Football) and a video with low object motion has small value (e.g., Claire). It is due to the fact that a video sequence with high motion uses all the modes from large to small blocks (i.e., total seven modes per MB more frequently); on the other hand, a video with low motion uses only large blocks (i.e., four modes per MB more frequently). For each MB the LM and proposed algorithms require 56 635 and 38 971 add-equivalent operations, respectively. From the table, we can

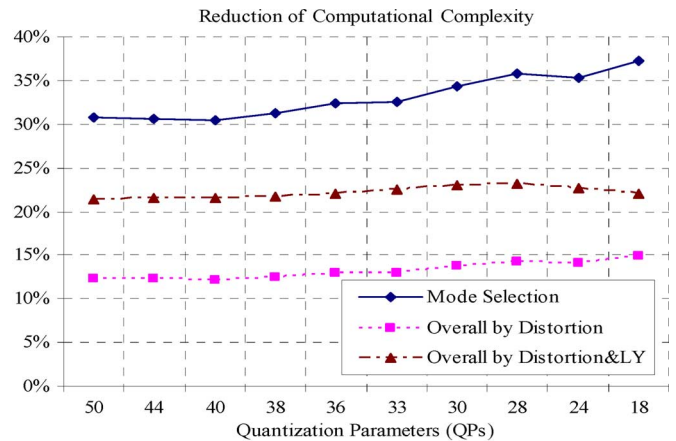


Fig. 7. Average reduction of computational complexity in mode selection and overall encoding time against the quantization parameters for six standard video sequences using the proposed distortion and the distortion with [5] (Distortion&LY) methods.

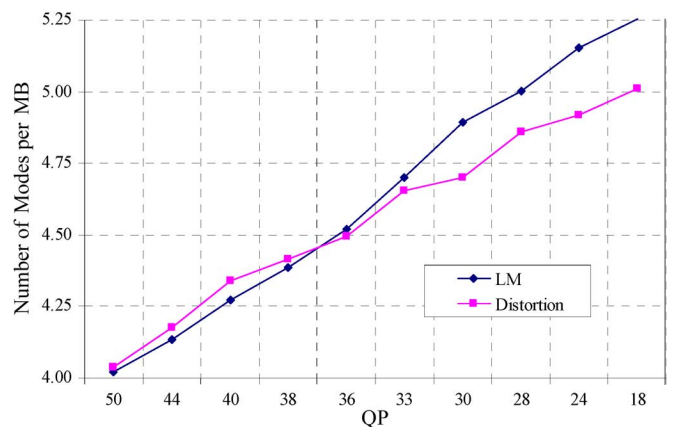


Fig. 8. Average number of modes processed by each MB using two different approaches against quantization parameters of H.264.

safely conclude that minimizing the distortions as a mode selection criterion reduces the computational time for mode selection by at least 31% compared to the LM optimization technique. The researchers already claimed that motion estimation, irrespective of a scene's complexity, typically comprises more than 60% of the processing overhead required to encode an inter picture with a software codec using the DCT [20], when full search is used. Thus, while full search is applied the proposed algorithm reduces $31 \times .40 = 12.4\%$ overall encoding time. Moreover, we did not consider memory access time for the VLC codes (such as *Coeff-Token*, *Total_Zeros*, and *Run_Before*) of

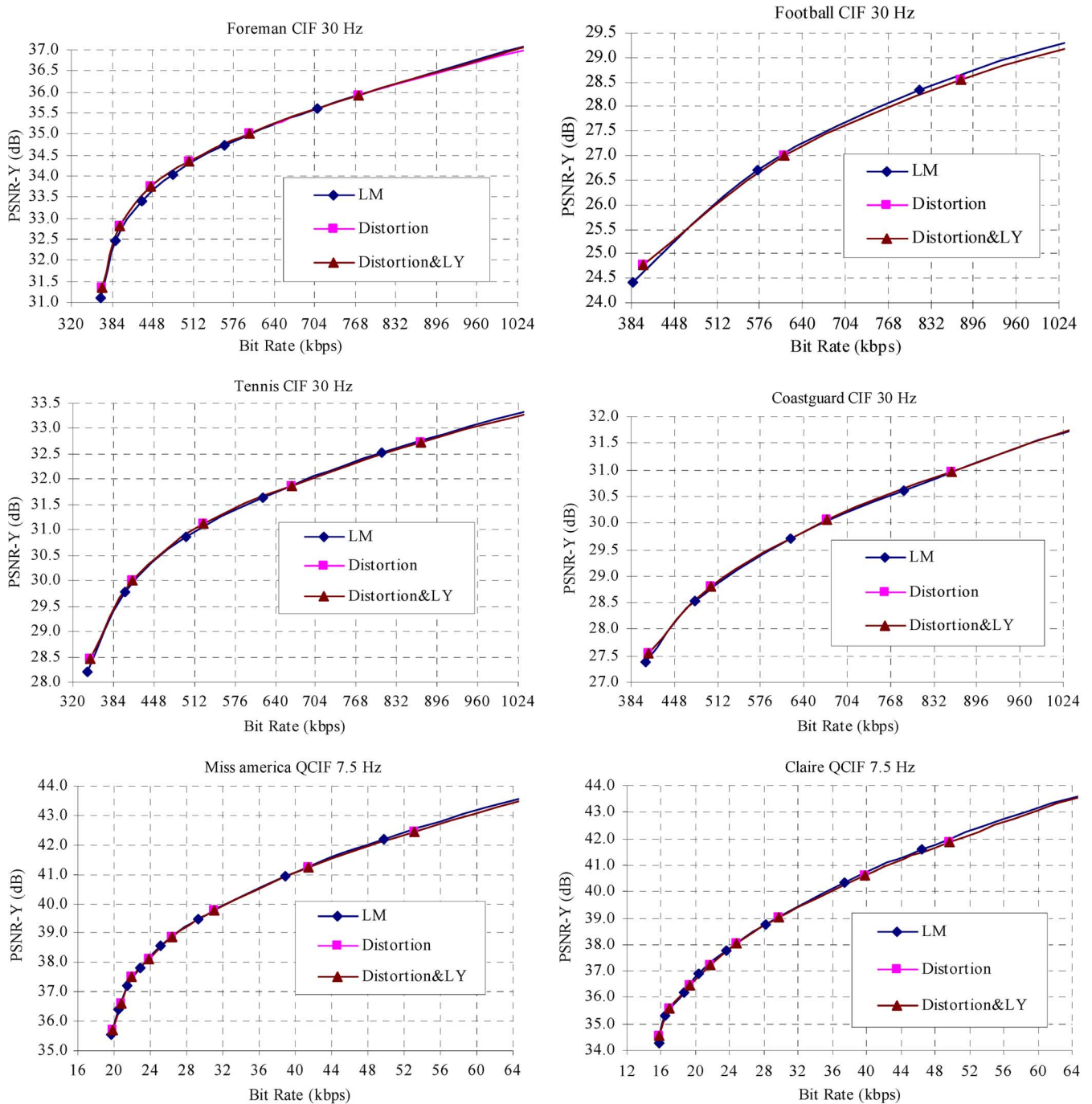


Fig. 9. Rate-distortion curves for standard video sequences.

bit rate calculation in LM process. Thus in real implementations our approach is even more efficient when compared to the LM approach. We note that this performance remains almost the same when the fast mode selection algorithm described previously in [5] is also applied in the LM and in our proposed algorithms.

Fig. 7 shows the average reduction of computational complexity in only mode selection and overall encoding time with QPs using six standard video sequences. The mode selection complexity is reduced by more than 30% using the proposed method but the overall encoding time reduction is more than 12% and 22% using the proposed method and the proposed method with Yang's [5] method. The figure indicates that the

computational reduction rate increases with the bit rates because the increasing rate of average number of modes per MB decreases with the bit rates compared to the LM technique (see Fig. 8). We also observed that the reduction rate in computational complexity also increases with the image size due to the more spatial correlation in the large image.

V. EXPERIMENTAL RESULTS

We have implemented our proposed algorithms based on the Baseline profile of H.264/AVC with full search motion estimation of maximum ± 7.5 pixel search width [15] for a number



Fig. 10. (a) *Foreman* frame number two, (b) and (c) reconstructed frames using the LM and Distortion algorithms, respectively, and (d) and (e) frame differences ($\times 10$) of (b) and (c), respectively, with respect to (a).

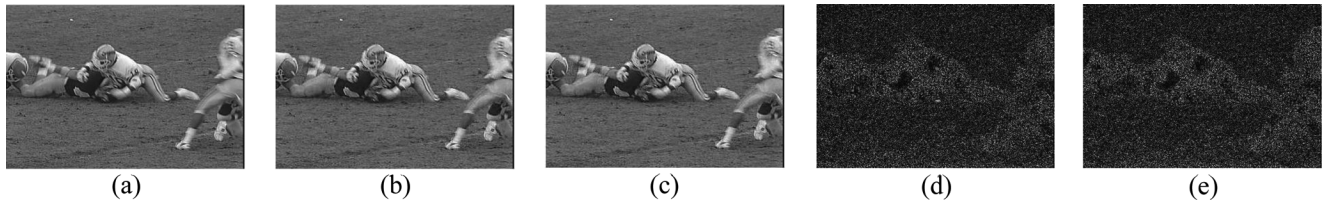


Fig. 11. (a) *Football* frame number two, (b) and (c) reconstructed frames using the LM and Distortion algorithms, respectively, and (d) and (e) frame differences ($\times 10$) of (b) and (c), respectively, with respect to (a).

TABLE II
PERFORMANCE COMPARISON OF PROPOSED TECHNIQUE
(DISTORTION) COMPARED TO THE STANDARD LM TECHNIQUE
USING BD-BIT RATE AND BD-PSNR

Video	BD-Bit Rate (%)	BD-PSNR
Foreman	0.48	0.06
Football	-7.01	-0.09
Tennis	-4.53	-0.08
Coastguard	-3.75	-0.04
Miss America	2.90	-0.03
Claire	-6.11	0.04

of standard [16] video sequences with CIF and QCIF format, namely, *Foreman*, *Football*, *Tennis*, *Coastguard*, *Mobile & Calendar*, *Miss America*, *Claire*, *Car phone*, *News*, *Salesman*, *Suzie*, and *Grandma*. For brevity we show our rate-distortion performance using the first 100 frames of six standard video sequences. In this experiment, the GOP size is 12 and we use only *I* and *P* frames. From now on, H.264/AVC with minimizing the Lagrangian cost function for mode selection is called the LM algorithm, and the corresponding H.264/AVC with minimizing the Distortion cost function including motion vectors and header bits with and without early termination algorithm [5] are termed as the Distortion and Distortion&LY algorithms respectively.

The experimental results show that the number of modes processed for each MB using the LM and the Distortions algorithms is almost the same. The distortion algorithm processed a smaller number of modes compared to the LM algorithm (Fig. 8) at very high bit rates. This has a small impact on coding performance at very high rates.

Table II shows performance comparison of the Distortion algorithm compared to the LM algorithm in terms of BD-Bit Rate and BD-PSNR [18], [19]. From the table we observe that on average only 0.02 dB BD-PSNR or 3% BD-Bit rate we sacrifice using our proposed technique compared to the standard LM technique. Fig. 8 shows that the proposed algorithm took more number of modes per MB (at low bit rate), i.e., relatively more small blocks are selected. Obviously small blocks provide less image distortion. If those small blocks require comparable bits at low bit rate (i.e., compared to the large blocks) due to

the coarse quantization, smooth motions, and large background, then the overall rate-distortion performance will be better. Due to the smoothness of the motion and large background in *Miss America* sequences (where around 30% block are skip blocks), the proposed algorithm provides better quality at low bit rate regions. But this scenario is not observed at mid \sim high rates. We take the integration summation up to mid-range bit rates/PSNR for BD-Bit rate and BD-PSNR calculations. That is why 2.9% BD-Bit rate gain is observed. If we take the data up to high range, it would be very similar. Note that the negative values indicate inferiority of our technique.

The final rate-distortion performance using three algorithms namely LM, Distortion and Distortion&LY is demonstrated in Fig. 9 for four CIF and two QCIF standard video sequences with 30 *frames per second* (fps) or 30 Hz and 7.5 fps (or 7.5 Hz), respectively. The experimental results reveal that at a wide range of bit rates using the Distortion method exhibits very similar performance to the LM method. To be more specific, the proposed method reduces mode selection and overall encoding time by around 31% and 12% (see Table I), respectively, with virtually no change in performance over a wide range of bit rates. Note that the proposed method has the same effectiveness when we incorporate the contemporary efficient algorithm [5] in the LM and our proposed algorithms.

The human visual system does not respond to stimuli in a straightforward manner. It is therefore, widely accepted that objective assessment based on PSNR does not always provide reliable assessments of video quality, since a higher PSNR may not always guarantee better video quality [16]. It has become common practice in international coding-standard activities to combine both objective and subjective assessments in evaluating and comparing video coding algorithms.

To compare the perceptual performance, we performed a subjective test using *Double-Stimulus Continuous Quality Scale* (DSCQS) assessment using the test conditions of [17]. In this method, we used a number of video sequences serially. In each test, the viewers are asked to rate the quality of two video sequences known as “A”, and “B” on a continuous scale ranging between “Excellent” and “Bad”. Either A or B (chosen at arbitrarily) was a reconstructed video sequence using LM algorithm while the other was a reconstructed sequence using the pro-

posed algorithm. The experimental results showed that viewers marked them as very similar.

To compare the perceptual performance of both algorithms, we reconstruct the images using both techniques. The original frame (frame number two), reconstructed frame, and frame differences are presented in Figs. 10 and 11 for the *Foreman* and *Football* sequences, respectively. The bits per frame and PSNR of these two sequences using both algorithms are 20831, 36.28 and 52184, 32.76 for *Foreman* and *Football*, respectively. The intensity of each frame difference image has been magnified by a factor of ten in order to provide an improved visual comparison. In both examples, reconstructed frames using the LM and Distortion algorithms can be readily perceived as similar.

These results demonstrate that the inclusion of the bits required for coding of DCT coefficients in the conventional cost function (3) is unnecessary, adding significantly to the computational cost without improving performance.

VI. CONCLUSIONS

In this paper, a simplified mode selection technique is proposed based on a new Lagrangian cost function using distortion, the numbers of bits required for motion vectors and block type. The experimental results demonstrate that this technique significantly reduces computational time in mode selection and entire encoding time by 31% and 12% compared to the original Lagrangian optimization technique while maintaining essentially the same rate-distortion performance over a wide range of bit rates. The proposed approach can be applied together with fast motion search algorithms and the mode filtering algorithms for further speed up.

REFERENCES

- [1] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.
- [2] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Proc. IEEE Int. Conf. Image Processing (ICIP-01)*, 2001, vol. 1, pp. 542–545.
- [3] T. Weigrand, H. Schwarz, A. Joch, and F. Kossentini, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–702, Jul. 2003.
- [4] Ahmad, A. N. Khan, S. Masud, and M. A. Maud, "Selection of variable block sizes in H.264," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2004, vol. 3, pp. 173–176.
- [5] L. Yang, K. Yu, J. Li, and S. Li, "An effective variable block-size early termination algorithm for H.264 video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 6, pp. 784–788, Jun. 2005.
- [6] D. Kim and J. Jeong, "A fast mode selection algorithm in H.264 video coding," in *Proc. IEEE Int. Conf. Multimedia and Expo (ICME-04)*, 2006, pp. 1709–1712.
- [7] H. Kim and Y. Altunhasak, "Low complexity macroblock mode selection for H.264/AVC encoders," in *Proc. IEEE Int. Conf. Image Processing*, 2004, vol. 2, pp. 765–768.
- [8] B. Jeon and J. Lee, "Fast Mode Decision for H.264," Document JVT-J033.doc, Dec. 2003.
- [9] H. Wang, S. Kwong, and C.-W. Kok, "An efficient mode decision algorithm for H.264/AVC encoding optimization," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 882–888, Jun. 2007.
- [10] Y.-H. Kim, J.-W. Yoo, S.-W. Lee, J. Shin, J. Paik, and H.-K. Jung, "Adaptive mode decision for H.264 encoder," *Electron. Lett.*, vol. 40, no. 19, pp. 1172–1173, Sep. 2004.
- [11] X. Jing and L.-P. Chau, "Fast approach for H.264 inter mode decision," *Electron. Lett.*, vol. 40, no. 17, pp. 1050–1052, Sep. 2004.
- [12] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: Tools, performance, and complexity," *IEEE Circuits Syst. Mag.*, vol. 4, no. 1, pp. 7–28, First Quarter 2004.

- [13] C.-P. Fan, "Fast 2-dimensional 4×4 forward integer transform implementation for H.264/AVC," *IEEE Trans. Circuits Syst. II*, accepted for publication.
- [14] H. S. Malvar, A. Hallapuro, M. Karczewicz, and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 598–603, Jul. 2003.
- [15] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*. New York: Wiley, 2003.
- [16] Y. Q. Shi and H. Sun, *Image and Video Compression for Multimedia Engineering Fundamentals, Algorithms, and Standards*. Boca Raton, FL: CRC, 1999.
- [17] Recommendation ITU-R BT.500-10, Methodology for the Subjective Assessment of the Quality of Television Pictures.
- [18] G. Bjontegaard, Calculation of Average PSNR Differences Between RD-Curves, VCEG-M33, ITU-T Q.6/SG16 VCEG, 2001.
- [19] S. Kondo and H. Sasai, "Motion-compensated video coding using sliced blocks," *Syst. Comput. Jpn.*, vol. 38, no. 7, pp. 12–22, 2007.
- [20] T. Shanableh and M. Ghanbari, "Heterogeneous video transcoding to lower spatio-temporal resolutions and different encoding formats," *IEEE Trans. Multimedia*, vol. 2, no. 2, pp. 101–110, Jun. 2000.



Manoranjan Paul (M'03) received the B.Sc.Eng. (hons.) degree in computer science and engineering from Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, in 1997 and the Ph.D. degree from the Monash University, Churchill, Australia, in 2005.

He joined the Computer Science and Engineering Department, Ahsanullah University of Science and Technology, Dhaka, Bangladesh, as a Lecturer in 1997 and was promoted to Assistant Professor in 2000. He worked as a Research Fellow in the

University of New South Wales, ADFA, Canberra, Australia, in 2005–2006. He is currently working as a research Fellow at Gippsland School of IT, Monash University. His major research interests are in the fields of image/video coding, multimedia communication, computer vision, and vertical handoff. He has published more than 30 refereed international journals, book chapters, and conference publications.

Dr. Paul is a full member of the ACS. He has served as a guest editor for the *Journal of Multimedia*.



Michael R. Frater (S'89-M'91) received the B.Sc. and B.E. degrees from the University of Sydney, Sydney, Australia, in 1986 and 1988, respectively, the Ph.D. degree from the Australian National University, Canberra, Australia, in 1991, and the M.H.Ed. degree from the University of New South Wales, Sydney, Australia, in 1996.

Since 1991, he has been with the University of New South Wales at the Australian Defence Force Academy, Canberra, where he is currently an Associate Professor. His research interests lie in the fields of audiovisual and multimedia communications and communication systems.

Prof. Frater has served as an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING.



John F. Arnold (S'77-M'85-SM'96) received the B.E. and M.Eng.Sc. degrees from the University of Melbourne, Melbourne, Australia, in 1976 and 1979, respectively, and the Ph.D. degree from the University of New South Wales, Canberra, Australia, in 1984.

Since 1978, he has been with the School of Electrical Engineering, University of New South Wales, initially at the Royal Military College, Duntroon, and more recently at the Australian Defence Force Academy, Canberra. He is currently a Professor of electrical engineering and Head of the School of Information Technology and Electrical Engineering. His research interests lie in the fields of video coding, error resilience of compressed digital video, and coding of remotely sensed data. He has published widely in these areas.

Prof. Arnold is an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.