

## **EFFICIENT MODE SELECTION ALGORITHM USING IMAGE DISTORTION FOR H.264 VIDEO ENCODER**

*Manoranjan Paul, Michael Frater, and John Arnold*  
UNSW@ADFA, Northcott Drive, Canberra ACT 2600, Australia  
E-mail: manoranjan@ieee.org

### **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 results to the full constrained optimization, while it saves 31% mode selection computational time.

### **1. INTRODUCTION**

The variable block size motion estimation and compensation in H.264/AVC is one of the most remarkable innovations when compared to previous video coding standards. The block sizes (popularly known as modes) are chosen in the range  $16 \times 16$  to  $4 \times 4$ . Choosing larger partition sizes requires a relatively small 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 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, in constrained of bit rates, 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, but their large computational complexity limits their application. 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][5]. 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 12~20% without increasing of bitstream size for the same image quality throughout the whole bit rate ranges when applied for some standard CIF video sequences (e.g., *Foreman*, *Football*, *Tennis*, and *Coastguard*). This process would suffer large performance degradation if there is a mistake in early termination due to the presence of a local minimum in the performance surface. Moreover, there is no significant encoding time saving for high motion video sequences with full search motion estimation.

There are two ways to reduce the computational time in encoding. One is reducing the search points in motion estimation and another is reducing operations in the mode selection. The above mentioned algorithms together with fast motion estimation algorithms (e.g., three step search [9]) tried to reduce computational time by reducing the search points in motion estimation. Undoubtedly, reducing the searching points also degrade the image quality and increase the bit stream. On the other hand, there is no efficient algorithm so far to reduce the computational time in mode selection compared to the LM approach used in [1]–[3]. 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 rate is considered the achieved bit rate is small but the distortion may be high [6]. In this paper, for the first time, we propose a simplified LM approach, in which the mode selection criterion takes into account the distortion together with bit information of motion vectors and variable block type. Experimental results confirm that this new technique reduces the computational complexity by at least 31% without significantly degrading image quality or increasing bit rate.

This paper is organized as follows: Section 2 describes mode selection approach using LM. Section 3 describes the proposed method. Section 4 analyses the computational time for both approaches. Section 5 shows the experimental results and Section 6 concludes the paper.

### **2. 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 dominant 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 *MacroBlock* (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 [6]. Better rate-distortion performance is achieved by combining both.

In this method, the LM ( $\lambda$ ) 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 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 original MB and 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:

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

where  $R^T$  is the target bit rate.

The LM is a function of QP and is used as a weighting factor between distortions and bit rates. 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].

### 3. 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 high bit rates. The main reason behind this is the tendency to select relatively small blocks when minimizing only 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 motion vectors and mode selection.

The value of the Lagrange multiplier,  $\lambda_m$ , could be chosen to be the same as the value of  $\lambda$  in the original LM method. In this paper, we choose

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

We do not claim that this choice of  $\lambda_m$  is optimal but simply observe that, with this arbitrary choice of  $\lambda_m$ , our algorithm provides very similar performance to the more complex approach. Since we ignore a number of bits of DCT coefficients in cost function calculation, obviously our new LM constant would be small compared to the existing one. It is left for future work to determine an optimal value for  $\lambda_m$ .

For the sake of completeness, a block diagram of mode selection using those techniques is given in Figure 1 where all the steps are needed for LM algorithm while only solid lines steps are needed for minimization of distortion. Thus, in the proposed algorithm, operations for bit rate calculation using CAVLC [9] are only required for final encoding using selected mode but not in mode selection. As a result, a significant number of computing and memory access (for VLC code in look up table) operations are saved.

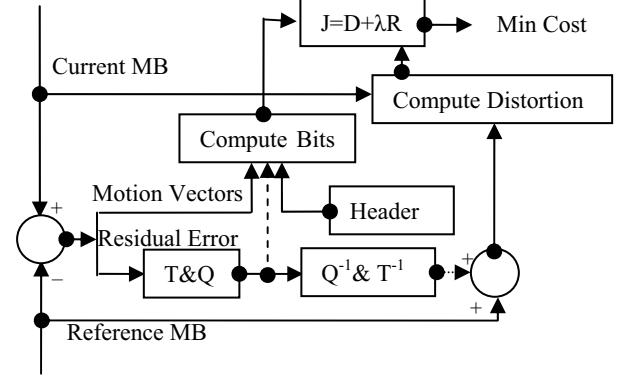


Figure 1: H.264 mode selection using LM (all parts) and minimization of distortion (all except the dotted parts).

### 4. COMPUTATIONAL COMPLEXITY OF VARIOUS APPROACHES

The computational complexity of above two approaches for mode selection and encoding is described in this section. Note that for simplicity, we defined add-equivalent operation which is equivalent to any addition, subtraction, comparison, or shift operations and half of the multiplication/division operation. Let  $c$  be the total number of modes investigated for every MB. For each mode, the following calculations are required for each 16x16 macroblock: 1) Residual error calculation: 256 (subtractions) add-equivalent operations. 2) DCT transformation: 1024 addition and 256 shift operations [7], i.e., 1280 add-equivalent operations. 3) Quantization: 256 multiplications (for transformation scaling [7]) and 256 multiplications (for quantization adjustment), 256 addition (for “dead zone”) and 256 shift operations (assuming multiple bit shift is permitted) [8], i.e., 1536 add-equivalent operations. 4) Bit rate calculation: (i) 256 comparison and 255 addition operations for finding total number of zeros/non-zeros coefficients, ii) 256 comparison and 48 addition operations for finding trailing ones, iii) 992

comparison operations for finding VLC index for *Coeff\_Token*, iv) 256 comparison and 256 addition operations for coding sign bit of non-zero coefficients, v) 64 comparison and 64 addition operations for finding the appropriate *Coeff\_Token* table and corresponding VLC code, vi) 256 comparison operations for finding the last most non zero coefficient, vii) 240 comparison and 224 addition operation for finding total zeros before the last most non-zero co-efficient, viii) 224 comparison and 16 addition operations for the finding the VLC code for given number of zeros before last most non-zeros and total coefficients, ix) 512 comparison and 256 addition operations for calculating *Suffix*, *Prefix*, and *ZeroBefore*, x) 224 comparison operations for *RunBefore*, and xi) 16 comparison operations for adjusting number of coefficients for neighboring block, [9], i.e., 4416 add-equivalent operations. Note that, we consider the worse case calculation here. 5) Inverse quantization: same as step 3, [8]. 6) Inverse DCT transformation: same as step 2 [7]. 7) Distortion calculation: 256 multiplication and 256 subtraction for pixel comparison and 255 additions for error calculation operations, i.e., 1023 add-equivalent operations.

The rate-distortion control mechanism of H.264 using Lagrangian multiplier for mode selection and encoding requires at the above mentioned all 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 final mode, it requires all steps. For final encoding we need to calculate only step four as all the data are available in mode selection process. Since a MB is encoded using either only first four modes or all the seven modes, we use experimental data (see Table 1) for number of modes per MB. The detail computational time comparison is shown in Table 1 where we find the value of  $c$  experimentally. From the table, we can 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. Moreover, we didn't consider memory access time for the VLC codes of bit rate calculation in LM process. Thus in real situation our approach is more efficient compared to LM approach. We note that this performance remains almost same when the fast mode selection algorithm described previously in [5] is also applied in our proposed algorithm.

## 5. EXPERIMENTAL RESULTS

We have implemented our proposed algorithms based on the Baseline profile of H.264/AVC with full search motion estimation of maximum  $\pm 7.5$  pixel search width [9] for a number of standard [10] video sequences with CIF and QCIF format namely, *Foreman*, *Football*, *Tennis*, *Coastguard*, *Mobile & Calendar*, *Miss America*, *Claire*, *Car phone*, *News*, *Salesman*, *Suzie*, and *Grandma*. But 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 twelve and we use only *I* and *P* frames. From now on, H.264/AVC with minimising the Lagrangian cost function for mode selection is called the LM algorithm, the corresponding H.264/AVC with minimising the Distortion cost function including motion vectors and header bits is termed as the Distortion.

Table 1: Computational Complexity comparisons using LM and Distortion algorithm.

Sequences	Modes per MB using LM Algorithm	Modes per MB using proposed Algorithm	Reduced Operations against LM
Foreman	4.55	4.55	31%
Football	5.30	4.79	38%
Tennis	4.98	4.56	37%
Coastguard	4.85	4.65	34%

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 (see Figure 2) at very high bit rates. This has a small impact on coding performance at very high rates as we shall see later.

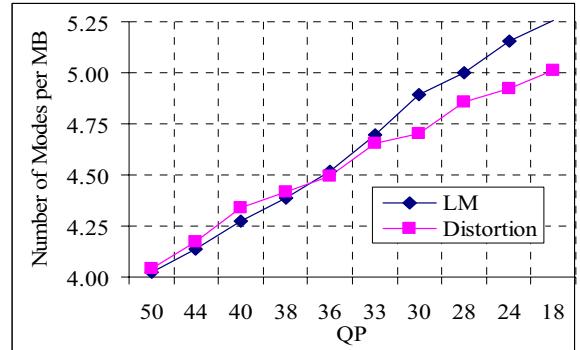


Figure 2: Average number of modes processed by each MB using two different approaches against different QP values.

The final rate-distortion performance is demonstrated in Figure 3 for four CIF and two QCIF standard video sequences with 30 frames per second (fps) and 7.5 fps 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 time by around 31% with virtually no change in performance over the range from low to high bit rates. Even at very high bit rates (2 mbps), the coding performance of the LM algorithm is only marginally better the proposed algorithm (not more than 0.2 dB). Note that the proposed method has the same effectiveness when we incorporate the contemporary efficient algorithm [5] in our proposed algorithm.

## 6. CONCLUSION

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 reduces significant computational time in mode selection by 31% compared to the original Lagrangian optimization technique while maintaining essentially the same rate-distortion performance over a wide range of bit rates. The improvement of this performance remains almost same when we incorporate early termination algorithm in our proposed algorithm.

## 7. REFERENCES

- [1] Sullivan, G. and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Magazine*, 74–90, 1998.
- [2] Wiegand, T. and B. Girod, "Lagrange multiplier selection in hybrid video coder control," *IEEE ICIP*, 1, 542–545, 2001.
- [3] Weigand, T., H. Schwarz, A. Joch, and F. Kossentini, "Rate-constrained coder control and comparison of video coding standards," *IEEE CSVT*, 13 (7), 688–702, 2003.
- [4] Ahmad, A. N. Khan, S. Masud, and M. A. Maud, "Selection of variable block sizes in H.264," *ICASSP*, 3, 173–176, 2004.
- [5] Yang, L., K. Yu, J. Li, and S. Li, "An effective variable block-size early termination algorithm for H.264 video coding," *IEEE Trans. on CSVT*, 15(6), 784–788, 2005.
- [6] Ostermann, J., 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 CS Magazine*, 7–28, 2004.
- [7] Fan, C. –P, "Fast 2-dimensional 4×4 forward Integer Transform implementation for H.264/AVC," *IEEE Transaction on CS II: Accepted*, 2005.
- [8] Malvar, H. S., A. Hallapuro, M. Karczewics, and L. Kerofsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Trans. on CSVT*, 13(7), 598–603, 2003.
- [9] Richardson, I. E. G., "H.264 and MPEG-4 video compression," *Wiley Press*, 2003.
- [10] Shi, Y.Q. and H. Sun, *Image and Video Compression for Multimedia Engineering Fundamentals, Algorithms, and Standards*, CRC Press, 1999.

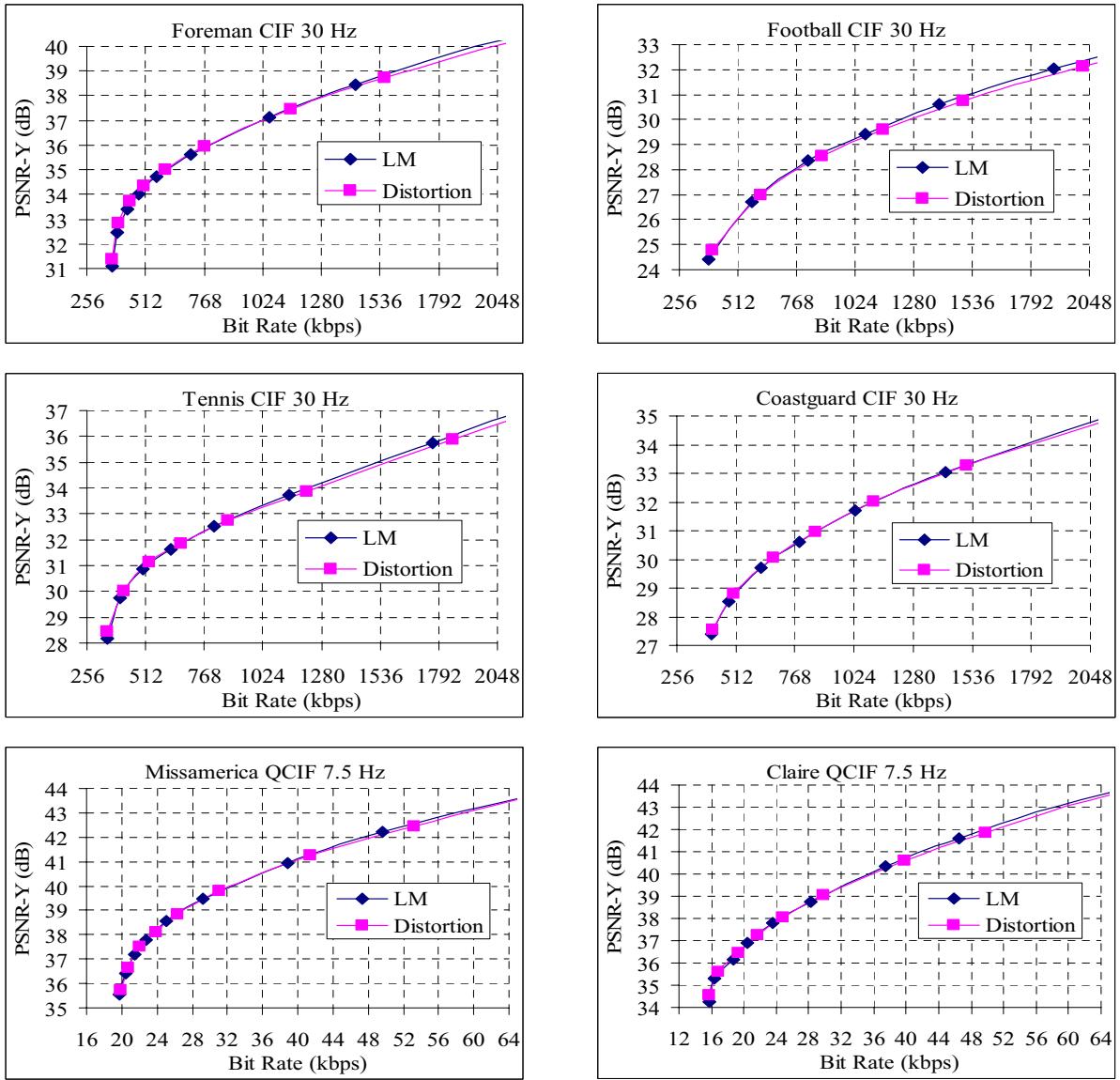


Figure 3: Rate-Distortion curves for four CIF and two QCIF standard video sequences.