A FAST MOTION-COST BASED ALGORITHM FOR H.264/AVC INTER MODE DECISION

E. Martínez-Enríquez, M. de-Frutos-López, J. C. Pujol-Alcolado, F. Díaz-de-María

Department of Signal Theory and Communications
Universidad Carlos III, Leganés (Madrid), Spain

ABSTRACT

The H.264/AVC standard achieves a high coding efficiency compared to previous standards. However, the encoder complexity results in very high computational cost due to motion estimation and macroblock mode decisions. In this paper we propose a fast mode decision for low computational complexity applications for which the rate distortion optimization mode decision becomes unacceptable. The proposed pruned mode decision method consists in a motion-cost based early termination algorithm and saves about 50% encoding time with negligible quality loss.

***Index Terms*—** H264, mode decision, low-complexity.

1. INTRODUCTION

The latest ISO/IEC H.264/AVC video coding standard has become an issue of active research in the last years. The new generation of video applications demands both quality and low bit rates. H.264/AVC achieves about a 50% rate reduction compared to previous standards such as MPEG2 and it is a versatile solution for a wide range of applications. Its key improvements revolve mainly around certain areas. In motion compensation, quarter pixel motion vectors are used, as well as up to sixteen reference frames and several block sizes. Entropy coding has been improved with both CAVLC and CABAC modes. A rate-distortion optimization (RDO) method has been developed for both Intra and Inter mode decision (MD).

This paper focuses on MD for P and B slices. H.264 offers a wide set of block sizes for motion compensation. A macroblock (MB) can be partitioned in blocks of 16x16, 16x8, 8x16 and 8x8 pixels for Inter coding. Each 8x8 block, called submacroblock (subMB), can be further divided into 8x4, 4x8 and 4x4 pixel blocks. Direct and Direct8x8 modes are a particular case of 16x16 and 8x8 MB partitions, respectively. We will refer to this set as INTER modes.

In the full search (FS) approach, a motion estimation (ME) procedure is carried out for each block size to obtain the best block matching. As stated in [1], H.264 evaluates a cost function, J_{motion}, for each available reference frame (Ref) and motion vector (MV) within the search range:

$$J_{motion} = SAD(MV, Ref) + \lambda_{motion} \cdot R(MV, Ref) \qquad (1)$$

where SAD is the sum of absolute differences between the original and predicted blocks (given MV and Ref), λ_{motion} is the Lagrange multiplier and R is the total amount of bits needed for encoding motion information.

For each mode i, a set of motion vectors $\{MV\}_i$ and reference indexes $\{Ref\}_i$ are selected according to (1). Next, the MD task is carried out by the RDO model. This model aims at minimizing a second cost function, J_{mode}, involving both quality and bit rate terms. Specifically, for each mode i:

$$J_{mode,i} = SSD(\{MV\}_i, \{Ref\}_i, i) + \lambda_{mode} \cdot R(\{MV\}_i, \{Ref\}_i, i) \qquad (2)$$

where SSD is the Sum of Square Differences between the original and the reconstructed MB and λ_{mode} is the Lagrange multiplier. R is the amount of bits needed for coding headers, motion vectors, reference indexes and residual transform coefficients.

When low-complexity applications are considered, the computational cost required to evaluate the RDO function described in (2), which involves the calculation of DCT, quantization, and inverse DCT, becomes unacceptable. An alternative solution consists in making use of the function cost evaluated with (1) in order to compare the different modes with a "non-optimized" R-D model.

Assuming the aforementioned simplification, the next most time consuming operation in a H.264 encoder is ME. Besides using fast ME algorithms, the number of evaluated INTER modes (and thus the number of ME operations) can be reduced by means of a fast MD method.

During the last years, several relevant research works concerning fast MD for H.264 have been published. In [2], the block homogeneity and stationarity are the key factors for mode selection, as in [3], where the movement complexity determines an early termination criterion. In [4] and [5], the evaluated modes are selected according to the monotonicity of J_{mode}; the latter also uses an early termination criterion, which is based on an average value of J_{mode}. Finally, in [6], a scalable fast MD with RDO is proposed based on the occurring probability of modes.

In this paper we propose an algorithm based on J_{motion} cost (1) statistics in order to reduce encoding time, while maintaining the quality as close as possible to the FS approach. This time reduction is achieved by using the non-optimized R-D model, as well as by decreasing the number of evaluated INTER modes. The differences between cost statistics for each mode are used to obtain successive thresholds for early termination.

The rest of the paper is organised as follows: In section 2, the statistical analysis of the motion cost for each mode is described. Section 3 presents the proposed algorithm. In section 4 the experiments and results are presented. Finally, conclusions and further lines are summarized in section 5.

2. STATISTICAL ANALYSIS OF MOTION COST FOR MODE DECISION

In order to reduce the number of ME evaluations, only a sub-set of INTER modes should be explored. For this purpose, the statistical properties of J_{motion} cost for the INTER modes have been studied. Essentially, the joint probability density function of motion cost J is defined as:

$$PDF(J) = \sum_i P(i) \cdot PDF(J_{motion,i} \mid i) \qquad (3)$$

where $P(i)$ is the ith mode occurring probability and $PDF(J_{motion,i} \mid i)$ is the conditional probability density function of motion cost for ith mode, given that i is the winner mode.

To illustrate the analysis, Fig.1 shows the results for B frames in the sequence "Football" with a quantization parameter (QP) value of 40.
The occurring probability for each mode is depicted in the histogram of Fig.1. Conditional PDFs for each mode are presented above as well as the joint PDF (dashed line). The means and standard deviations for two values of QP are shown in Table 1 as an example, for the same sequence "Football". P8x8 is the accumulated cost for 8x8 MB mode and subMB modes (regardless of the subMB mode selected for each block). The motion costs for each particular subMB mode are also listed at the bottom of the table.
As a result of this analysis, some conclusions can be drawn:

- A larger block size implies lower motion costs.
- At MB level, it can be noticed that 16x16, 16x8 and 8x16 modes exhibit similar statistics. Nevertheless, their statistics are usually quite different from those of Direct and P8x8 modes.
- At subMB level, rectangular modes (8x4 and 4x8) can not be jointly considered with 8x8 mode.
- Costs mean and deviation values increase along with QP, just like the occurring probability of larger modes does.
- In statistic terms, J_{motion} cost is larger in P than in B frames.

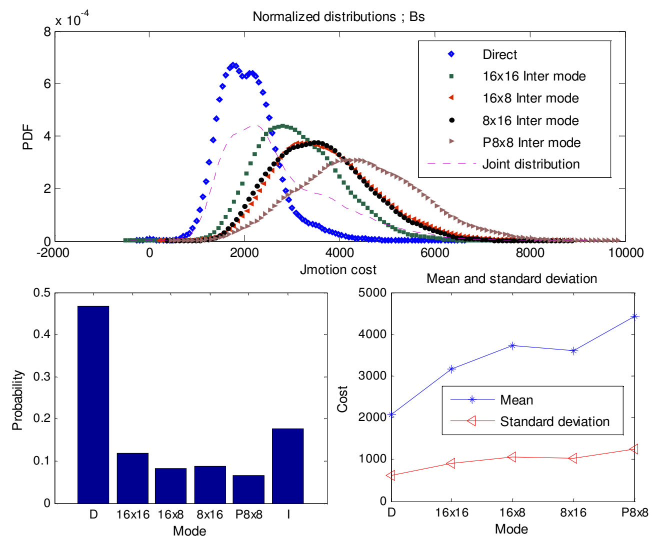

Fig. 1. B frames statistics for "Football" at QP=40.

	QP=32		QP=40	
	Mean	Std	Mean	Std
Direct	1444	261	2081	624
16x16	1968	408	3157	901
16x8	2097	526	3716	1060
8x16	2008	541	3613	1021
P 8x8	2892	862	4435	1241
Dir8x8	352	157	559	262
8x8	553	209	954	393
8x4	838	301	1319	436
4x8	801	311	1254	449
4x4	1136	338	1692	419

Table 1. Detail motion cost analysis for B frames for "Football".

3. PROPOSED ALGORITHM

Based on the analysis of the motion cost described in Section 2, an efficient fast MD algorithm is presented. The explanation will be focused on the case of B frames, but almost the same scheme is applicable to P frames (as can be seen in Fig. 2).

Considering the PDFs shown in Fig.1, it is expected that a first threshold can make a reasonable separation between the Direct mode and the remaining ones (due to the distance between their PDFs). In the same way, it is expected that a second threshold allows to distinguish between “large modes” (Direct, 16x16, 16x8 and 8x8) and subMB modes (Dir8x8, 8x8, 8x4, 4x8 and 4x4). A similar analysis can be performed on subMB modes. In this case, 8x8 can be clearly separated from Dir8x8 and from smaller block sizes (8x4, 4x8 and 4x4 modes).
Summarizing, four thresholds (two at MB level and two at subMB level) can be used in order to early terminate the sequential mode evaluation. The complete algorithm is shown in Fig.2.

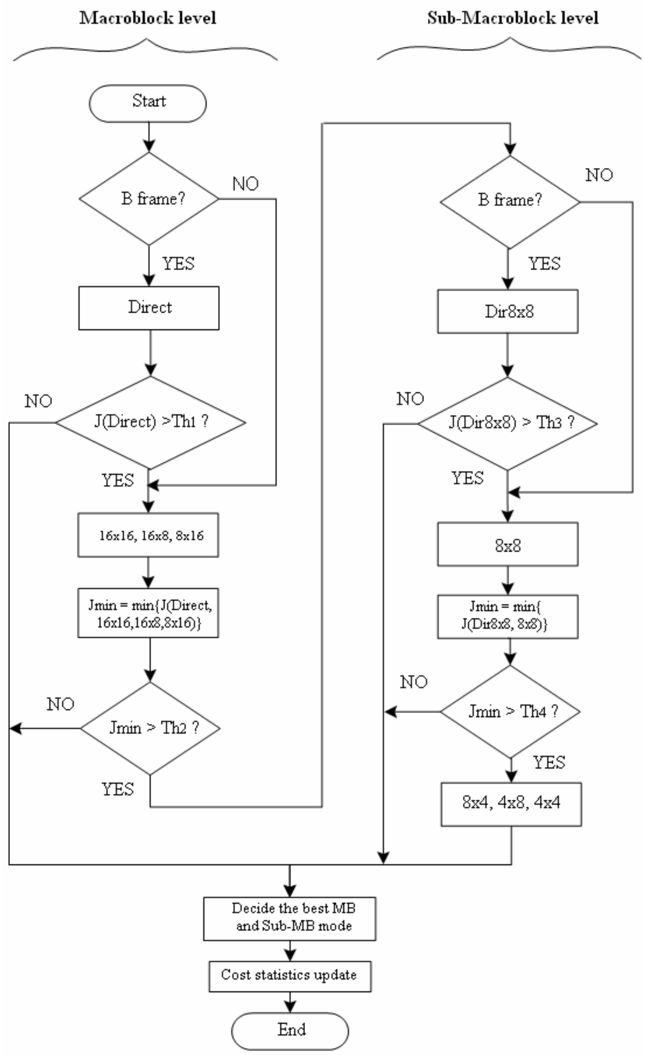

Fig. 2. Algorithm flowchart for MB and subMB mode decision.

The adaptive thresholds are defined for a better control of the tradeoff between complexity and distortion as:

$$Th_k = E - \alpha_k \sigma \tag{4}$$

where E and σ take different values for MB and subMB, and are calculated separately for P and B frames:

$$E_{MB} = \sum_{i=0}^{n_{MB}-1} \frac{1}{n_{MB}} \left(J_{\mathrm{MB}}(i)\right) \qquad \sigma_{MB}^2 = \sqrt{\frac{1}{n_{MB}} \sum_{i=0}^{n_{MB}-1} \left(J_{MB}(i) - E_{MB}\right)^2}$$
$$E_{SubMB} = \sum_{i=0}^{n_{sub}-1} \frac{1}{n_{sub}} \left(J_{\mathrm{SubMB}}(i)\right) \qquad \sigma_{sub}^2 = \sqrt{\frac{1}{n_{sub}} \sum_{i=0}^{n_{sub}-1} \left(J_{sub}(i) - E_{sub}\right)^2} \tag{5}$$

where n_{MB} is the number of times that one of the “large” modes (16x16, 16x8 or 8x16) is selected, and $J_{MB}(i)$ is the achieved minimum for the J_{motion} cost. On the other hand, n_{subMB} is the number of times that 8x8 is selected as the best subMB block size, and $J_{subMB}(i)$ is the J_{motion} cost for the winner mode.

Assuming that sequences have no scene cuts, the mean and standard deviation are updated for each new winner mode along the whole sequence.

Finally, the four thresholds are:

$$\begin{aligned} Th_1 &= E_{MB} - \alpha_1 \sigma_{MB} \\ Th_2 &= E_{MB} - \alpha_2 \sigma_{MB} \\ Th_3 &= E_{subMB} - \alpha_3 \sigma_{subMB} \\ Th_4 &= E_{subMB} - \alpha_4 \sigma_{subMB} \end{aligned} \tag{6}$$

Note that Th_1 and Th_3 are defined only for B frames, whereas Th_2 and Th_4 have the above mentioned definition both for P and B frames. The selection of α_i makes a trade-off between quality and timesaving. Low values of α_i involve a complexity reduction and a certain loss of quality and vice versa. These values were empirically selected.

4. EXPERIMENTS AND RESULTS

4.1 Convergence analysis.

The convergence of motion cost statistics was studied in order to evaluate the accuracy and good performance of the proposed algorithm. Fig.3 is an example of this accuracy test for motion cost in B frames for the sequence “Football” at QP=40. Note that the cost statistics displayed are only updated in B frames in this particular example.

The mean and standard deviation values show a convergence to the reference values obtained in section 2. This means that the proposed algorithm makes almost the same mode selection as the JM10.2 reference software [7], incurring in a very low bit-rate increment.

A similar performance is obtained for the rest of sequences and QPs, and for P frames motion cost statistics.

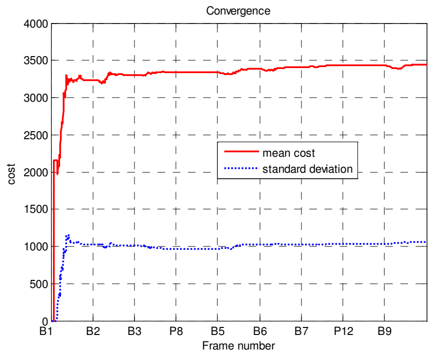

Fig. 3. Mean and standard deviation convergence. B frames statistics in "Football" at QP=40.

4.2 Detailed performance evaluation.

The proposed algorithm has been integrated in the H.264 reference software JM10.2 for the performance evaluation. The test sequences are “Container”, “Paris”, “Mobile”, “Coastguard”, “Football”, “Tempete”, “Akiyo” and “Foreman” in CIF format. The test conditions are as follows: MV search range is ±16 pixels; Hadamard transform is not used; 5 reference frames are used; RD Optimization is disabled; GOP structure is IBBBP; and the number of frames per sequence is 100.

A group of experiments were carried out on the test sequences with four different QP values, namely: 28, 32, 36 and 40. Average PSNR differences *(ΔPSNR)* and average bit rate differences *(ΔBitrate)* between the proposed algorithm and JM10.2 are listed in Table 2. An interpolation is carried out in order to obtain PSNR and Bit Rate differences between R-D curves. Note that shown *ΔPSNR* and *ΔBitrate* are complementary data and must not be considered at the same time. Speed up has been measured in an INTEL Pentium-3Ghz processor with 2GB RAM. *ME_saving* is the ratio between the number of motion estimations carried out by the proposed algorithm and the number of motion estimations carried out by the reference algorithm, and thus, is not processor dependent.

Sequence	*ΔPSNR* (dB)	*ΔBitrate* (%)	Speed-up (%)	ME_saving
Container	-0.04	0.85	75	0.10
Paris	-0.04	0.92	52	0.17
Mobile	-0.10	1.80	49	0.19
Coastguard	-0.01	0.33	46	0.22
Football	-0.02	0.44	42	0.28
Tempete	-0.04	1.02	43	0.25
Akiyo	-0.02	0.22	76	0.09
Foreman	-0.01	0.07	54	0.18

Table 2. Experimental Results

5. CONCLUSIONS AND FURTHER WORK

A novel fast MD algorithm has been proposed for low complexity applications. In this scenario, the MD performed in H.264 even without considering RDO leads to a large number of ME evaluations. The proposed algorithm achieves high timesaving without negligible quality losses.

The results displayed in Table2 show that the proposed algorithm exhibits a different performance depending on video sequence. An interesting further work line consists in adapting the α_i values according to the different contents in video sequences.

Besides, a dynamic adjustment procedure for the thresholds based on the distances between cost distributions of different modes could improve the performance of the proposed algorithm. Also, for sequences with scene cuts, further improvement could be achieved by resetting cost statistics at the instant of each scene change.

6. REFERENCES

[1] T. Wiegand, H.Schwarz, A. Jocj G.J. Sullivan, and, “Rate-Distortion Optimization for Video Compression,” *IEEE Signal Processing Magazine*, pp. 74-90, November 1998.

[2] D. Wu, F. Pan, K.P. Lim, “Fast Intermode Decision in H.264/AVC Video Coding,” *IEEE Transactions on Circuits and Systems for Video Technology,* vol.15, pp. 953-958, July 2005.

[3] M. Yang, and W. Wang, “Fast Macroblock Mode Selection based on Motion Content Classification in H.264/AVC,” *International Conference on Image Processing, ICIP 2004*, pp. 741-744.

[4] P. Yin, H.C. Tourapis, and A.M. Tourapis, “Fast Mode Decision and Motion Estimation for JVT/H.264,” *International Conference on Image Processing, ICIP 2003*.

[5] C. Grecos, and M. Yang, “Fast Mode Prediction for the Baseline and Main Profiles in the H.264 Video Coding Standard,” *IEEE Transactions on Multimedia*, vol. 8, pp. 1125-1134, December 2006.

[6] F. Pan, H. Yu, and Z. Lin, “Scalable Fast Rate-Distortion Optimization for H.264/AVC,” *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1-10.

[7] JVT H.264/AVC reference software v.10.2 available online: http://iphome.hhi.de/suehring/tml/download/old_tml/