COMPLEXITY-CONSTRAINED H.264 HD VIDEO CODING THROUGH MODE RANKING

Tiago A. da Fonseca and Ricardo L. de Queiroz

Universidade de Brasilia
Department of Electrical Engineering
Brasilia, Brazil

ABSTRACT

Video compression is a computation-intensive task and real-time implementation of state-of-the-art video standards is a challenge. In this paper we propose a complexity scalable framework to H.264/AVC prediction module since the "intra" and "inter" prediction steps are responsible for almost all computation complexity. The idea is to employ a subset of prediction modes instead of testing all modes recommended by H.264/AVC standard. Only the most "popular" prediction modes are ranked and tested until reaching a complexity budget. The ranking is done by sampling macroblocks of previous frames. Results show a negligible the quality loss while achieving large computational savings.

***Index Terms*—** H.264/AVC, mode decision, high-definition, complexity scalability, Digital TV.

1. INTRODUCTION

H.264/AVC is the latest international video coding standard [1]. It was jointly developed by the Video Coding Experts Group (VCEG) of the ITU-T and the Moving Picture Experts Group (MPEG) of ISO/IEC. The many small improvements over previous encoding methods led to enhanced coding efficiency for a wide range of applications including video telephony, video conferencing, digital TV, streaming video etc. The H.264/AVC coder has been very well described in the literature [2, 3, 4, 5], showing performance comparisons against other coders and also exploring less known features of the H.264/AVC.

In order to achieve substantial compression, the H.264 encoder tries to make the best possible prediction of the encoding signal in order to spend less bits to represent the residue. When predicting image blocks, its encoder explores temporal (Inter) and spatial (Intra) correlations. "Inter" prediction generates a prediction macroblock from one or more previously encoded video frames using block-based motion estimation and compensation. Important advances from earlier video standards include the support for a range of block sizes (16×16 and down, following a quadtree-like partitioning scheme) and refined motion vectors (quarter-pixel resolution for the luminance component).

In "Intra" prediction, a prediction block is formed based on planar extrapolation of previously encoded and reconstructed neighbouring pixels. The prediction is subtracted from the current block, prior to encoding. A macroblock can be partitioned into blocks of 4×4, 8×8 or 16×16 pixels. The former ones have a total of nine optional prediction modes for the luminance component while the latter has only four modes. The encoder may either select the prediction mode for each block that minimizes the difference between the predicted block and the block to be encoded, or use rate-distortion optimization [6].

Because of all these prediction mode options, the H.264 encoder software implementation is computation-hungry. When encoding high definition sequences, complexity is an issue and real-time video coding is challenging. As the computational complexity of H.264/AVC is mainly concentrated in the prediction stage, trying to reduce the complexity of this stage seems to be the most efficient path towards reducing the overall computational costs. This work suggests a strategy where the suppression of least frequent prediction modes is used to control the complexity.

2. H.264/AVC COMPLEXITY PROFILE

As a first step, we profiled the H.264/AVC (*High Profile*) encoder complexity. We used `gprof`[1] and JM13.2[2] to encode high definition video sequences, with rate-distortion optimization turned on, four frames reference buffer and *UMHexS* motion estimation [7]. Results are presented in Table 1.

Complexity estimates similar to those in [8] are presented in Table 1, where different motion search window sizes and motion estimation techniques were applied. We observe that the encoder spends great part of the execution time doing "Inter" prediction, due to the extensive tests required to find the best match.

This work was supported by HP Brasil and by Conselho Nacional de Desenvolvimento Cientifico e Tecnologico, CNPq, Brasil, under grant 473696/2007-0.

[1] http://www.gnu.org/software/binutils/

[2] JM Available: http://iphome.hhi.de/suehring/tml/

Table 1. Relative computational complexity for HD Pedestrian Area in JM13.2 H.264/AVC *High Profile* coding and various motion estimation search-window sizes.

	Window Size (pixels)/QP			
Coding Stage	**64**/16	**64**/28	**64**/36	**144**/28
Inter Prediction	56,7%	66,4%	69,1%	72,2%
4×4 Intra Prediction	10,1%	7,0%	5,9%	5,7%
8×8 Intra Prediction	7,2%	5,1%	4,5%	4,2%
16×16 Intra Prediction	1,1%	1,2%	1,2%	1,1%
Other Stages	24,9%	20,3%	19,3%	16,8%
Total	100,0%			

2.1. Complexity Reduction Techniques

As shown in [8] and in Table 1, the encoder complexity distribution indicates that complexity reduction can be most efficiently achieved through simplifying the prediction module, particularly the motion estimation step.

Sub-optimal fast motion estimation techniques were proposed [7, 9] and incorporated in the H.264/AVC reference software. In exploring the variety of macroblock partitions available in H.264/AVC, there are works [10, 11] that only carry motion estimation for the most probable partition. The early prediction of skipped macroblocks prior to motion estimation [12, 13] is also used for complexity control. Intra-prediction tests can also be reduced by means of selecting the most probable best mode according to heuristics [14, 15]. Composition of previously proposed complexity reduction techniques were experienced and implemented in H.264, achieving substantial computational savings without a complexity control mechanism [16]. An effective approach is to generalize the rate-distortion analysis to add a complexity optimization variable. This concept is well suited to the emerging field of wireless digital video communications, where energy and delay constraints are stringent [17, 18, 19].

2.2. Prediction Mode Bias

The H.264/AVC prediction stage is rather complex due to the many tests of various prediction modes available to each macroblock. However, when compressing high definition 1080p video sequences (1920×1080 pixels per frame), we verify that the prediction modes applied to encode the signals become concentrated in small classes. The frequency profile of selected prediction modes for Pedestrian Area sequence at different resolutions, ranging from QCIF (176×144 pixels) to 1080p (1920×1080 pixels), is shown in Fig. 1. Each graph color represents a prediction mode [1] and P≤8×8 incorporates all motion compensated prediction modes where the block size is smaller or equal to 8×8. These modes were grouped due their low occurrency frequency in HD encoding.

We can observe for this sequence that when we increase the resolution, prediction modes tend to polarize themselves around bigger macroblock partitions. Profiles were also built for other sequences [20] and the general behavior suggests that some computational effort could be saved when encod-

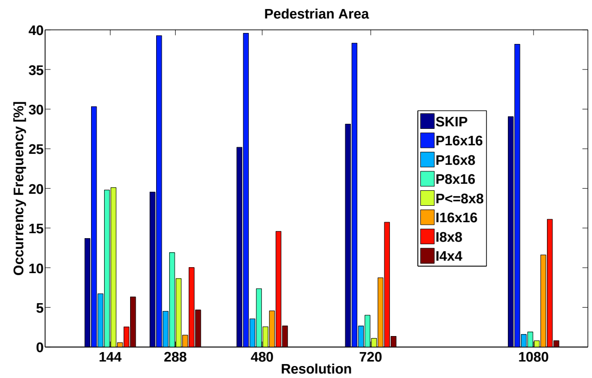

Fig. 1. Prediction modes occurrency frequency × resolution for Pedestrian Area sequence.

ing high definition video sequences by avoiding less frequent macroblock prediction modes.

3. COMPLEXITY CONTROL MODEL

In order to control the encoding time according to the user adjustments, the complexity is scaled based on the suppression of less frequent prediction modes using the framework shown in Fig. 2. We start by randomly selecting macroblocks to preview the prediction modes frequency distribuition of the next frame. Then, we select the dominant modes, i.e. the set of modes which corresponds to the desired complexity C. These modes will be used for the next frame. We use the following algorithm:

Let each frame have $\boldsymbol{N}$ macroblocks. For the $\boldsymbol{n}$-th P- or B-frame

1. Randomly select a set of $\boldsymbol{S}$ of $\boldsymbol{N_S}$ macroblocks of the $\boldsymbol{n}$-th frame. The remaining $\boldsymbol{N - N_S}$ macroblocks form the complement set $\boldsymbol{S'}$.
2. Test all prediction modes for macroblocks in set $\boldsymbol{S}$, in order to pick the best mode.
3. Test only dominant modes for macroblocks in $\boldsymbol{S'}$.
4. Set dominant mode set $\boldsymbol{D}$ as empty.
5. Rank best modes in set $\boldsymbol{S}$.
6. Assign the most chosen modes to $\boldsymbol{D}$ while the accumulated complexity is within a desired complexity budget $\boldsymbol{C}$.
7. Set $\boldsymbol{n \leftarrow n + 1}$ and repeat.

Even though the best mode frequency distribution is not stationary, tests have shown that this is a good approximation for adjacent frames [20]. Errors in finding the dominant modes will be reflected by a small degradation on the encoder RD performance.

An important issue is the sampling population size, N_S, which will be used in the prediction of next frame dominant modes. The smaller N_S, the larger the savings, but the worse the performance. In [20], we can verify that a sampling population of $N_S = 0.10N$ can be choosen without incurring in great rate-distortion burden.

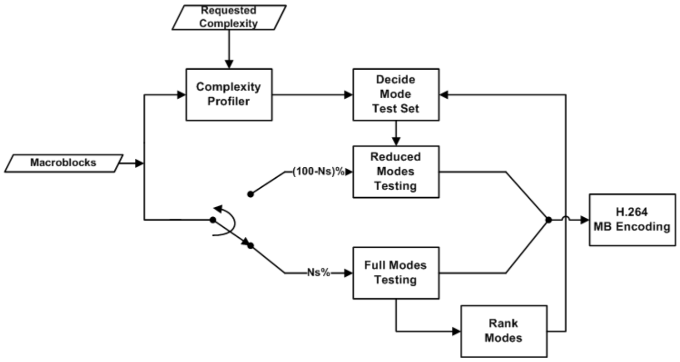

Fig. 2. Complexity control framework.

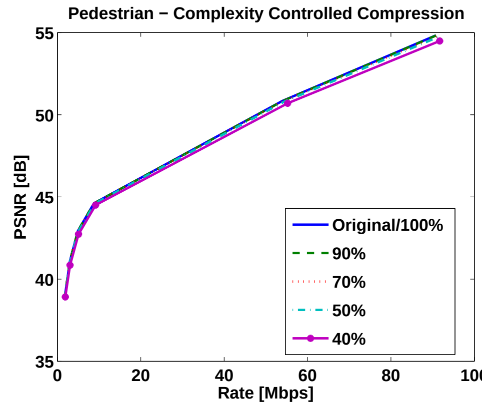

Fig. 3. Different complexity settings rate-distortion curves for sequence "Pedestrian Area". The curves are nearly co-located.

3.1. Rate-distortion-complexity Optimization

Step **6** in the algorithm in Section 3 requires an rate-distortion-complexity optimization approach. Let I_m be the statistical relevance of a prediction mode, i.e. its occurrency frequency in a frame, and C_m its encoding complexity, measured as the average encoding time. For each frame, the complexity control model maximizes a profit function described as

$$\overline{\lambda} = argmax_{\lambda_m} \sum_{m} \lambda_m \cdot I_m \qquad (1)$$

where $\lambda_m \in \{0, 1\}$, $\lambda_m = 1$ represents the choice mode m, $\lambda_m = 0$ its supression, and m stands for the prediction modes:

$$m \in \left\{ \begin{array}{cc} \text{P16×16} & \text{I16MB} \\ \text{P16×8} & \text{I8MB} \\ \text{P8×16} & \text{I4MB} \\ \multicolumn{2}{c}{\text{P≤8×8}} \end{array} \right\}.$$

Note that Eq. 1 has to be constrained to meet the user required complexity C,

$$\sum_{m} \lambda_m \cdot C_m \leq C. \qquad (2)$$

The obtained $\overline{\lambda}$ will be a set of seven binary values which will indicate the selected prediction modes to be applied to the next frame to be encoded.

4. EXPERIMENTAL RESULTS

The proposed modification was implemented in the H.264 reference software JM13.2. The first 50 frames of each sequence were used. *UMHexS* motion estimation was employed and the results were obtained by varying the QP (quantization parameter) over the range $12 \leq QP \leq 36$. Within this range, we respect HD video constraints of quality and rate for broadcasting [4]. Fig. 3 presents rate-distortion, RD, perfomance curves for Pedestrian Area sequence, for different complexity settings: Original (100% of encoding complexity), 90% of encoding complexity, 70% of encoding complexity and so on. We observe that the modified codec performance is very close to the reference verification model in such a way that we can not readly perceive the diferences between them.

A more detailed approach is to plot the average difference between the performance curves for different sampling population sizes, as show in Figs. 4 and 5. The average PSNR and bitrate differences between RD-curves were evaluated as described in [21].

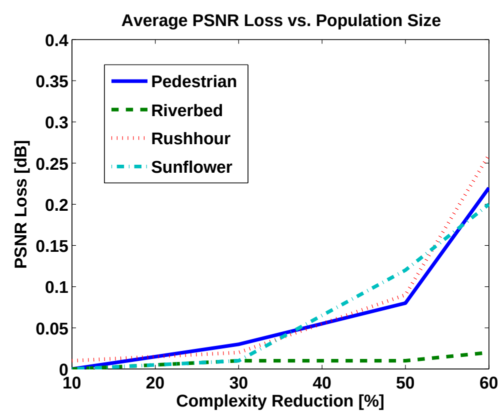

Fig. 4. Average PSNR loss *vs.* complexity reduction for different HD video sequences.

We can observe a very small quality loss when predicting through dominant modes. This is due to eventual mismatches between the best prediction mode evaluated by the two methods. The rate loss is small (below 5%) if the complexity savings remains bellow 50%. For the particular case where the intra-predicted macroblocks are dominant, like in "Riverbed" sequence [20], the computational savings are greater due to the fact that motion estimated prediction modes are not included in dominant set for some frames. This sequence is very challenging because of the difficulty in getting good matches through motion estimation. Nevertheless, this characteristic was properly "tracked" by our algorithm.

5. CONCLUSIONS

We proposed a complexity-controlled method to carry the prediction mode tests in H.264/AVC for high-definition se-

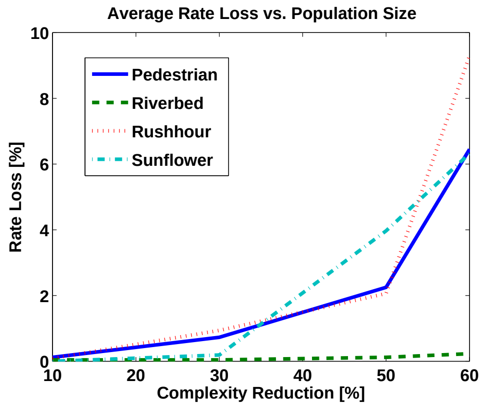

Fig. 5. Average rate loss *vs.* complexity reduction for different HD video sequences.

quences. Rather than testing all prediction modes available, we search for a "dominant" mode subset in a complexity-constrained framework. The dominant mode set is computed in a reduced sampled population of macroblocks and is inherited from a frame to the next. All modes are profiled and the encoder picks those most popular modes until a complexity budget is met. Our tests have shown that the RD performance is barely affected by the prediction mode test skipping, while achieving significant complexity reduction. The method does not require a new decoder implementation because only non-normative codec aspects are modified.

6. REFERENCES

[1] JVT of ISO/IEC MPEG and ITU-T VCEG, "Advanced video coding for generic audiovisual services," Tech. Rep. 14496-10:2005, March 2005.

[2] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons Ltd, 2003.

[3] T. Wiegand, G. J. Sullivan, G. Bjntergaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[4] G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions," *Proc. SPIE Conference on Applications of Digital Image Processing XXVII, Special Session on Advances in the New Emerging Standard: H.264/AVC*, August 2004.

[5] R. L. de Queiroz, R. S. Ortis, A. Zaghetto, and T. A. Fonseca, "Fringe Benefits of the H.264/AVC," in *International Telecommunication Symposium*, 2006, pp. 208–212.

[6] 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 and Systems Magazine*, pp. 7–28, 2004.

[7] Z. Chen, P. Zhou, and Y. He, "Hybrid Unsymmetrical-cross Multi-Hexagon-grid Search Strategy for Integer Pel Motion Estimation in H.264," *Picture Coding Symposium*, April 2003.

[8] Y.-Y. Huang, B.-Y. Hsieh, S.-Y. Chien, S.-Y. Ma, and L.-G. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 4, pp. 507–522, April 2006.

[9] H.-Y. C. Tourapis, A. M. Tourapis, and P. Topiwala, "Fast Motion Estimation within the H.264 Codec," *Proc. of International Conference on Multimedia and Expo. ICME*, vol. 3, pp. 517–520, July 2003.

[10] T.-Y. Kuo and C.-H. Chan, "Fast Variable Block Size Motion Estimation for H.264 Using Likelihood and Correlation of Motion Field," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 10, pp. 1185–1195, October 2006.

[11] B.G. Kim, S.-K. Song, and C.-S. Cho, "Efficient Inter-Mode Decision Based on Contextual Prediction for the P-Slice in H.264/AVC Video Coding," *IEEE International Conference on Image Processing*, pp. 1333–1336, September 2006.

[12] C. S. Kannangara, I.E.G. Richardson, M. Bystrom, J. Solera, Y. Zhao, A. MacLennan, and R. Cooney, "Complexity Reduction of H.264 Using Lagrange Multiplier Methods," *Proc. IEEE Int. Conf. on Visual Information Engineering*, April 2005.

[13] C. S. Kannangara, Y. Zhao, I. E. Richardson, and M. Bystrom, "Complexity Control of H.264 Based on a Bayesian Framework," *Picture Coding Symposium*, November 2007.

[14] B. La, M. Eom, and Y. Choe, "Fast Mode Decision for Intra Prediction in H.264/AVC Encoder," *IEEE International Conference on Image Processing*, vol. V, pp. 321–324, September 2007.

[15] C. Hwang, S.S. Zhuang, and S.-H. Lai, "Efficient Intra Mode Selection Using Image Structure Tensor for H.264/AVC," *IEEE International Conference on Image Processing*, vol. V, pp. 289–292, September 2007.

[16] R. Almeida and F. Pereira, "H.264/AVC low complexity encoding: a novel fast coding mode selection solution," in *Proc Conf. on Telecommunications - ConfTele*, Peniche, Portugal, May 2007.

[17] Z. He, Y. Liang, L. Chen, I. Ahmad, and D. Wu, "Power-Rate-Distortion Analysis for Wireless Video Communication Under Energy Constraints," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 645–658, May 2005.

[18] E. Akyol, D. Mukherjee, and Y. Liu, "Complexity Control for Real-Time Video Coding," *IEEE International Conference on Image Processing*, vol. I, pp. 77–80, September 2007.

[19] R. Vanam, E. A. Riskin, S. S. Hemami, and R. E. Ladner, "Distortion-complexity optimization of the h.264/mpeg-4 avc encoder using the gbfos algorithm," in *DCC '07: Proceedings of the 2007 Data Compression Conference*, Washington, DC, USA, 2007, pp. 303–312, IEEE Computer Society.

[20] T. A. da Fonseca and R. L. de Queiroz, "Complexity reduction techniques applied to the compression of high definition sequences in digital TV," *XXVI Simposio Brasileiro de Telecomunicacoes*, pp. 43–47, September 2008.

[21] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves.," *Coc. VCEG-M33*, April 2001.