Open-Loop Prediction in H.264/AVC for High Definition Sequences

Tiago A. da Fonseca[+], Yuxin Liu[*] and Ricardo L. de Queiroz[+]

Abstract— **H.264/AVC is the newest, state-of-the-art, video compression standard. It leads to substantial performance improvement compared to other existing standards. However, like other video standards, it is a hybrid predictive-transform coder. In this paper we propose a more efficient approach to implement the prediction stage. The idea is to employ original data rather than reconstructed ones to perform prediction tests before choosing the best mode. The residue, however, is evaluated using previously decoded data to avoid drifting. Results show that, for high definition sequences, the quality loss is negligible. We are, thus, able to parallelize the intra and inter-prediction parts, which are the most time consuming operations in H.264/AVC.**

Keywords— **H.264/AVC, mode decision, high-definition, parallelization.**

I. INTRODUCTION

H.264/AVC is the latest international video coding standard [1]. Is 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 added up and promoted 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 well described in the literature [2]-[5], showing performance comparisons against other coders and also exploring less known features of the H.264/AVC.

In this paper, we propose a different way to implement the prediction module of H.264/AVC codec that does not depend on locally reconstructed data and allows for parallelism of implementation.

II. MACROBLOCK PREDICTION IN H.264/AVC

H.264/AVC is a hybrid video codec, i.e. along with a transform module, it has a prediction module, a differential stage and a feedback loop [6]. In order to perform prediction, a DPCM coder, like in any recursive lossy coding scheme, uses previously reconstructed samples (it must reproduce the decoder locally) as input to the prediction model. This avoids mismatching between coder and decoder data, and allows for perfect decoding. However, H.264/AVC has a rather complex prediction stage composed by a set of prediction models.

In Fig. 1, we can see the prediction stage highlighted in a H.264/AVC encoder block diagram. Note that it can be divided into temporal (Inter) and spatial (Intra) models.

"Inter" prediction generates a prediction macroblock from one or more previously encoded video frames using block-based motion estimation and compensation. This model is responsible for almost 90% of the complexity of an H.264/AVC baseline encoder [7]. Important advances from earlier video standards include the support for a range of block sizes (16x16 and down, in Fig. 2) and refined motion vectors (quarter-sample resolution for the luminance component).
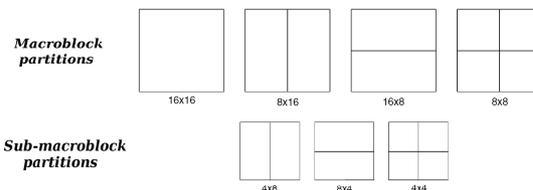

Fig. 2.   Macroblock and submacroblock partitions for motion compensation in Inter Prediction.

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 in 4x4, 8x8 or 16x16 pixels. The former ones have a total of nine optional prediction modes for luminance while the latter has only four modes as illustrated by Fig. 3. The encoder typically selects the prediction mode for each block that minimizes the difference between the predicted block and the block to be encoded.
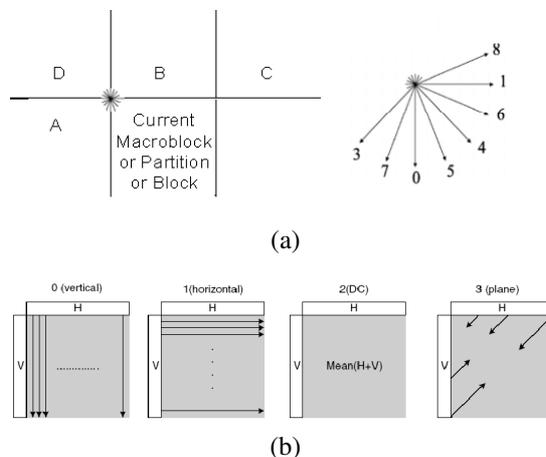

(a)

(b)

Fig. 3.   Intra prediction modes and their respective planar extrapolation directions for (a) 4x4/8x8 blocks and (b) 16x16 blocks. In (a), mode 2 is DC prediction.
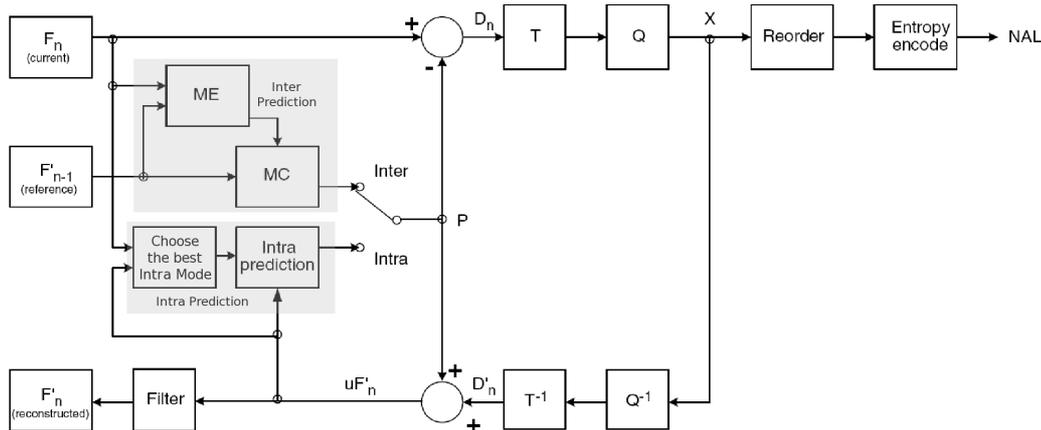
Fig. 1.   H.264/AVC Encoder block diagram. Note the prediction stages.

A prediction for the current macroblock is created from image samples that have already been encoded either in the same frame or in a previously encoded one. This prediction is subtracted from the current macroblock and the residual is compressed and transmited, together with information required for the decoder to repeat the prediction process (motion vectors, prediction modes, etc.). The decoder creates an identical prediction and adds it to the decoded residual or block. The encoder bases its prediction on encoded and decoded image samples (rather than on original video frame samples) in order to ensure that the encoder and decoder predictions are identical.

III. PREDICTION BASED ON ORIGINAL DATA

As illustrated in Fig. 1, before deciding on the actual prediction, the encoder performs many tests to choose either the best motion vector or the best intra prediction mode.

If we regard H.264/AVC in the HD (high definition) video scenario, only a low level of distortion is tolerated which implies a high degree of similarity between the compressed video and its original version. So, if we employ original data in prediction tests instead of reconstructed ones, it is likely that the best prediction mode chosen employing original data will be the same mode chosen when using reconstructed data. The proposed method (see Fig. 4) consists in applying original data to choose the best prediction mode but locally decoded data to do compensation in order to avoid drifting. All data/modes used to evaluate the residue are available to the decoder. If the best mode found employing original data was different from the "real" one, the residue coding will consume more rate, but distortion will be governed by quantization.

In Fig. 4, the prediction tests do not depend on reconstructed/decoded data as shown on Fig. 1 but depend only on original data. The search for the best prediction mode, the most time consuming stage of an H.264/AVC encoder, can be parallelized given that we do not need to wait for previously encoded/decoded data, thus "opening" the prediction loop (see Fig. 4).

Now, consider an engine responsible for the prediction tests process which is capable to analyze a reference frame
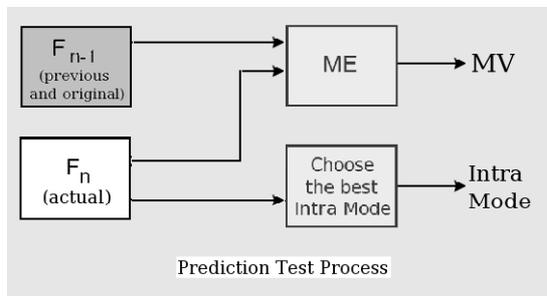
(a)
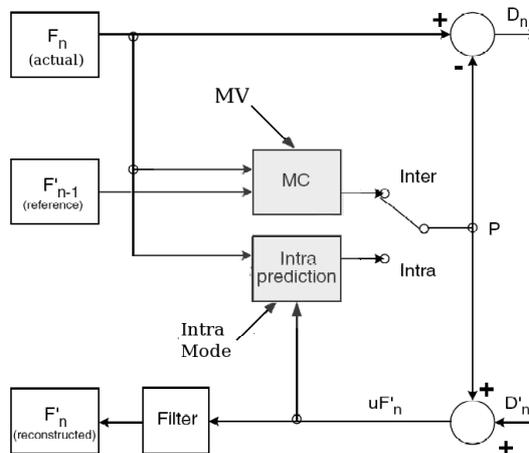
(b)

Fig. 4.   Proposed parallel prediction structure.

in Δt seconds. If there are n of them working in parallel as suggested, one can spend Δt seconds to find the best prediction mode for n reference frames simultaneously.

IV. EXPERIMENTAL RESULTS

The proposed modification was implemented in X264[1], a free H.264/AVC "High" profile implementation. Besides being

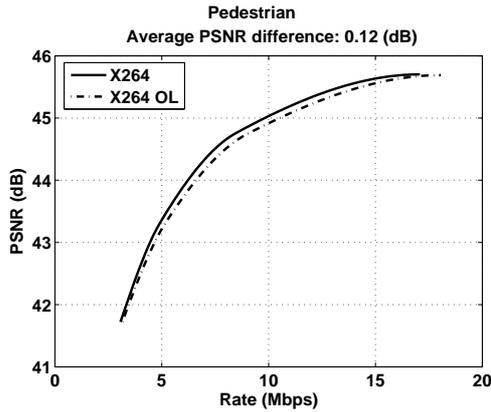[1]X264 Available: http://www.videolan.org/developers/x264.html

Fig. 5.   Rate-distortion curves for Pedestrian Area: X264 vs. X264-OL
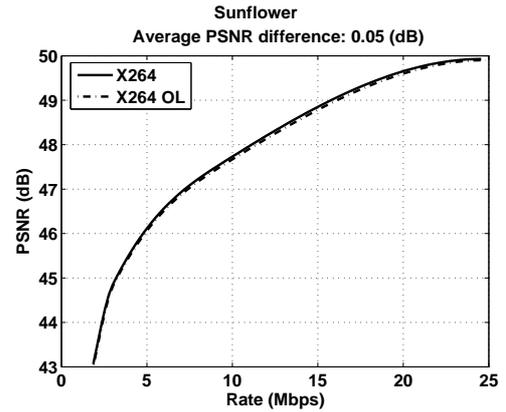

Fig. 7.   Rate-distortion curves for Sun Flower: X264 vs. X264-OL
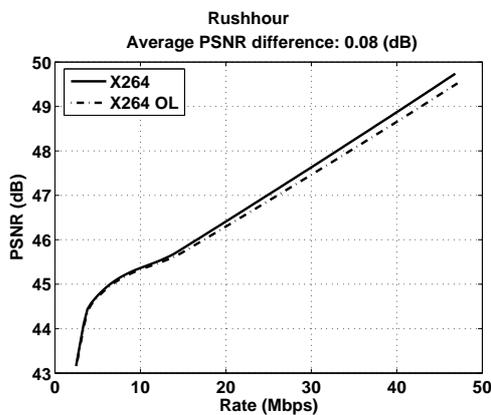

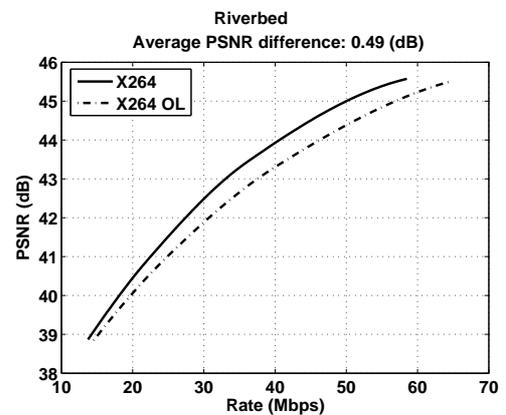Fig. 6.   Rate-distortion curves for Rush Hour: X264 vs. X264-OL

Fig. 8.   Rate-distortion curves for Riverbed: X264 vs. X264-OL

an open codec, it has mature rate-distortion behavior and has interesting features like architecture-optimized implementation.

Standard test sequences were employed:

- Pedestrian Area (1920x1080, 25 fps, progressive) is a shot of a pedestrian area. The camera is static at a low position while pedestrians pass by.
- Sunflower (1920x1080, 25 fps, progressive) is a very detailed shot. There is a bee at the sunflower, with small color differences. The camera is fixed and the scene has slow global motion.
- Rush-hour (1920x1080, 25 fps, progressive) is a shot of rush-hour in Munich. There are many cars moving slowly, a high depth of focus and the camera is fixed.
- Riverbed (1920x1080, 25 fps, progressive) is a shot of a riverbed seen through the water. Challenging compression.

Each sequence was made up of 50 frames and the results were obtained by varying the QP over the range (8,36). Within this range, we somewhat respect HD video constraints of quality and rate for broadcasting. We present plots in Figs. 5 through 7, each with two rate-distortion curves: one for the original codec (X264) and the other for the modified codec (X264-OL). The average PSNR differences between RD-curves were evaluated as described in [8].

We can observe a negligible quality loss when using original data rather than reconstructed ones in the X264 prediction step.

This is due to eventual mismatch between the best prediction mode evaluated by the two methods. We can observe a major difference only for the Riverbed sequence due to its chaotic motion field and highly complex textures (reflective water effects are common), which result in larger mismatch between modes calculates employing original and reconstructed data.

V. CONCLUSIONS

We propose a different way of carrying the prediction mode tests in H.264/AVC for high-definition sequences. Rather than using previously decoded macroblocks, we propose to use the original macroblocks. In other words, we open the prediction loop. The original image data is just used for the prediction mode decision, while the residue is found using locally decoded data. Hence, drifting is avoided. In tests with broadcast quality coding of HD sequences, the performance loss is negligible in most cases. Its main advantage is that it allows for the parallelization of the implementation since all predicition modes and motion vectors can be tested simultaneously. Hence, encoding can be made faster. Future work will concentrate on estimating and correcting possible failure modes.

REFERENCES

[1] Final Draft International Standard of Joint Video Specification, ITU-T Recommendation H.264, ISO/IEC 14496-10 AVC, March 2003.

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

[3] T. Wiegand, G. J. Sullivan, G. Bjøntergaard, 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 et al., "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. da Fonseca, "Fringe benefits of the H.264/AVC", *Proc. of Intl. Telecom. Symp.*, Fortaleza, Brazil, pp. 208-212, Sep. 2006.

[6] K. Sayood, *Introduction to Data Compression*, 2nd edition, Morgan Kaufmann Publishers, 2000.

[7] 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 Trans. on Circuits and Systems for Video Technology*, Vol 16, No. 4, pp. 507-522, April 2006.

[8] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves.", Coc. VCEG-M33, Apr. 2001.