Least-Squares Directional Intra Prediction in H.264/AVC

Diogo Caetano Garcia and Ricardo L. de Queiroz, *Senior Member, IEEE*

***Abstract*—A new intra-prediction mode for the H.264/AVC standard is proposed. Each pixel within a block is predicted by a weighted sum of its neighbours, according to an Nth order Markov linear model. The weights are obtained through a least-squares estimate from reconstructed data in the neighbouring blocks, so that no overhead is necessary to convey the weights to the decoder. Results show significant improvements in H.264/AVC compression for images that are rich in directional structures, and moderate improvements for the other images and sequences tested.**

***Index Terms*—H.264/AVC intra prediction, image coding, least square prediction.**

I. Introduction

The H.264/AVC standard [1] represents the state-of-the-art in video coding, outperforming previous standards such as MPEG-2 and MPEG-4 Part 2 [2]. Moreover, H.264/AVC introduced new intra-prediction methods which, allied to other innovations (e.g. context-adaptive entropy encoding and in-loop deblocking filter), offer a still image coding performance that is comparable or superior to the JPEG and JPEG2000 image coding standards [3].

The H.264/AVC intra-prediction methods are directional in nature. There are different block sizes, each having different number of modes and making use of up to 4 neighbouring blocks for prediction. Despite this great encoding flexibility, which adapts itself to the image characteristics, only the block's surrounding pixels are used for prediction.

In this letter, a new intra-prediction mode for H.264/AVC is proposed, based on a backward-adaptive linear predictor [4]. Previous image and video coding schemes used similar approaches for prediction [5]–[10]. Here, however, we apply the prediction model directly in the H.264/AVC intra prediction stage [11]–[15].

II. Proposed Intra-Prediction Method

The proposed method assumes that, for all pixel positions $\mathbf{n}$ within an image, each pixel $X(\mathbf{n})$ has a linear prediction $\hat{X}(\mathbf{n})$ which is found using an Nth order Markov model, i.e.

$$\hat{X}(\mathbf{n}) = \sum_{i=1}^{N} a_i X(\mathbf{n} - \mathbf{n}_i) \tag{1}$$

Manuscript received May 19, 2010; revised July 07, 2010; accepted July 08, 2010. Date of publication July 19, 2010; date of current version August 09, 2010. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Lap-Pui Chau.

D. C. Garcia is with the Department of Electrical and Computer Engineering, Universidade de Brasília, Brasília, Brazil (e-mail: diogo@image.unb.br).

R. L. de Queiroz is with the Department of Computer Science, Universidade de Brasília, Brasília, Brazil (e-mail: queiroz@ieee.org).

Digital Object Identifier 10.1109/LSP.2010.2060192

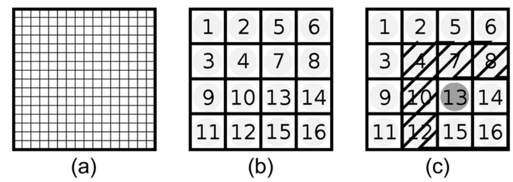

Fig. 1. H.264/AVC zig-zag intra prediction for 4 × 4 blocks: (a) current macroblock (16 × 16 pixels), (b) subdivision in 4 × 4 blocks and corresponding encoding order, and (c) available reference blocks for the 13th encoded block.

where $\{\mathbf{n}_i\}$ represent the offset positions of a causal prediction window of neighbouring pixels, and $\{a_i\}$ are their corresponding weights.

Before (1) can be applied to predict pixels in the image to be encoded, the weights must be computed. To do so, a training window is defined, composed of previously reconstructed pixels. We assume that the model fits the data in the training window, where a given pixel $X(\mathbf{n})$ corresponding to a prediction $\hat{X}(\mathbf{n})$ is associated to a vector $\mathbf{x}_{pred}$ of $M \times 1$ elements, and $\{X(\mathbf{n}-\mathbf{n}_i)\}$ are the pixel's causal neighbours, composing a matrix $\mathbf{x}_{neigh}$ of $M \times N$ elements. Arranging the weights $\{a_i\}$ in an $N \times 1$ vector, $\mathbf{a}$, we obtain

$$\mathbf{x}_{pred} = \mathbf{x}_{neigh}\mathbf{a}. \tag{2}$$

We perform a linear regression of (2), such that $\mathbf{a}$ minimizes the error in this model in a least-squares sense, i.e.

$$\hat{\mathbf{a}} = \left(\mathbf{x}_{neigh}^T \mathbf{x}_{neigh}\right)^{-1} \mathbf{x}_{neigh}^T \mathbf{x}_{pred}. \tag{3}$$

The training window and the prediction window ($\{\mathbf{n}_i\}$, M and N) may change from region to region, as we will discuss later.

In H.264/AVC, for 4 × 4- and 8 × 8-pixel blocks, intra prediction is performed in a zig-zag fashion inside the macroblock (a 16 × 16-pixel block). Fig. 1(a) represents a macroblock, and Fig. 1(b) shows the encoding order for 4 × 4 blocks. For each block to be encoded, there are different neighbourhoods with reconstructed data. For example, Fig. 1(c) shows the available reference blocks (containing reconstructed data) for the 13th block inside the macroblock.

Fig. 2 summarizes all possible reference-block configurations, and Table I relates these configurations to the encoding order of 4 × 4 and 8 × 8 blocks, regardless of whether they occur inside or outside of the current macroblock. Note that not all of these reference blocks are used for prediction in regular H.264/AVC intra mode, and our training window uses as many reference blocks as available.

The proposed model predicts entire blocks of 4 × 4 and 8 × 8 pixels, as does H.264/AVC intra prediction. In order to do so, (1) is applied to each pixel of a block at a time, where $\{X(\mathbf{n}-\mathbf{n}_i)\}$ represent reconstructed pixels within a causal neighbourhood,

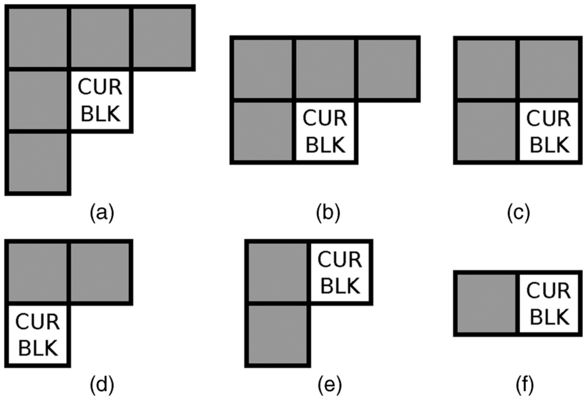

Fig. 2. Possible cases of available reference blocks (painted in grey) throughout H.264/AVC's zig-zag intra prediction, for 4 × 4 and 8 × 8 blocks (*CUR BLK*).

TABLE I
OCCURRENCE OF THE SIX CASES OF AVAILABLE REFERENCE BLOCKS (FIG. 2) ACCORDING TO THE ENCODING ORDER FOR 4 × 4 AND 8 × 8 BLOCKS

CASE	4x4 CODING ORDER	8x8 CODING ORDER
(a)	1, 3, 5, 9 and 13	1
(b)	2, 6, 7, 10, 11 and 15	2 and 3
(c)	4, 6, 8, 12, 14 and 16	4
(d)	1, 3, 9 and 11	1, 3
(e)	1 and 5	1
(f)	2 and 6	2

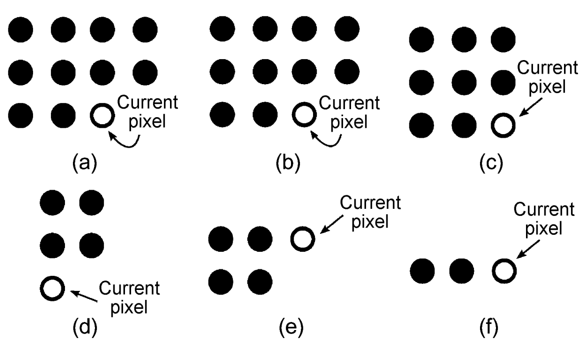

Fig. 3. Pixels used for prediction, according to the available reference blocks depicted in Fig. 2.

relative to the pixel to be predicted. If the reconstructed value is not available, its predicted value is used. Fig. 3 depicts the possible set $\{\mathbf{n}_i\}$ of predictors, according to the available reference blocks. Once computed, the same set of weights $\{a_i\}$ is used throughout the whole block. Within a block, we process pixels in row-major order, with the exception of case in Figs. 2(e) and 3(e), in which case we follow a collumn-major order.

As the set of weights $\{a_i\}$ minimize the model error in a least-squares sense, pixels that are unsuited for prediction in a given block receive smaller weights. In this manner, the proposed prediction model adapts itself to the local characteristics of each image.

Since prediction is carried using information that is available to both the encoder and the decoder, our backward-adaptive approach requires no side information to transmit the weights to the decoder. We have opted to include our directional adaptive mode to replace one of H.264/AVC's intra-prediction modes. For both 4 × 4 and 8 × 8 blocks, the "horizontal-up" mode is replaced, which is the last mode in the H.264/AVC prediction table, making it the least likely to be chosen. Results show that moving up the proposed prediction method on the table incurs in negligible gains. In the cases where (3) cannot be solved, due to non-invertible matrices that are bound to appear, the original "horizontal-up" mode is used. Hence, the choice for adaptive directional prediction is made on a block basis and is encoded just as is regular intra-prediction mode selection.

The whole prediction process can be summarized as

for all 4 × 4 and 8 × 8 blocks **do**

  Determine from Figs. 2 and 3 the M-pixel positions of available reference blocks $(\mathbf{blk}_{ref})$ and the N-pixel predictor set $(\mathbf{set}_{pred})$

  Arrange available reference blocks into an $M \times 1$ vector $(\mathbf{x}_{pred})$

  for $i = 1$ to M **do**

    for $j = 1$ to N **do**

      $\mathbf{x}_{neigh}(i,j) = \mathbf{X}(\mathbf{blk}_{ref}(i) - \mathbf{set}_{pred}(j))$

    end for

  end for

  $\hat{\mathbf{a}} = (\mathbf{x}_{neigh}^T \mathbf{x}_{neigh})^{-1} \mathbf{x}_{neigh}^T \mathbf{x}_{pred}$

  for all pixel positions $\mathbf{n}$ in current block **do**

    $\hat{X}(\mathbf{n}) = 0$

    for $i = 1$ to N **do**

      if $X(\mathbf{n} - \mathbf{set}_{pred}(i))$ is available **then**

        $\hat{X}(\mathbf{n}) = \hat{X}(\mathbf{n}) + \hat{\mathbf{a}}(i) X(\mathbf{n} - \mathbf{set}_{pred}(i))$

      else

        $\hat{X}(\mathbf{n}) = \hat{X}(\mathbf{n}) + \hat{\mathbf{a}}(i) \hat{X}(\mathbf{n} - \mathbf{set}_{pred}(i))$

      end if

    end for

  end for

end for

III. EXPERIMENTAL RESULTS

The proposed intra-prediction mode was implemented in the H.264/AVC reference software JM10.2. All results were evaluated using the luminance component, and enabling CABAC (context-adaptive binary arithmetic coding), in-loop deblocking filter, and rate-distortion (RD) optimization. QP values of 22, 27, 32, and 37 were used to generate the RD performance curves for the 2048 × 2560-pixel image *Bike*, as well as for the 512 × 512-pixel images *Barbara*, *Lena*, *Monarch*, *Peppers* and *Spoke* (part of the *Bike* image, as shown in Fig. 4). We also ran tests with these QP values for 300 frames of CIF-resolution sequences *Foreman*, *Mother and Daughter* and *Paris*, for 260

Fig. 4. Test image *Spoke*.

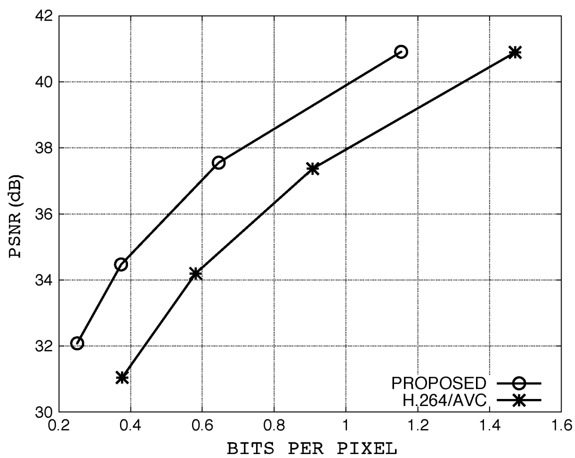

Fig. 5. RD performance for the luminance component of image *Spoke*.

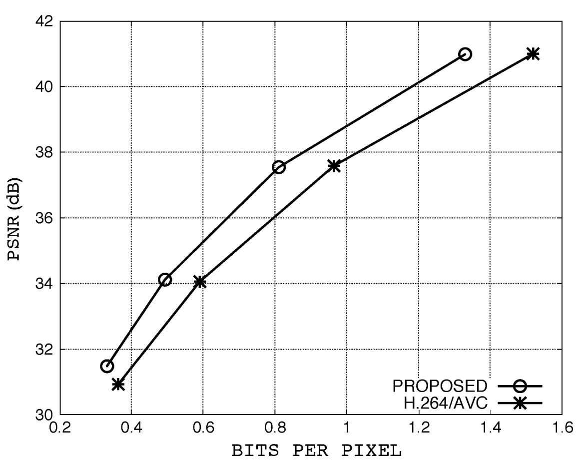

Fig. 6. RD performance for the luminance component of image *Barbara*.

frames of CIF-resolution sequence *Tempete*, and for the first ten frames of 720p50-resolution sequences *Mobcal*, *Parkrun* and *Shields*. For these sequences, all frames were coded in intra mode [16].

Figs. 5 and 6 present the RD performances for the proposed method. Clearly, it offers great gains for directional images.

Table II shows the PSNR gains and the bit-rate reduction for each image or sequence tested here, using the BJM metric [17].

TABLE II
PSNR GAIN AND RATE REDUCTION FOR DIFFERENT IMAGES AND SEQUENCES

IMAGE/SEQUENCE	PSNR GAIN (dB)	RATE REDUCTION (%)
Barbara	1.17	18.68
Bike	0.56	9.69
Foreman	0.08	1.44
Lena	0.1	2.27
Mobcal	0.01	0.26
Monarch	0.13	1.82
Mother and Daughter	0.06	1.21
Paris	0.15	1.68
Parkrun	0.04	0.59
Peppers	0.16	3.93
Shields	0.02	0.47
Spoke	2.74	51.92
Tempete	0.12	1.51

TABLE III
AVERAGE ENCODING/DECODING TIME IN SECONDS FOR DIFFERENT IMAGES AND SEQUENCES

IMAGE/SEQUENCE	PROPOSED	H.264/AVC
Barbara	4.56 / 0.24	2.06 / 0.07
Bike	97.92 / 2.97	47.88 / 0.79
Foreman	1908.5 / 18.23	815.85 / 7.39
Lena	4.28 / 0.16	1.79 / 0.07
Monarch	4.30 / 0.14	1.83 / 0.09
Mother and Daughter	1779.4 / 13.28	703.18 / 6.77
Spoke	4.55 / 0.25	2.11 / 0.07

For the test images, the rates were generally inside the range of 0.2 and 1.4 bits per pixel. For the test sequences, they had a wide range, from 400 to 7000 kbits/s for CIF-resolution sequences, and from 10 to 100 Mbits/s for 720p50-resolution sequences. From these results, it can be seen that the proposed intra-prediction mode offers substantial gains for directional images and sequences, and it yields a slightly better performance than regular H.264/AVC for the other images and sequences.

Table III shows the average encoding and decoding times for some of the tested images and sequences. There is roughly a 150% increase in encoding and decoding times from adding the proposed intra-prediction mode to H.264/AVC. Note that the code was not optimized for performance.

Tables IV and V show, for images *Spoke* and *Lena*, respectively, the average percentage of chosen intra modes using the original H.264/AVC "horizontal-up" prediction mode (mode 8), and substituting this mode by the proposed method. The top three modes for each situation are marked in bold letters. It can be seen that the proposed mode is always ranking among the top three modes. In the case of the *Spoke* image, the proposed method is fundamental for the high PSNR gain and rate reduction, and in the case of the *Lena* image, it does not incur in great gains, despite being selected in a great number of times. Other tests show that these two images are representative in terms of mode choices.

Comparing these results with previous works, we find mixed gains and losses. For instance, Graziosi *et al.* [5] report PSNR gains similar to ours for the *Barbara* and *Lena* images, using the

TABLE IV
AVERAGE PERCENTAGE OF CHOSEN INTRA MODES FOR THE *Spoke* IMAGE

MODE	4x4 ORIG.	4x4 PROP.	8x8 ORIG.	8x8 PROP.
0	**25.33**	2.22	**21.24**	1.80
1	5.37	2.20	3.92	1.74
2	**27.58**	**31.59**	**25.16**	**27.01**
3	4.49	3.97	3.80	3.01
4	**16.27**	**10.32**	**21.35**	**10.90**
5	9.89	4.73	12.49	4.21
6	4.39	3.20	5.11	2.32
7	4.24	7.17	4.28	4.13
8	2.42	**34.67**	2.64	**44.88**

TABLE V
AVERAGE PERCENTAGE OF CHOSEN INTRA MODES FOR THE *Lena* IMAGE

MODE	4x4 ORIG.	4x4 PROP.	8x8 ORIG.	8x8 PROP.
0	**31.55**	**17.19**	**30.60**	**17.08**
1	5.84	4.53	4.68	4.01
2	**22.08**	**22.05**	**18.61**	**18.91**
3	**10.78**	9.83	10.39	9.76
4	5.30	4.64	5.49	4.60
5	6.37	5.39	7.44	5.81
6	3.29	3.98	3.89	3.88
7	8.57	10.57	**11.29**	10.92
8	6.24	**21.83**	7.60	**25.04**

same type of intra-prediction methods in a much more complex image coding algorithm.

Zheng *et al.* [11] enhance the template matching prediction method by adding an adaptive illumination compensation. They present bit-rate savings of 12.80, 4.53 and 3.42% for the CIF-resolution sequences *Foreman*, *Paris* and *Tempete*, respectively, and 18.32% for a non-standard image that presents non-uniform illumination, denominated *Toy example*. Our proposed method achieves 1.44, 1.68 and 1.51% bit-rate savings for the same sequences, and 3.11% for the *Toy example*. Directional images such as *Barbara*, *Bike* and *Spoke* were not tested by Zheng *et al.*, which represent the highest gains for our method. Furthermore, the *Toy example* is highly directional, but suffers from great differences in illumination, which indicates that our proposed method could also benefit from the adaptive illumination compensation presented by Zheng *et al.*.

Liu *et al.* [15] present an edge-oriented intra-prediction method for H.264/AVC and JPEG2000, reporting maximum gains of 1.478 and 0.341 dB for the *Foreman* sequence and the *Barbara* image. Our method presents corresponding PSNR gains of 0.08 and 1.17 dB. Nevertheless, Liu *et al.* achieve these gains by using only 16×16 blocks, as opposed to the 4×4 and 8×8 blocks used here, which makes the comparison harder to judge.

IV. CONCLUSIONS

In this letter, we proposed a new intra-prediction mode for the H.264/AVC standard, following an Nth order Markov linear predictor model. The weights are calculated by a least-squares optimization at the neighbouring blocks that contain reconstructed data, so that the decoder can perform the same calculation, without demanding transmission overhead. Results for the proposed method show that, for directional images such as *Barbara* and *Spoke*, the proposed method achieves substantial rate reductions (18.68 and 51.92%, respectively), while other test images and sequences present moderate, yet positive, gains.

REFERENCES

[1] T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE. Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[2] D. Marpe, T. Wiegand, and S. Gordon, "H.264/MPEG4-AVC fidelity range extensions: Tools, profiles, performance, and application areas," in *Proc. ICIP 2005*, Genoa, Italy, Sep. 2005, vol. 1.

[3] A. Al, B. Rao, S. Kudva, S. Babu, D. Sumam, and A. Rao, "Quality and complexity comparison of H.264 intra mode with JPEG2000 and JPEG," in *Proc. ICIP 2004*, Singapore, Oct. 2004.

[4] X. Li and M. Orchard, "Edge directed prediction for lossless compression of natural images," *IEEE. Trans. Image Process*, vol. 10, no. 6, pp. 813–817, Jun. 2001.

[5] D. Graziosi, N. Rodrigues, E. da Silva, S. de Faria, and M. de Carvalho, "Improving multiscale recurrent pattern image coding with least-squares prediction mode," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[6] L. Liu, Y. Liu, and E. Delp, "Enhanced intra prediction using context-adaptive linear prediction," in *Proc. PCS 2007—Picture Coding Symp.*, Nov. 2007.

[7] S. Lee and N. Cho, "Intra prediction method based on the linear relationship between the channels for YUV 4:2:0 intra coding," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[8] S. Takamura, M. Matsumura, and Y. Yashima, "A study on an evolutionary pixel predictor and its properties," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[9] G. Ulacha and R. Stasinski, "Context based lossless coder based on RLS predictor adaption scheme," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[10] X. Li, "Least-square prediction for backward adaptive video coding," *EURASIP J. Appl. Signal Process.*, vol. 2006, no. 1, pp. 126–126, 2006.

[11] Y. Zheng, P. Yin, O. Escoda, X. Li, and C. Gomila, "Intra prediction using template matching with adaptive illumination compensation," in *Proc. ICIP 2008*, San Diego, CA, Oct. 2008.

[12] Y. Dai, Q. Zhang, and C. Kuo, "Fast 2D intra prediction (2DIP) mode decision for image and video coding," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[13] Y. Piao and H. Park, "An adaptive divide-and-predict coding for intra-frame of H.264/AVC," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[14] J. Chen and W. Han, "Adaptive linear prediction for block-based lossy image coding," in *Proc. ICIP 2009*, Cairo, Egypt, Nov. 2009.

[15] D. Liu, X. Sun, F. Wu, and Y. Zhang, "Edge-oriented uniform intra prediction," *IEEE. Trans. Image Process*, vol. 17, no. 10, pp. 1827–1836, Oct. 2008.

[16] T. Tan, G. Sullivan, and T. Wedi, Recommended Simulation Common Conditions for Coding Efficiency Experiments. Marrakech, Morocco, ITU-T Q.6/SG16, VCEG-AE010, Jan. 2007.

[17] G. Bjøntegaard, Calculation of Average PSNR Differences Between RD-Curves VCEG Contribution VCEG-M33, Apr. 2001.