CONTEXT-ADAPTIVE PIXEL BASED PREDICTION FOR INTRA FRAME ENCODING

Yongbing Zhang[1], Li Zhang[2], Siwei Ma[2], Debin Zhao[1], Wen Gao[2]

[1]Harbin Institute of Technology, Harbin, China
[2]Peking University, Beijing, China
{ybzhang, zhanglili, dbzhao, swma, wgao}@jdl.ac.cn

ABSTRACT

Intra prediction is one effective method to remove the spatial redundancies in intra frame coding. Better intra prediction will result in the residual with less energy, which will decrease the number of bits needed to reconstruct the signal at decoder. To improve the accuracy of intra prediction, a context-adaptive pixel based prediction (CAPBP) algorithm is proposed in this paper. For each pixel within the target block to be encoded, the prediction is calculated as the linear weighted summation of the reconstructed pixels within the left column and the above row. Based on the assumption that pixels having the same coordinates within one block own the same prediction weights, we calculate the corresponding weights for each pixel within the target block by the least square method. The same processing is also performed at decoder; hence the prediction weights do not need to be sent to the decoder. Experimental results verify that the proposed algorithm is able to improve the efficiency of intra frame coding up to 0.5dB.

Index Terms— Intra prediction, pixel adaptive weighted prediction, least square method

1. INTRODUCTION

Intra prediction, which is used to eliminate the spatial redundancy originating from high correlation within adjacent regions within one frame, is one of the most important ingredients in intra frame coding. The sample predictor block is usually created by extrapolating the reconstructed pixels surrounding the target block to be coded. The original block subtracts the predictor to form the residual block, which is then coded using transformation, quantization and entropy coder.

Intra prediction can be dated back to [1], which is then followed by the proposals [2], [3] and [4]. All these techniques evolved into the current form of intra prediction in the H.264/AVC standard specification [5]. To better capture the local properties of video signals, H.264/AVC divides the block sizes for intra prediction into 4x4 (intra4x4), 8x8 (intra8x8) and 16x16 (intra16x16). For intra4x4 and intra8x8 modes, nine prediction modes (i.e., eight directional modes and one DC mode) are employed for luma samples. Additionally, four prediction modes (horizontal, vertical, DC and plane modes) are utilized for intra16x16 luma and for the chrominance components of all the intra prediction modes. The optimal mode is selected from all the candidate ones via rate distortion optimization criterion [6]. It should be noted that all these intra prediction modes have fixed extrapolation weights, which neglect the varying properties of neighboring contexts. Actually, video content may be spatial variant within one block. As a consequence, it would be desirable to devise a prediction mode capturing the varying properties of local contexts.

To improve the accuracy of intra prediction, a context-adaptive pixel based prediction (CAPBP) is proposed in this paper. In the proposed algorithm, each pixel within the target block is predicted as the weighted summation of reconstructed pixels within the left column and the above row relative to the target block. The prediction weights of each pixel are varying accompanying with the coordinate change of the predicted pixel within the target block. Utilizing the reconstructed pixels adjacent to the target block, we estimate the prediction weights of pixels located at each position within the target block. To avoid the overhead of transmitting the prediction weights of each pixel, encoder and decoder perform the same procedure to estimate the prediction weights.

We replace the DC mode of intra 4x4 block and intra 8x8 block by the proposed CAPBP method on the platform KTA 2.4 [8]. Various experimental results verify that the proposed algorithm is able to improve the coding efficiency of intra frame coding up to 0.5dB.

The organization of this paper is as follows. Section 2 presents the background of intra prediction. Section 3 gives the detail description of the proposed CAPBP. Section 4 provides the experimental results. Finally, Section 5 concludes this paper.

2. BACKGROUND

In this section, we will give the brief description of the intra prediction modes for the 4x4 and 8x8 luminance blocks as specified in the H.264/AVC standard. It provides DC mode and 8 direction modes to predict the target block by extrapolating along the 8 corresponding directions, which is shown in Fig.1. The extrapolating is realized by weighted summing the reconstructed pixels surrounding the target block with fixed coefficients. These extrapolations achieve good results for the frames with simple textures. However, since the extrapolating coefficients are fixed, they can not adapt to the varying content of the target block, which is often observed in the real frame content exhibiting complex textures. Consequently, it is very desirable to devise an effective intra prediction method which is able to capture the varying local contexts of the target block.

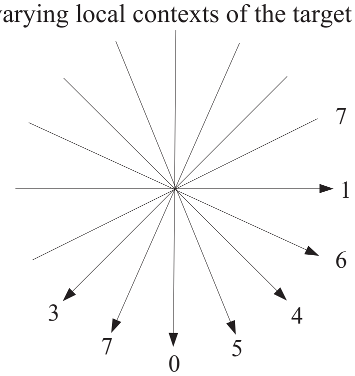

Fig.1 8 directional intra prediction modes

3. PROPOSED CONTEXT-ADAPTIVE PIXEL BASED INTRA PREDICTION

3.1. CAPBP Description

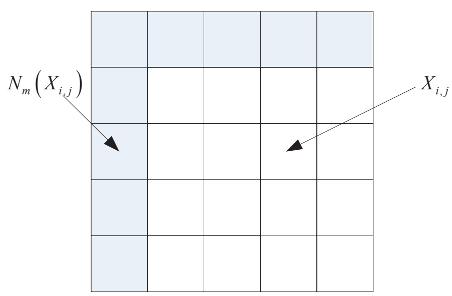

Fig.2 Context-adaptive pixel based prediction

The proposed CAPBP is depicted in Fig.2, where the blank square represents the pixel to be coded in the target block and the light gray square represents the reconstructed pixels. In the proposed CAPBP, each pixel within the target block is predicted as the linear weighted summation of the reconstructed pixels in the left column and the above row relative to the target block. As shown in Fig.2, pixel located at (i, j) within the target block is estimated as

$$\hat{X}_{i,j} = \sum_{m=1}^{M} w_{i,j}(m) \times N_m(X_{i,j}), \tag{1}$$

where $\hat{X}_{i,j}$ is the estimated intensity value with $0 \le (i, j) \le 3$ for 4x4 block and $0 \le (i, j) \le 7$ for 8x8 block, M represents the number of reconstructed pixels within the left column and the above row of the target block, $N_m(X_{i,j})$ represents the *m*th neighboring reconstructed intensity value of $X_{i,j}$ and $w_{i,j}(m)$ represents the estimating weight of the *m*th neighboring reconstructed pixel.

Different from the traditional intra prediction modes with fixed extrapolation weights, the CAPBP provides the adaptive prediction weights, which can be adjusted to the local context. Besides, it is noted that each pixel within the target block has unique prediction weights. This enables the CAPBP to better capture the spatial varying properties of the target block, which can further improve the prediction accuracy.

Since each predicted pixel has its own extrapolation weights, the overhead would be very high if each set of extrapolation weights are transmitted to the decoder. To reduce such high overhead of transmitting extrapolating weights, we will present an efficient weighting coefficients derivation in the next subsection.

3.2. Weighting Coefficients Derivation

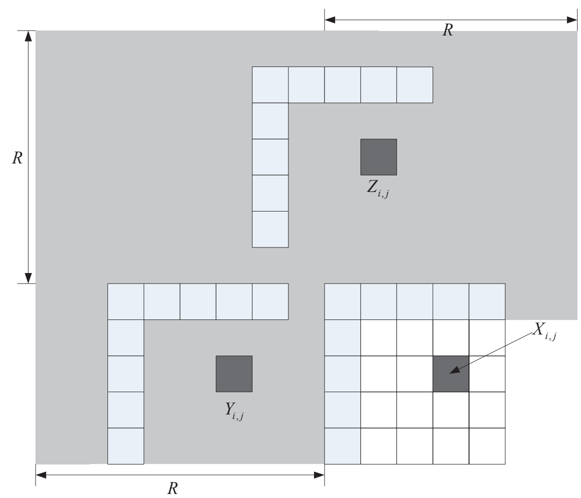

Fig.3 The mapping between target sample and training sample

To reduce the overhead of transmitting extrapolating coefficients, we devise an algorithm to derive the optimal coefficients. We first select some training samples from the reconstructed pixels adjacent to the target block. As shown in Fig.3, the training samples are represented by the heavy gray regions, which have at most *R* pixels to the left, above and right of the up-left corner of the target block. *R* is defined to be radius of the training sampling region in this paper. For each pixel to be predicted $X_{i,j}$, we find all the valid training samples in the training region. If we say one training sample valid, the following two requirements must be satisfied.

1. The training sample must have been reconstructed.
2. The block, in which the training sample has the same coordinate relative to the block with the pixel to be predicted, is located. All the pixels within the

left column and the above row of the located block must have reconstructed.

Two examples are shown in Fig. 3 to further interpret what valid means here. Take $Y_{i,j}$ for example, if we want to predict pixel $X_{i,j}$, which locates at the second row and the third column of the current 4x4 block, we first test whether pixel $Y_{i,j}$ has already been reconstructed. If the answer is positive, we locate the block, in which $Y_{i,j}$ is also located at the second row and the third column. Then we figure out the pixels, as shown the light gray squares surrounding $Y_{i,j}$, within left column and the above row of the located block. We test whether all the pixels figured out have already been reconstructed. If the answer is also positive, $Y_{i,j}$ is defined to be valid. The same testing process is also applied to $Z_{i,j}$ and other training samples.

For all the valid training samples (take $Y_{i,j}$ for example here), we approximate them as the linear weighted summation of the figured out pixels as

$$\hat{Y}_{i,j} = \sum_{m=1}^{M} w_{i,j}(m) \times N_m(Y_{i,j}), \tag{2}$$

where $N_m(Y_{i,j})$ represents the mth neighboring reconstructed pixel values of $Y_{i,j}$. The distortion between the approximated and the actual values of the valid training sample $Y_{i,j}$ can be computed as

$$D(Y_{i,j}) = \left(Y_{i,j} - \hat{Y}_{i,j}\right)^2, \tag{3}$$

where $\hat{Y}_{i,j}$ represents the approximated values obtained by Eq. (2). The optimal extrapolation weights correspond to the one which minimizes the distortion between all the actual valid training samples and the approximated training samples, which can be expressed as

$$\bar{W}^* = \min_{W} \sum_{s_i \in S_{valid}} D(s_i), \tag{4}$$

where S_{valid} represents the set of all the valid training samples and s_i represents the ith sample within S_{valid}. According to the least-squares method, the optimal extrapolation weight vector $\bar{W}^*$ can be calculated as

$$\bar{W}^* = \left(C^T C\right)^{-1}\left(C^T S\right), \tag{5}$$

where S is a column vector with length of U, C is a $U \times V$ matrix, with U representing the number of valid training samples and V representing the number of figured out neighboring pixels for each training sample.

Since the weighting coefficients derivation process is performed on the already reconstructed pixels, the same process can also be performed to the decoder if the target block selects the CAPBP as the best mode. As the consequence, the proposed weighting coefficients derivation algorithm can avoid the mismatch between the encoder and decoder if we omit the transmission of extrapolation weights to decoder.

4. SIMULATION RESULTS

To verify the performance of the proposed intra prediction method, the proposed CAPBP was implemented in the latest KTA 2.4 reference software. Since both DC mode and the proposed CAPBP use the weighted summation of surrounding reconstructed pixels as the prediction of the target block, we replace the DC mode of 4x4 block and 8x8 block. The applied test conditions were as follows,

1) Three resolution (QCIF, CIF and WQVGA) sequences, each of which has three test sequences.
2) 50 frames
3) All intra
4) CABAC
5) Intra mode-dependent directional transform (MDDT) is enabled
6) 8x8 transformation is on
7) *R*=10 for 4x4 block and *R*= 20 for 8x8 block

RD curves of some test sequences are plotted in Figures 4-6. Here “anchor” represents the result achieved by the default intra prediction method in KTA 2.4 software. It is to observe that the proposed CAPBP outperforms the anchor from low to high bitrates for all the three test sequences. Another observation is that the performance improvement is relatively low at the low bitrate, however it gets improved with the increase of bitrates.

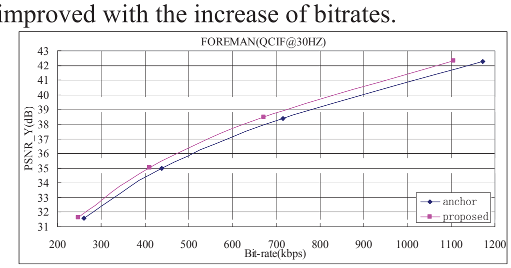

Fig.4 RD curves of *Foreman* (QCIF@30HZ)

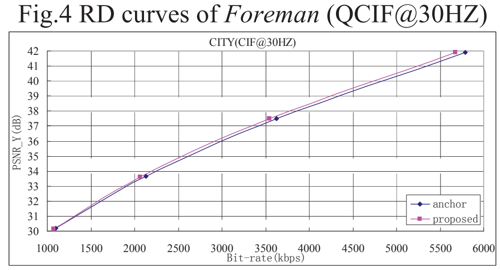

Fig.5 RD curves of *City* (CIF@30HZ)

Figures 7 and 8 give the mode ratios of 4x4 and 8x8 blocks within the first frame of Foreman (QCIF@30HZ). It can be seen that the ratio of DC mode in the proposed CAPBP is higher than that in the anchor. This is because the higher accuracy prediction brought by CAPBP forces more DC modes are selected by the rate distortion optimization criterion.

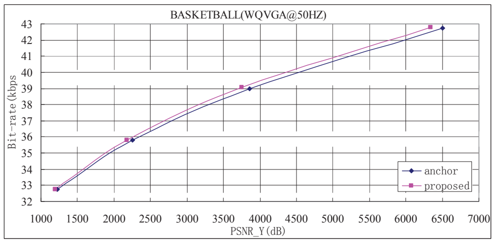

Fig.6 RD curves of *Basketball* (WQVGA@50HZ)

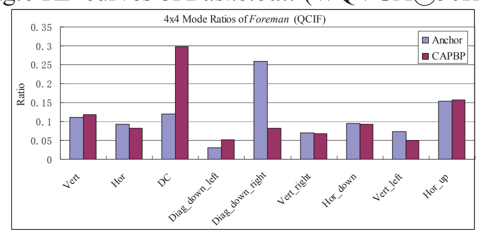

Fig.7 Mode ratio of intra 4x4 block within the first frame of *Foreman* (QCIF@30HZ)

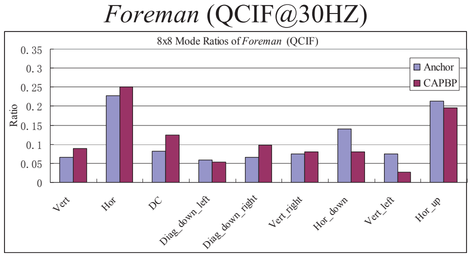

Fig.8 Mode ratio of intra 8x8 block within the first frame of *Foreman* (QCIF@30HZ)

Table 1 Comparison between anchor and the proposed CAPBP

Resolution	Sequence	$\Delta PSNR-Y$ (dB)	ΔR (%)
QCIF	*Foreman*	0.505	-6.85
	Carphone	0.171	-2.18
	Hall	0.155	-1.83
	Avg	**0.277**	**-3.62**
CIF	*City*	0.153	-2.15
	Foreman	0.221	-3.98
	Highway	0.176	-5.06
	Avg	**0.183**	**-3.73**
WQVGA	*Basketball*	0.195	-3.21
	Flowervase	0.172	-2.44
	Keiba	0.084	-1.32
	Avg	**0.150**	**-2.32**

The average luma PSNR gains and the average bitrate reduction percentage calculated as in [7] compared with KTA 2.4 reference software are shown in Table 1. It can be seen that the average PSNR gains of QCIF, CIF and WQVGA sequences are 0.277dB, 0.183dB and 0.15dB, respectively. Meanwhile, the average bitrate reductions of these three resolution sequences are 3.62%, 3.73% and 2.32%, respectively. Especially, for *Foreman* (QCIF), the average PSNR gain can be up to 0.505dB and the corresponding bitrate reduction is up to 6.85%. The performance improvement brought by the proposed CAPBP is mainly attributed to its ability to tune the extrapolation weights according to the local contexts of the target block.

5. CONCLUSIONS

In this paper, we proposed a novel CAPBP algorithm to improve the intra prediction accuracy. Each pixel within the target block is predicted as the linear weighted summation of the reconstructed pixels surrounding the target block. To avoid the heavy overhead of transmitting extrapolation weights to decoder, we compute the weights at both encoder and decoder by exploiting the similarity contexts between the target block and the neighboring reconstructed regions. Experimental results demonstrate that the proposed CAPBP is able to improve the intra prediction accuracy.

ACKNOWLEDGEMENT

This work was supported in part by National Science Foundation of China (60803068), and National Basic Research Program of China (973 Program, 2009CB320904).

6. REFERENCES

[1] Gisle Bjontegaard, "Coding improvement by using 4x4 blocks for motion vectors and transform", *ITU-T / Study Group 16 / Video Coding Experts Group (Question 15)*, Eibsee, Bavaria, Germany, document Q15-C-23, 2-5 December, 1997.

[2] Gisle Bjontegaard, "Response to Call for Proposals for H.26L", *ITU-T / Study Group 16 / Video Coding Experts Group (Question 15)*, Seoul, Korea, document Q15-F-11, 3-6 November, 1998.

[3] Nokia Research Center, "MVC Video Codec - Proposal for H.26L", *ITU-T / Study Group 16 / Video Coding Experts Group (Question 15)*, Seoul, Korea, document Q15-F-24, 3-6 November, 1998.

[4] Greg Conklin, "More Results on New Intra Prediction Modes", *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6)*, Geneva, CH, document JVT-B080, Jan. 29-Feb. 1, 2002.

[5] ISO/IEC JTC1 and ITU-T, "H.264/AVC - Advanced Video Coding for Generic Audiovisual Services", *ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 part 10)*, version 3, 2005.

[6] T. Wiegand et al, "Overview of the H.264/AVC Video Coding Standard", IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, Jul. 2003.

[7] Pateux et al., "An excel add-in for computing Bjontegaard metric and its evolution", VCEG-AE07, 2007.

[8] "JM11.0KTA2.4r1," http://iphome.hhi.de/suehring/tml/down load/ KTA," July. 2009