Reduced 4x4 Block Intra Prediction Modes using Directional Similarity in H.264/AVC

MYOUNG-SEO KIM[1], CHEONG-GHIL KIM[2], GI-HOON HONG[2], and SHIN-DUG KIM[2]
[1]System LSI Division, Semiconductor Business
Samsung Electronics CO., LTD.
Gyeonggi-Do, KOREA

[2]Department of Computer Science
Yonsei University
134 Shinchon-Dong, Seodaemun-Ku, Seoul 120-749
KOREA

Abstract: - This paper presents a low complexity 4x4 block intra prediction algorithm for H.264/AVC by reducing the original 9 prediction modes to 3 sub-sampled ones. The proposed algorithm takes advantage of the directional similarity found in the original ones to predict the most probable mode. The simulation result shows that the proposed algorithm can the computational complexity around 50~70% while maintaining similar SNR of Y, U, and V as compared with the full search algorithm.

Key-Words: - Video coding, MPEG-4, H.264/AVC, Intra prediction, Full search.

1 Introduction

H.264/AVC [1] is the newest video coding standard jointly developed by the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group (MPEG). It provides improved coding efficiency over existing video coding standards such as H.263+ and MPEG-4 Part-2. This improvement comes from not a single feature but a number of enhanced coding techniques including 4x4 integer DCT (Discrete Cosine Transform), intra-prediction in I-frame, multiple reference frames and multiple block sizes for P-frame, quarter-pixel motion compensation, and so on [2, 3]. However, these coding gains inevitably result in the increase of coding complexity greatly.

Intra prediction algorithm generates a prediction for a Macroblock (MB) based on spatial redundancy. That is, the intra prediction of H.264/AVC is defined in the pixel domain by referring to neighboring samples of the previously coded blocks. For the Luma samples, a prediction block is formed either for each 4x4 block or for a 16x16 macroblock. There are 9 optimal prediction modes for each 4x4 Luma block, four modes for a 16x16 Luma block, and another four modes for the chroma components. The encoder typically selects a prediction mode for each block, which minimizes the difference between a prediction block and the original block to be encoded.

Originally, intra prediction is conducted using the full search of all possible modes. Even though the full search can have an optimal mode selection, it requires heavy computational expenses. Recently, several algorithms have been proposed to reduce the heavy computational complexity of intra prediction in H.264/AVC. Taking advantage of the feature that that intra prediction modes are established based on direction, a Sobel edge operator is devised to reduce the intra prediction modes in [4]. Another paper [5] efficiently explores the neighborhood direction around the minimum one and thus skips other unlikely ones. In [6], a fast mode decision algorithm is introduced utilizing integer transform which can find the directions of the original image and adaptive threshold for early termination.

In this paper, we propose reduced 4x4 block intra-prediction modes using directional similarity of existing modes in order for low complexity of intra prediction. As a result, the original 9 modes can be reduced to 3 modes to predict the most probable mode. Our experimental results show that the proposed algorithm can save around 50~70% of 4x4 intra-prediction computational complexity while keeping the output quality nearly the same as taking the full search (FS) algorithm.

The rest of this paper is organized as follows. In Section 2, the H.264/AVC intra prediction algorithm is briefly reviewed. Our proposed algorithm is illustrated in Section 3. Experimental results are shown in Section 4 and conclusions are presented in Section 5.

2 4x4 Intra Prediction

In H.264/AVC, intra prediction is performed on 4x4 and 16x16 blocks and takes around 7 to 10 percent of total computational resource [7]. For 4x4 bock intra prediction, a 16x16 Luma block is divided into sixteen 4x4 blocks and each 4x4 block is predicted separately. In the 16x16 block intra mode, the whole 16x16 Luma macroblock is predicted. Fig. 1 (a) shows a 4x4 block with its predictor pixels labeled from A to M. The sixteen pixles in one 4x4 block labeled from *a* to *p* are the pixels to be predicted. Nine prediction modes are supported by the 4x4 intra mode. Depending on the prediction mode, some pixels are chosen and used as the predictors. Fig. 1 (b) shows the directions of eight prediction modes. The number given with each arrow represents the mode number. The pixels *A* to *M* belong to the neighboring blocks and are assumed to be already encoded and reconstructed and are therefore available in the encoder and decoder to generate a prediction for the current MB. Each 4x4 Luma prediction mode generates 16 predicted pixel values using some or all of the neighboring pixels *A* to *M* as shown in Fig. 2. The arrows indicate the direction of prediction in each mode. The predicted pixels are calculated by a weighted average of the neighboring pixels *A*-*M* for each mode except Vertical, Horizontal, and DC modes.

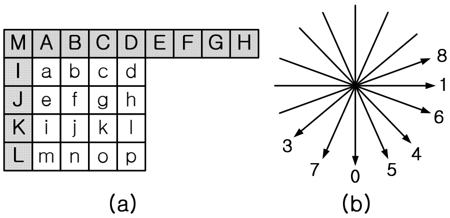

Fig. 1. Overview of 4x4 intra prediction.

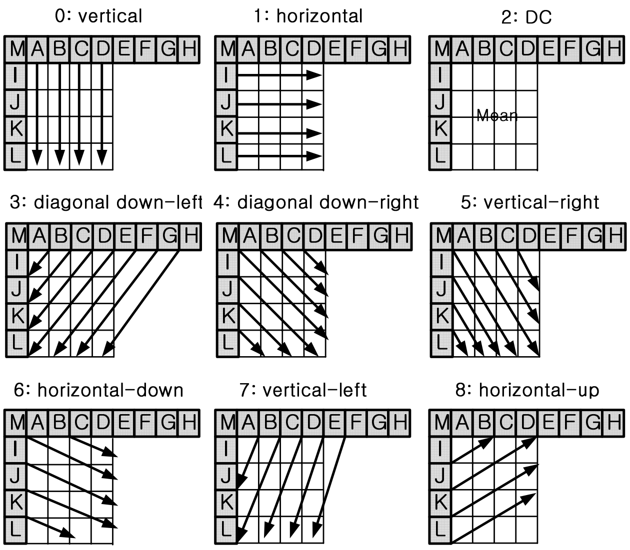

Fig. 2. 4x4 Luma prediction modes.

In the reference software called JM8.0 [8] from JVT (Joint Video Team), a full search is used to examine all the 9 modes to find the optimal one. The computational flow is as follows. First, a 4x4 predicted block is generated according to a mode. Second, calculate the sum of absolute difference (SAD16) between the original 4x4 block and the predicted block. Third, compute the bellowing Eq. (1).

$$Cost_{4x4} = SAD16 + 4P\lambda(Qp), \quad (1)$$

where $\lambda(Qp)$ is an approximate exponential function of the quantization factor *Qp* and *P* equals zero for the most probable mode and one for the other modes. Finally, repeat the first through the third steps for all the 9 modes, and choose the one that has the minimum cost. Although the full search algorithm can achieve an optimal prediction mode selection, it is computationally expensive. Consequently, it may be a serious bottleneck point in I-frame coding. Therefore, it is highly desirable to develop a fast algorithm for 4x4 intra-prediction mode selection.

3 Proposed 4x4 Intra Prediction

3.1 Motivation

In general, a 4x4 block would have rather smooth texture because of its small block size. Therefore, it is possible to represent a 4x4 block by its sub-sampled pixels roughly. By taking this characteristic, we can obtain a good prediction scheme by checking selected pixels within a 4x4 block instead of checking all 16 ones.

When the intra-prediction modes of the neighboring top and left blocks are known, we can surmise the most probable mode for the current block without any additional computation. This fact offers a very good opportunity to reduce the overall complexity. One more useful thing to help us reduce the complexity is that the 9 modes are based on the directional characteristics. So if a particular mode gives minimum cost after performing a partial search by using only some groups of pixels, the optimal prediction mode may have a high possibility to be either this mode or one of its two neighbors. Taking advantage of this operational characteristic, we can reduce the computational complexity by performing three probable searching modes. Since the proposed algorithm does not need to examine all the 3 groups of pixels for each mode, the predicted pixels of a specific mode should be generated only when they are required to be checked.

For this purpose, we have investigated the characteristics of intra coding through several experiments. We obtain four interesting characteristics through our experiments. First, among the SAEs (sum of absolute errors) of all 9 modes, some modes have small SAEs comparing with other modes. Hence, we can understand that the best match to the actual current block is given by a specific mode because this mode gives the smallest SAE. Second, in the 4x4 intra prediction, the optimal mode usually has similar direction to the other good modes. Third, the directional features of the 4x4 blocks can be preserved roughly even though down-sampling is performed. Fourth, none of the modes has good prediction in a chaotic area. Based on the above four conditions, we apply several profiling information and threshold values on the available information to determine whether it is necessary to search the optimal mode. As a result, we propose a new intra prediction algorithm for 4x4 blocks to predict the most probable mode with the help of early termination and optimal threshold setting by profiling. Our experimental results show that the proposed algorithm can save around 50~70% of 4x4 intra-prediction computational complexity while keeping the output quality nearly the same as using the full search algorithm.

3.2 Reduced Intra Prediction Modes

Fig. 3 depicts the proposed reduced intra prediction modes that can classify the 24 pixels into 3 groups according to the directional similarity of specific prediction modes and the Luma similarity of pixels. Sub-sampled mode 1 selects pixels of *a*, *d*, *m*, *p*, *f*, *g*, *j*, and *k* for the original modes 0, 1, and 2. Sub-sampled mode 2 does those of *a*, *c*, *i*, *k*, *f*, *h*, *n*, and *p* for the original modes 4, 5, and 6. Sub-sampled mode 3 does those of *b*, *d*, *j*, *l*, *e*, *g*, *m*, and *o* for the original mode 3, 7, and 8. Each mode is a “down-sampled” version of the original block, where the features of an original block are reflected. If prediction mode can be determined by checking only some of the groups, computational reduction can be achieved greatly. The cost required for one group is represented as:

$$Cost_{one\ group} = SAD_{one\ group} + 4/3P\lambda(Qp), \qquad (2)$$

where $SAD_{one\ group}$ is the sum of absolute difference for the 8 pixels in the group, $\lambda(Qp)$ and P are the same as in Eq. (1). The sum of $Cost_{one\ group}$ for the 3 groups can be obtained from Eq. (2). Note that the $P\lambda(Qp)$ needs to be computed only when the comparison involves the most probable candidate mode. In fact, it does not need to be computed when comparing two modes that are not most probable, because the $P\lambda(Qp)$ is the same for them.

The computational steps of the proposed algorithm can be summarized as following. First, with given first block, start the full search of the 9 modes and initialize *T*. Here, we can obtain a threshold *T* that represents the average of SAEs obtained by 8 modes except for the highest SAE. This step can cause a significant impact for the accuracy of prediction depending on any image pattern for the next steps. Second, choose the 6 modes with minimum SAE. Through the profiling information of our experiments, we know that only 6 modes except those modes with high SAE among 9 modes are necessary. Third, if the block is at the top or left of the frame, check two modes using 16 pixels, namely, DC and horizontal mode for the top block, and DC and vertical mode for the left block. Choose one mode with minimum cost, and stop. This step does not need to perform the first and second step because of a particular case of block arrangement. Forth, if the block is not at the top or left of the frame, check the other modes. Note that the 9 modes are based on directional features. Therefore, we can obtain an optimal prediction mode after partial search of some modes. As a result, the optimal prediction mode for the current block may most likely be either this mode or one of its two neighbors. So we can reduce the number of candidate modes to search. Finally, if the threshold *T* is larger than the SAE of any determined mode of previous step, we take the most probable mode as the prediction mode; otherwise, compute the cost of group pixels in specific 6 modes gained by the profiling except for the determined mode of previous steps. Those specific 5 modes will be either contained in the minimum two groups or contained in the maximum three groups. So we will have a computational complexity of either 16 or 24 pixels respectively.

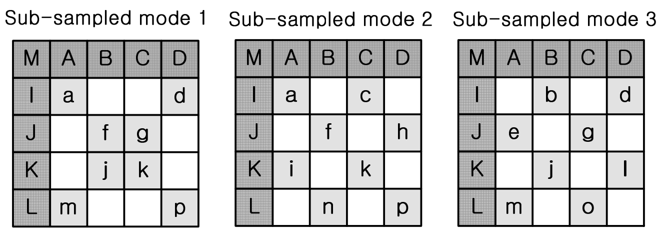

Fig. 3. Reduced modes.

Although the proposed algorithm takes a full search overhead of the initial 4x4 block, it may provide a significant clue which can reduce the next 4x4 intra prediction computation according to its image pattern. In terms of computational complexity, the full search technique checks roughly 16 x 9 = 144 pixels to determine the prediction mode for a 4x4 block. In the best case, the proposed technique ends in the third step without checking the first and second step because the block is at the top or left of the frame. The next best case is that only 24 pixels are checked. Although the worst case may happen, the proposed algorithm may check 3 x 8 + 3 x 8 = 48 pixels to determine the prediction mode for a 4x4 block. As we can see, the worst case degrades the computational efficiency greatly. In addition, we obtain the result that the prediction accuracy using the proposed algorithm is similar to that of the full search algorithm. The SNR (signal to noise ratio) of Y is used as a metric to verify the prediction accuracy. Details are provided in the next section.

4 Simulation Results

The computational complexity of H.264/AVC is examined as shown in Table 1. The intra prediction module in the encoder occupies about 7% of computation resource. So the fast intra-prediction method can reduce the complexity of encoder. The simulation is conducted with the reference JVT software JM8.0 [8]. All sequences are encoded to 250 frames. The frame rate is 30 with various *QP*s. The sequences are all in the common *YUV* 4:2:0 format, widely used in the video research community.

Table 1: Computational complexity of H.264/AVC

Encoder		Decoder	
Module	Complexity Proportion(%)	Module	Complexity Proportion(%)
ME	80.26	MC	43.15
4x4 DCT	6.12	IDCT	21.94
Intra Pred.	6.73	–	–
Loop Filter	0.86	Loop Filter	15.2
Other(Mem, IO)	6.03	Other(Mem, IO)	19.71

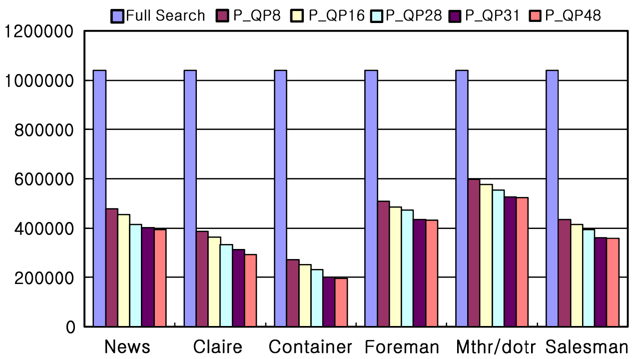

Fig. 4. Computational complexity of FS and the proposed algorithm with variable QP, QCIF, 250 frames.

The proposed algorithm and the FS are simulated on the six QCIF (Quarter Common Intermediate Format) sequences, Claire, Container, Foreman, Mother and Daughter, News, and Salesman as publicly available. In addition, the five CIF (Common Intermediate Format) sequences, namely bridge, highway, mobile, paris, and tempete are simulated for the improvement of objectivity. The most important feature is how much computational complexity can be reduced and whether the predicted image quality can be preserved respectively.

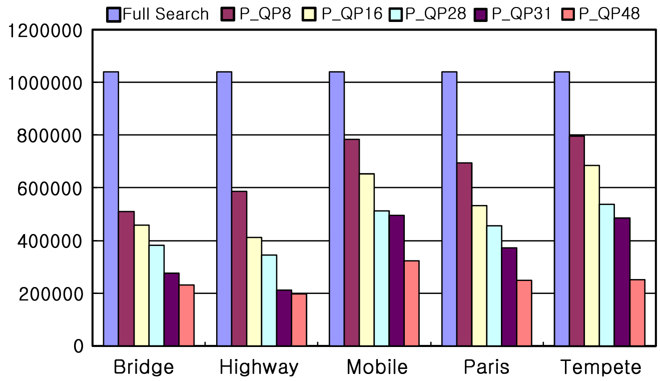

Fig. 5. Computational complexity of FS and the proposed algorithm with variable QP, CIF, 250 frames.

Table 2: Comparison results with variable QP, QCIF, 250 frames.

QP	Benchmark	SNR Y (dB)		Bit rate (Kbit/s)		Total bits	
		FS	Proposed	FS	Proposed	FS	Proposed
8	News	38.22	38.20	73.47	74.87	729792	733441
	Claire	40.39	40.38	29.32	29.56	480816	483220
	Container	37.20	37.18	41.37	41.58	410952	413007
	Foreman	35.67	35.65	112.73	123.82	1495584	1503062
	Mthr_dotr	36.90	36.89	46.34	46.54	770832	774686
	Salesman	38.47	38.46	45.05	45.32	672728	676092
28	News	36.80	36.78	65.13	65.62	646968	648262
	Claire	39.93	39.90	26.25	26.53	430520	431381
	Container	36.13	36.12	32.35	32.62	321328	321971
	Foreman	36.72	36.70	105.65	106.82	1401624	1404427
	Mthr_dotr	36.47	36.46	40.85	41.12	679504	680863
	Salesman	35.73	35.71	38.26	38.43	571320	572463
48	News	36.43	36.42	64.86	64.90	644232	645520
	Claire	39.84	39.81	26.12	26.16	428288	429145
	Container	35.80	35.77	32.87	32.89	326496	327149
	Foreman	35.41	35.36	105.24	105.34	1396160	1398952
	Mthr_dotr	36.42	36.38	40.96	41.01	681320	682683
	Salesman	35.19	35.14	38.22	38.26	570800	571942

The SNR of Y, U, and V, and the computational complexity results are compared with those of the full search technique. Also, the SNR of Y is used to evaluate the quality of a reconstructed image sequence quantitatively. The SNR of U and V is not used to evaluate because the 4x4 intra-prediction is predicted for luminance component except chrominance component. The results as shown in Tables 2 and 3 show that the proposed algorithm maintains approximately similar to the FS in terms of the output quality. The computational complexity analysis is shown in Figs. 4 and 5 with various *QP* factors. We obtain around 50~80% reduction of the computational complexity using the QCIF video formats, News, Claire, and Container; however, we obtain around 30~60% reduction of the computational complexity with the QCIF video formats, Foreman, Mthr/dotr, and Salesman. As for video formats, we obtain around 50~80% reduction of the computational complexity with Bridge and Highway and 20~70% reduction with Mobile, Paris, and Tempete. Generally, the more the value of quantization parameter is increased, the poorer the predicted image quality is caused. But the efficiency of compression is improved because the value of quantization parameter is specified as large.

As shown in Tables 2 and 3, the proposed algorithm can achieve significantly complexity reduction comparing with full search technique, but the aspect of bit rate and total bits tends to be increased slightly because the proposed algorithm is performed by less accurate prediction compared to full search technique. Nevertheless, the degradations are still within an acceptable range because human visual perception is unable to distinguish the PSNR difference of less than 0.2dB.

Table 3: Comparison results with variable QP, CIF, 250 frames.

QP	Benchmark	SNR Y (dB)		Bit rate (Kbit/s)		Total bits	
		FS	Proposed	FS	Proposed	FS	Proposed
8	Bridge	37.28	37.27	192.09	192.49	3195168	3211144
	Highway	38.14	38.12	193.33	193.53	3215752	3231831
	Mobile	32.67	32.65	1617.70	1617.90	26907696	27042234
	Paris	34.89	34.89	692.34	692.57	11515904	11573484
	Tempete	33.78	33.77	1029.07	1029.37	17116800	17202384
28	Bridge	37.09	37.08	186.95	187.12	3109536	3121974
	Highway	37.96	37.94	187.19	187.43	3113664	3126119
	Mobile	32.63	32.61	1608.56	1608.86	26755672	26862695
	Paris	34.23	34.23	684.49	684.72	11385344	11430885
	Tempete	33.72	33.71	1021.82	1022.06	16996232	17064217
48	Bridge	37.06	37.04	187.63	187.86	3120888	3127130
	Highway	37.92	37.89	187.20	187.41	3113832	3120060
	Mobile	32.60	32.57	1603.71	1606.97	26675064	26728414
	Paris	34.07	34.03	684.88	685.14	11391920	11414704
	Tempete	33.68	33.65	1020.41	1021.41	16972752	17006698

5 Conclusion

In this paper we presented a low complexity 4x4 block intra prediction algorithm for H.264/AVC by reducing the original 9 prediction modes to 3 sub-sampled ones. For this purpose, we first analyze the available information for optimal mode selection from all the 9 modes, that is, we set up a threshold value and choose the 6 modes with minimum SAE through the profiling information. Then we apply directional similarity of a specific mode and threshold values to the available information to determine whether it is necessary to search the optimal mode. So we can reduce the number of candidate modes to search as three modes. As a result, our experimental results show that the proposed algorithm can save around 50~70% of 4x4 intra-prediction computational complexity while keeping the output quality nearly the same as using the full search technique.

Acknowledgement

This work was supported by Seoul Research and Business Development Program (10698).

References:

[1] ISO/IEC 14496–10:2003, “Coding of Audiovisual Objects—Part 10: Advanced Video Coding,” 2003, also ITU-T Recommendation H.264 “Advanced video coding for generic audiovisual services.”

[2] ISO/IEC 14496–2: “Information technology-coding of audiovisual objects-part 2: visual,” Geneva, 2000.

[3] P. Topiwala, G. Sullivan, et. Al., “Performance evaluation of H.26L, TML 8 vs H.263++ and MPEG-4,” Document VCEG-N18, ITU-T Video Coding Experts Group (VCEG) Meeting, Santa Barbara, CA, USA, 24-27 Sept. 2001.

[4] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. J. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Tans. Circuits Syst. Video Technol.*, Vol. 15, No. 7, July. 2005, pp. 813-822.

[5] Chao-Chung Cheng, Tian-Sheuan Chang, "Fast three step intra prediction algorithm for 4/spl times/4 blocks in H.264", IEEE ISCAS 2005, Vol. 2, May 2005, pp. 1509-1512.

[6] Rui Su, Guizhong Liu, Tongyu Zhang, “Fast Mode Decision Algorithm for Intra Predition in H.264/AVC,” IEEE Int’l Conf. Speech and Signal Processings, Vol 2, May 2006, pp. II-921-II-924.

[7] 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,” *Circuit and Systems Magazine, IEEE*, Vol. 4, Issue 1, 2004, pp. 7-28.

[8] Joint Video Team (JVT), reference software JM8.0, http://bs.hhi.de/~suehring/tml/download/old_jm/jm80.zip