Efficient Intra Mode Decision Via Statistical Learning

Chiuan Hwang and Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
g946319@oz.nthu.edu.tw, lai@cs.nthu.edu.tw

Abstract. Intra mode selection and motion estimation for spatial and temporal prediction play important roles for achieving high video compression ratio in the latest video coding standards, such as H.264/AVC. However, both components take most of the computational cost in the video encoding process. In this paper, we propose an efficient intra mode prediction algorithm based on using the mode conditional probability learned from a large amount of training video sequences with the ground truth modes of each block to be encoded and its neighboring block modes as well as its associated image content features. By applying the proposed intra-mode selection algorithm into the H.264 reference code, we show significant reduction of the computation time with negligible video quality degradation for H.264 video encoding.

Keywords: Image coding, image analysis, video coding, video compression, intra prediction, video codec.

1 Introduction

To improve the computational efficiency of the intra-mode decision in H.264 video coding [1], there have been many different algorithms proposed recently. The traditional method is to run all intra modes, as listed in Table 1, for Luma and Chroma blocks, but it takes too much encoding time. Later, Cheng and Chang [2] used the block content characteristics, such as the block RD-cost correlation between the mode and its neighboring modes, to reduce the search of nine Luma 4x4 modes to no more than six modes. In addition, the block gradient information in the local block was used to determine a rough edge direction in some previous works [3-5]. Based on this direction, they used the corresponding directional mode and its two neighboring modes plus the DC mode to determine the mode with the lowest cost. In addition, Pan et al. [4] proposed a different way to predict the direction in a block. They collected statistics by computing the gradient of every pixel in a block and then chose the most possible direction from the statistics as their primary prediction mode. Sim and Kim [6] presented an efficient mode decision algorithm based on the conditional probability of the best mode with respect to the best modes of the adjacent blocks. Furthermore, Huang et al. [7] developed a fast intra frame coding system by combining several improved components, including context-based decimation of unlikely candidates, subsampling of matching operations, bit-width truncation, and interleaved full-search/partial-search strategy.

H.H.S. Ip et al. (Eds.): PCM 2007, LNCS 4810, pp. 148–157, 2007.

In this paper, we propose a new and efficient intra mode decision algorithm based on the mode conditional probability with respect to its neighboring encoded modes and its image features. This mode conditional probability is learned from a collection of video sequences with the ground truth modes determined by the H.264 reference program. We demonstrate the superior performance of the proposed algorithm over previous mode decision methods through experiments.

2 Intra Prediction in H.264/AVC

The H.264/AVC standard [1] provides variable block size on motion estimation and more different directional modes for intra prediction than previous MPEG standards. For the Luma 16x16 component and Chroma 8x8 component, they support four modes, including three directional modes and one DC mode. These different modes are listed in Table.1.

Table 1. (a) Luma 16x16 prediction modes and (b) Chroma 8x8 prediction modes

(a)

Mode	Mode description
0	Vertical
1	Horizontal
2	DC
3	Plane

(b)

Mode	Mode description
0	DC
1	Vertical
2	Horizontal
3	Plane

The modes for Luma 16x16 and Chroma 8x8 are different only with their orders. For Luma 4x4 sub-blocks, there are nine modes, including eight directional modes and one DC mode. The Luma 4x4 modes are listed in Figure 1.

Mode	Mode description
0	Vertical
1	Horizontal
2	DC
3	Diagonal_Down_Left
4	Diagonal_Down_Right
5	Vertical_Right
6	Horizontal_Down
7	Vertical_Left
8	Horizontal_Up

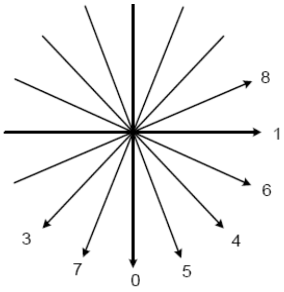


Fig. 1. The nine Luma 4x4 intra prediction modes and their corresponding directions

In H.264, to achieve better video quality and optimal bitrate compression, the RDO (Rate Distortion Optimization) is employed but with more computational cost. It compares the RD costs for different modes to determine which mode is optimal for the macroblock. The H.264 encoding process is illustrated in Figure.2.

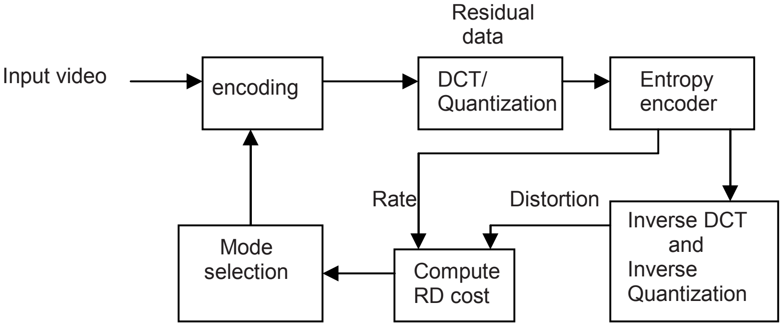


Fig. 2. H.264 encoding flowchart

The RD cost is computed with the following formula.

$$\begin{aligned} & J(s,c,MODE \mid QP,\lambda_{MODE}) \\ & = SSD(s,c,MODE \mid QP) + \lambda_{MODE} \cdot R(s,c,MODE \mid QP) \end{aligned} \tag{1}$$

Where s and c are the source video signal and the reconstructed video signal, respectively, QP is the quantization parameter, λ_{MODE} is the Lagrange multiplier, MODE indicates a macroblock mode, such as P16x16, P16x8, P8x16, P8x8, I16x16, I4x4, etc., SSD is the sum of the square differences between s and c, R(s,c,MODE|QP) is the number of bits associated with the chosen macroblock MODE and QP.

When it starts to encode a macroblock, as shown in the above flow chart, it needs to check all the mode combinations to choose the one with the minimal RD cost in the mode selection procedure. In I-frame, it only needs to do intra coding to decide the intra mode, and the total number of intra mode combinations is C8*(L4*16+L16), where C8, L4, and L16 denote the numbers of the Chroma8x8 modes, Luma4x4 modes and Luma16x16 modes, respectively. Thus, the total number of mode combinations is 4*(9*16+4)=592. This will take a considerable amount of time to compute the RD costs for all mode combinations, hence we need to reduce the total number of mode combinations in the mode decision process to speed up the computation.

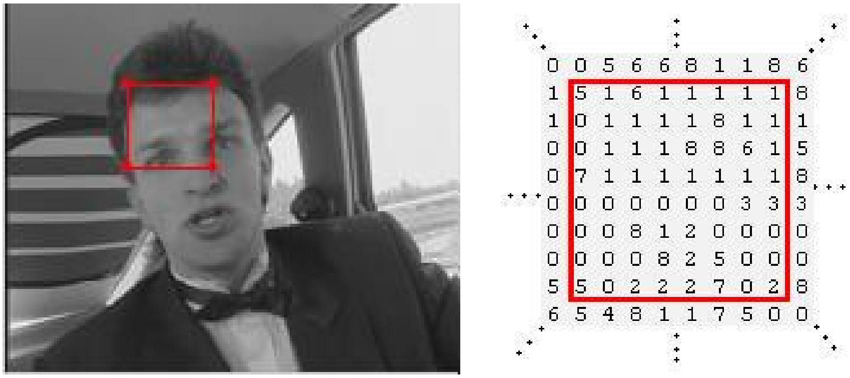
3 Neighboring Block Mode and Image Features

3.1 Neighboring Block Mode

Now, let us illustrate why the encoding block is related to its neighboring block modes in intra mode decision. The intra 4x4 mode for each macroblock is collected when JM 10.2 [8] encodes the sequence. Figure 3 and 4 depict examples of intra 4x4 mode maps from two video sequences. As we see from these two examples, the encoded block mode is closely related to its neighboring block modes and its own image content gradient features. So if we can utilize this characteristic to predict the mode of a block, we can estimate which mode may achieve lower bitrate and better video quality. This can also be applied to intra 16x16 mode and Chroma 8x8 mode.

8	8	8	8	7	7	7	7	2	2
8	8	8	7	7	7	8	7	2	2
8	8	8	7	7	8	8	7	2	2
3	3	3	8	7	7	7	8	2	2
3	3	8	8	7	7	7	7	2	8
2	2	2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8	8	8
1	1	8	8	8	8	8	8	8	8
8	1	6	8	8	8	8	8	8	8
8	8	8	8	8	8	8	8	8	8

Fig. 3. image and 4x4 intra mode map (the fiftieth frame of Forman CIF sequence)



0	0	5	6	6	8	1	1	8	6
1	5	1	6	1	1	1	1	1	8
1	0	1	1	1	1	8	1	1	1
0	0	1	1	1	8	8	6	1	5
0	7	1	1	1	1	1	1	1	8
0	0	0	0	0	0	0	3	3	3
0	0	0	8	1	2	0	0	0	0
0	0	0	0	8	2	5	0	0	0
5	5	0	2	2	2	7	0	2	8
6	5	4	8	1	1	7	5	0	0

Fig. 4. image and 4x4 intra mode map (the fiftieth frame of CarPhone QCIF sequence)

3.2 Image Gradient Feature

Feature based algorithms [2-5] have been commonly used for intra mode decision. It is critical to extract representative image features very fast. Several previous algorithms compute edge features to reduce modes in the search, but they usually lead to higher bitrates and larger distortion.

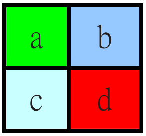
In this work, we extract very simple image features for intra mode decision. These image gradient features are the convolutions with the gradient mask shown in Figure 5. Wang et al. [5] also used similar gradient masks for extracting features for intra mode prediction.

$$\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \qquad\qquad \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}, \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

(a) (b)

Fig. 5. Gradient masks for image feature extraction, (a) two and (b) three masks

We apply the above masks for convolution with a 4x4, 8x8, or 16x16 block. Let us take an NxN block for example:



a,b,c,d are all the mean of N/2 * N/2 block

Fig. 6. 2x2 matrix for applying the mask

In our implementation, we quantize the computed gradient features for computing the conditional probability by using the following two different schemes:
Scheme (a):

1. Compute the correlation between the 2x2 matrix and the two masks in Figure 5(a) to obtain the two values S_v and S_h, respectively.
2. Normalize S_v and S_h to obtain the normalized coefficients Q_1 and Q_2 as follows:

$$\text{Sum}=\text{abs}(S_v) + \text{abs}(S_h) \tag{2}$$

$$Q_1=S_v/\text{Sum},\ Q_2=S_h/\text{Sum} \tag{3}$$

3. Quantize Q_1 and Q_2 to the levels inx_1 and inx_2, respectively.

Scheme (b):

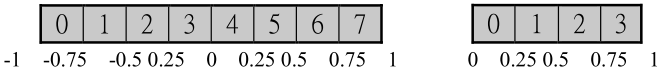
1. Compute the correlation between the 2x2 matrix and the three masks in Figure 5(b) to obtain the three values S_v, S_h and S_{diag}, respectively.
2. Normalize S_v, S_h and S_{diag} to obtain the normalized coefficients Q_1, Q_2, and $Q_{3,}$, respectively, as follows:.

$$\text{Sum}=(\text{abs}(S_v) + \text{abs}(S_h))+\text{abs}(S_{diag}) \tag{4}$$

$$Q_1=\text{abs}(S_v)/\text{Sum},\ Q_2=\text{abs}(S_h)/\text{Sum},\ Q_3=\text{abs}(S_{diag})/\text{Sum} \tag{5}$$

3. Quantize Q_1, Q_2 and Q_3 to inx_1, inx_2 and inx_3, respectively.

Note that we quantize the gradient features in scheme (a) and (b) into 8 and 4 levels, respectively. Figure 7 shows the uniform quantization used in our implementation.



(a) (b)

Fig. 7. Uniform quantization for (a) 8 and (b) 4 levels

4 Proposed Intra Mode Decision Algorithm

The proposed algorithm consists of the training and execution phases. We first describe the procedure for training in the following.

Step 1. Apply the exhaustive search to determine the optimal intra modes and extract the image gradient features for each macroblock in some training video sequences to obtain the training data.

Step 2. From the training data, we count the occurrence of joint event for the optimal modes for the current block and its neighboring blocks and the associated image features, Let the count of joint event be denoted by $C(m_c, \mathbf{m}_p, \mathbf{g})$, where m_c is the optimal intra mode for the current block, $\mathbf{m}_p$ is the vector containing the optimal modes in the left and upper blocks, and $\mathbf{g}$ is gradient feature vector computed from section 3.2.

Step 3. Compute and store the conditional mode probability distribution $p(m_c \mid \mathbf{m}_p, \mathbf{g})$ by normalizing the counts of occurrences of joint events as follows:

$$p(m_c | \mathbf{m}_p, \mathbf{g}) = \frac{C(m_c, \mathbf{m}_p, \mathbf{g})}{\sum_{m_c \in M} C(m_c, \mathbf{m}_p, \mathbf{g})} \quad (6)$$

where M is the set of all possible modes. For Luma 4x4, it contains mode 0 to 8. For Luma 16x16 and Chroma 8x8, the set M only contains mode 0 to 3.

After we have the above mode conditional probability learned from the training data set, we can use it to determine the most probable candidate modes from the encoded neighboring modes and the associated gradient features. Then, the search for the optimal mode is reduced to only these candidate modes, thus the computational cost for the intra mode decision is significantly reduced. To be more specific, we compute the gradient features and quantize them for each block as described in section 3.2. Then, the mode conditional probability distribution given these gradient features as well as the intra modes of the left and upper neighboring blocks is used to decide the most probable modes as the candidates in the refined search for the optimal intra modes from their RD cost values. In our implementation, we select four candidate modes in the refined search for the case of 4x4 blocks, and only two candidate modes are selected for 16x16 blocks and 8x8 blocks.

The flowchart of the proposed algorithm is depicted in Figure 8. We give the process flowcharts the training and encoding process.

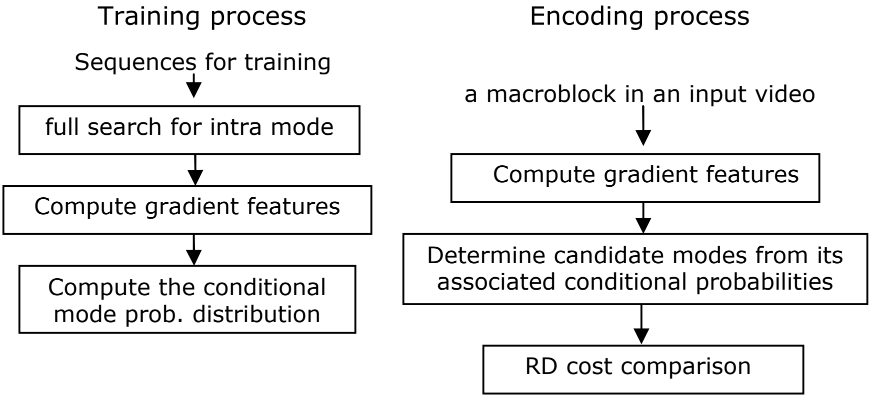


Fig. 8. The flowchart of the proposed algorithm

5 Experimental Results

We implement our algorithm in JM 10.2. The option for transform 8x8 is off, and the new intra 8x8 for luma component is also off. Based on the same experimental setting, we compare the efficiency and compression performance of the proposed intra mode decision algorithm with other methods [4-5].

Table 2-5 summarize the experimental results of these different intra mode decision methods for some benchmarking CIF and QCIF video sequences. The numbers in parentheses denotes the number of frames in the sequences. The sequence name with gray background means this sequence was included in our training dataset. In our experiments, we encode all video sequences in all I-frames. From the

Table 2. Experimental comparison on QCIF (176x144) sequences

Note. QP=28		Akiyo (300)	Carphone (96)	Claire (494)	Foreman (300)	Grandma (870)	Hall_monitor (300)
JM 10.2	PSNR	39.6648	38.7239	41.0825	37.8954	37.991	38.8264
	BITRATE(KB/s)	544.46	325.228	355.115	729.723	634.201	674.943
	TIME(ms)	138655	46959	210396	152354	414618	150699
Pan et al.[4]	PSNR↓	-0.0988	-0.0959	-0.141	-0.1006	-0.1569	-0.0983
	BITRATE↑	17.34%	14.40%	25.75%	10.58%	8.09%	11.86%
	TIME SAVING	52.46%	52.60%	53.87%	53.44%	54.48%	52.50%
Wang et al. [5]	PSNR↓	-0.0563	-0.0464	-0.1165	-0.1012	-0.0625	-0.0634
	BITRATE↑	8.79%	6.87%	9.86%	10.26%	4.74%	6.14%
	TIME SAVING	60.96%	59.96%	61.36%	59.88%	61.17%	59.59%
Proposed algorithm	PSNR↓	-0.029	-0.0464	-0.0728	-0.0613	-0.0667	-0.0691
	BITRATE↑	2.68%	2.38%	3.70%	2.08%	2.20%	1.97%
	TIME SAVING	64.50%	64.49%	64.37%	64.35%	64.67%	64.72%

Table 3. Experimental comparison on QCIF (176x144) sequences (cont.)

Note. QP=28		mother-daughter (300)	News (300)	Salesman (449)	Silent (300)	Suzie (150)
JM 10.2	PSNR	39.1073	38.5508	37.248	37.2562	39.4939
	BITRATE(KB/s)	467.291	791.906	843.803	791.274	442.549
	TIME(ms)	136670	157224	241274	157249	67846
Pan et al.[4]	PSNR↓	-0.1432	-0.0569	-0.1467	-0.1383	-0.145
	BITRATE↑	14.33%	12.85%	10.90%	10.89%	10.55%
	TIME SAVING	54.08%	53.99%	54.46%	55.71%	54.68%
Wang et al. [5]	PSNR↓	-0.0667	-0.0158	-0.0625	-0.052	-0.0506
	BITRATE↑	7.87%	6.37%	4.51%	5.94%	5.80%
	TIME SAVING	61.11%	61.90%	61.00%	60.27%	59.79%
Proposed algorithm	PSNR↓	-0.0609	-0.0651	-0.0883	-0.0621	-0.0596
	BITRATE↑	3.55%	2.49%	2.07%	2.46%	3.15%
	TIME SAVING	64.39%	65.41%	65.35%	65.17%	65.09%

Table 4. Experimental comparison on CIF (352x288) sequences

Note. QP=28		Coastguard (300)	Container (300)	Foreman (300)	Highway (2000)	Stefan (90)	Tempete (260)
JM 10.2	PSNR	37.2135	38.1535	38.393	39.1406	37.4565	36.9332
	BITRATE(KB/s)	3155.139	2466.599	2310.821	1062.933	4287.33	4447.909
	TIME(ms)	703916	639519	613121	3621431	218673	645080
Pan et al.[4]	PSNR↓	-0.1649	-0.1138	-0.1022	-0.0558	-0.1399	-0.1656
	BITRATE↑	10.76%	11.13%	10.58%	16.83%	10.13%	8.72%
	TIME SAVING	56.96%	56.99%	54.54%	52.14%	56.53%	56.34%
Wang et al. [5]	PSNR↓	-0.0749	-0.0639	-0.0811	-0.0288	-0.1012	-0.0827
	BITRATE↑	4.68%	5.37%	8.56%	10.75%	4.85%	4.15%
	TIME SAVING	64.36%	64.44%	62.11%	63.08%	63.44%	63.54%
Proposed algorithm	PSNR↓	-0.0541	-0.0516	-0.0559	-0.0118	-0.1053	-0.1045
	BITRATE↑	0.85%	1.26%	1.87%	2.25%	1.85%	1.42%
	TIME SAVING	67.55%	67.05%	66.23%	66.51%	66.55%	66.96%

Table 5. Experimental comparison on SIF (352x240) sequences

		Football (125)	Garden (115)	Mobile (140)	Tennis (112)
JM 10.2	PSNR	35.6657	35.6175	35.2449	36.2946
	BITRATE(KB/s)	4395.85	6826.50	7547.33	3251.02
	TIME(ms)	280307	301705	373034	222486
Pan et al.[4]	PSNR↓	-0.1476	-0.2412	-0.1916	-0.0838
	BITRATE↑	4.90%	2.38%	4.80%	4.72%
	TIME SAVING	57.97%	59.87%	58.81%	55.65%
Wang et al. [5]	PSNR↓	-0.0829	-0.13	-0.1185	-0.0554
	BITRATE↑	2.77%	1.74%	2.49%	2.19%
	TIME SAVING	63.60%	63.98%	64.21%	63.21%
Proposed algorithm	PSNR↓	-0.0814	-0.1505	-0.1271	-0.0601
	BITRATE↑	1.25%	0.69%	1.24%	0.38%
	TIME SAVING	67.55%	67.43%	66.80%	66.68%

experimental results, it is obvious that the proposed algorithm can achieve slightly higher bitrate with significant computational reduction. Our algorithm provides much better bitrate reduction compared to the previous methods. Note that all of the experimental results are obtained by using scheme (a) in the gradient feature extraction. We show the performance of both scheme (a) and (b) in our algorithm from the RD curves for four different sequences in Figure 9.

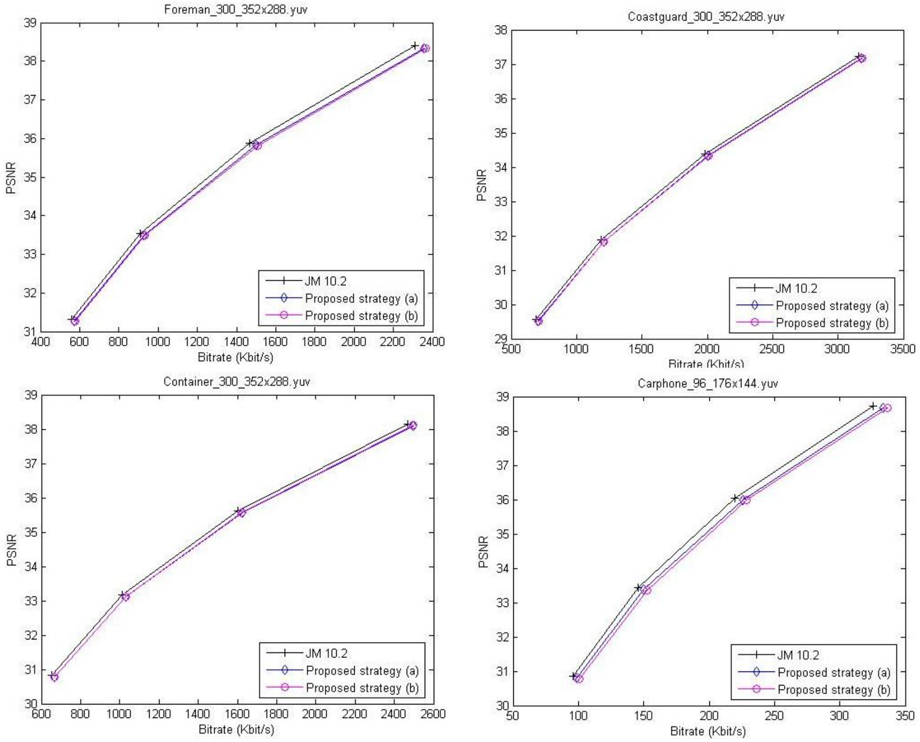


Fig. 9. RD-Curve comparison of the two schemes in the proposed algorithm with the JM reference program on four different video sequences. These RD curves are obtained with QP set to 28, 32, 36, and 40.

6 Conclusion

In this paper, we proposed a learning based algorithm for efficient intra mode decision. The conditional probability distribution of the optimal mode given the neighboring encoded modes and the image gradient features is learned from a collection of video sequences encoded with H.264. In the proposed fast mode decision algorithm, a small number of candidate modes are selected based on the associated mode conditional probability distribution to reduce the search for the optimal RD cost. Our experimental results show the proposed algorithm significantly reduces the computational cost with negligible bitrate increase. Compared with other previous methods, the proposed algorithm consistently outperforms the other methods on different video sequences.

Acknowledgements

This research work was supported in part by MOEA project under grant 95-EC-17-A-01-S1-034 and National Science Council, Taiwan, under grant 95-2220-E-007-028.

References

1. ITU-T Rec. H.264/SO/IEC 11496-10, Advanced video coding, Final Committee Draft, Document IVT F100 (December 2002)
2. Cheng, C.-C., Chang, T.-S.: Fast three step intra prediction algorithm for 4x4 blocks in H.264. In: Proc. IEEE Intern. Symp. Circuits and Systems, vol. 2, pp. 1509–1512 (2005)
3. Zhang, Y.-D., Feng, D., Lin, S.-X: Fast 4x4 intra-prediction mode selection for H.264. In: Proc. IEEE International Conf. Multimedia and Expo., vol. 2, pp. 1151–1154 (2004)
4. Pan, F., Lin, X., Rahardja, S., Lim, K.P., Li, Z.G., Wu, D., Wu, S.: Fast mode decision algorithm for intra prediction in H.264/AVC video coding. IEEE Trans. Circuits Systems for Video Technology 15(7), 813–822 (2005)
5. Wang, J.-F., Wang, J.-C., Chen, J.-T., Tsai, A.-C., Paul, A.: A novel fast algorithm for intra mode decision in H.264/AVC encoders. In: Proc. IEEE International Symp. Circuits and Systems (May 2006)
6. Sim, D.-G., Kim, Y.: Context-adaptive mode selection for intra-block coding in H.264/MPEG-4 Part 10. Real-Time Imaging 11, 1–6 (2005)
7. Huang, Y.-W., Hsieh, B.-Y., Chen, T.-C., Chen, L.-G.: Analysis, fast algorithm, and VLSI architecture design for H.264/AVC intra frame coder. IEEE Trans. Circuits and Systems for Video Technology 15(3), 378–401 (2005)
8. JVT Reference Software JM10.2, http://iphome.hhi.de/suehring/tml/download/old_jm/jm10.2.zip