Fast Mode Decision Algorithms for Inter/Intra Prediction in H.264 Video Coding

Ling-Jiao Pan, Seung-Hwan Kim, and Yo-Sung Ho

Gwangju Institute of Science and Technology (GIST)
1 Oryong-dong Buk-gu, Gwangju, 500-712, Korea
{hedypan, kshkim, hoyo}@gist.ac.kr

Abstract. In this paper, we propose fast mode decision algorithms for both intra prediction and inter prediction in H.264. In order to select the candidate modes for intra4x4 and intra16x16 prediction efficiently, we have used the spatial correlation and directional information. We have also applied an early block size selection method to reduce the searching time further. The fast inter mode decision is achieved by an early SKIP mode decision method, and a fast mode decision method for 16x16 and P8x8 modes. Extensive simulations on different test sequences demonstrate a considerable speed up by saving the encoding time up to 82% for intra prediction and 77% for inter prediction on average, compared to the H.264 standard, respectively. This is achieved at the cost of negligible loss in PSNR values and small increase in bit rates.

Keywords: H.264/AVC, video coding, fast mode decision.

1 Introduction

H.264 is the latest international video coding standard. Compared to previous video coding standards, it can achieve considerably higher coding efficiency. This is accomplished by a number of advanced features incorporated in H.264 [1]. One of the new features is multi-mode selection for intra-frames and inter-frames. In H.264, intra-frame mode selection dramatically reduces spatial redundancy in I-frames, while inter-frame mode selection significantly affects the output quality of P-/B-frames by selecting an optimal block size with motion vectors or a mode for each macroblock (MB). In H.264, the coding block size is not fixed. It supports variable block sizes to minimize the overall error.

H.264 provides a rate distortion optimization (RDO) technique to select the best coding mode among all the possible modes [2]. With this technique, we can maximize image quality and minimize coding bits by checking all the mode combinations for each MB exhaustively. However, the RDO technique increases the coding complexity drastically, which makes H.264 not suitable for real-time applications. Thus, fast mode decision methods are needed to reduce the encoding time.

Recently various efforts have been made to reduce coding complexity of H.264. Pan *et al.* [3] proposed a fast intra mode decision algorithm based on the edge direction information. However, they needed additional operations in calculating the

H.H.S. Ip et al. (Eds.): PCM 2007, LNCS 4810, pp. 158–167, 2007.

edge direction information. In other approach based on the idea of reducing possible candidate directions [4], RD costs of 6 to 7 out of 9 modes need to be computed in Intra4x4 prediction. This approach cannot select the best candidate mode efficiently because it examines unnecessary modes all the time. Recently, various fast inter mode decision algorithms have been proposed. Jeon *et al.* [5] proposed four conditions for the early SKIP mode decision. Another approach [6] not only considered the early SKIP mode, but also developed the early 16x16 mode decision using motion vectors of 16x16 and SKIP modes, reference frame, and the sub-optimal best mode. Although these methods provide fast and accurate mode decision for the SKIP mode and 16x16 mode, they still need to calculate J_{mode} (16x16) for the early SKIP mode decision. Furthermore, they do not include any fast mode decision algorithm for the P8x8 mode which consumes a large part of the encoding time.

In our work, we propose fast mode decision algorithms for both intra prediction and inter prediction. This paper is organized as follows. Section 2 describes the mode decision in H.264. Section 3 and Section4 describe fast mode decision algorithms for intra and inter prediction, respectively. Section 5 shows the simulation results, and we draw conclusions in Section 6.

2 Mode Decision in H.264

H.264 uses three different types of intra prediction for the luminance component Y. They are Intra4x4 (I4_MB), Intra8x8 (I8_MB, only for High profile) and Intra16x16 (I16_MB). In I4_MB, the prediction unit is a block of 4x4 pixels. The samples above and to the left (labeled A-M in Fig. 1(a)) have been coded and reconstructed previously and are therefore available both at the encoder and decoder to form a prediction reference. The pixels in the prediction unit are calculated based on the samples A-M by using one of the nine prediction modes. Fig. 1(b) shows the eight specific prediction directions for each mode. Mode 2 (DC mode) is not a directional mode, and all pixels are predicted by the mean of samples A-M. In I16_MB, only four prediction modes are applied to the whole macroblock. They are vertical prediction, horizontal prediction, DC prediction and plane prediction. The prediction method is similar to the I4_MB case. The only difference is that they are applied to the whole macroblock, instead of the 4x4 unit. The four chroma prediction modes are very similar to those of the I16_MB prediction, except that the order of modes is different.

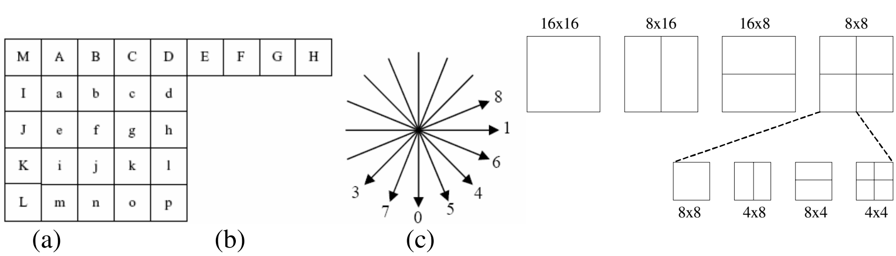


Fig. 1. (a) 4x4 intra block and neighboring pixels (b) Eight prediction directions for intra 4x4 prediction (c) Inter macroblock partitions and P8x8 sub-partitions

H.264 also supports inter prediction to reduce the temporal redundancy. H.264 uses seven different block sizes (16x16, 16x8, 8x16, 8x8, 8x4, 4x8, and 4x4) for interframe motion estimation/compensation. These different block size actually form a two-level hierarchy tree structure inside each MB. The first level includes the block size of 16x16, 16x8, and 8x16. In the second level, specified as the P8x8 type, each 8x8 block can be sub-divided into a smaller block size, such as 8x8, 8x4, 4x8 or 4x4. The relationship between these different block sizes is shown in Fig. 1(c). There is also a SKIP mode in P slice referring to the block size of 16x16, where no motion and residual information is encoded. In general, a homogeneous region of similar motions is more likely to be coded using a large block size, such as the SKIP or 16x16 mode. The area containing the boundaries of a moving object is more likely to be coded using a smaller block size. In order to decide the best motion vector (MV), the reference frame and the mode, H.264 uses the RDO method based on the Lagrangian function [2] to minimize the motion cost J_{motion} and the mode cost $J_{mode.}$

3 Fast Intra Mode Decision Algorithm

3.1 Intra Mode Decision for 8x8 Chroma Blocks

Since the choice of the prediction modes for chroma components is independent to the luma component, we can optimize chroma and luma components separately. Because the transformed coefficients are ultimately coded, we can achieve a better estimation of the mode cost using the Hadamard transform, instead of the DCT transform. SATD (the sum of absolute Hadamard transform differences) in H.264 is defined by:

$$SATD = \sum_{i}^{N} \sum_{j}^{N} |c_{ij}| \tag{1}$$

where c_{ij} denotes the (i, j)th element of C, which is the Hadamard transform of the residual block. The performance of SATD is close to the Lagrangian function, while the computational load is much lower [7]. In our work, we determine the best chroma mode by choosing the mode of the minimum SATD. Then, the following mode decision processes are performed with the best chroma mode.

3.2 Early Block Type Selection

We have observed that the block size depends mainly on the smoothness of a region. A large block size is likely to be used in homogeneous regions, while a small block size works well for complex texture regions. The main idea behind our approach is that the smooth filter does not affect the homogeneous area but will blur the details in the complex region. In our approach, we apply 1x5 and 5x1 mean-value filters to the top and left boundary of each macroblock separately. The filtered pixel value can be obtained by

$$p_{ij} = \frac{1}{5} \sum_{k=j-2}^{j+2} p_{ik} \qquad p_{ij} = \frac{1}{5} \sum_{k=i-2}^{i+2} p_{kj} \tag{2}$$

Then we calculate the sum of absolute differences between the original pixel value and the filtered pixel value (SADOF). Two thresholding operations, with the bottom threshold Th1 and up threshold Th2 are applied. If SADOF<Th1, the 16x16 intra prediction is further explored. If SADOF>Th2, the 4x4 intra prediction is adopted for the following mode decision. If SADOF is located between the two thresholds, both block sizes need to be checked. Since a large block size is preferred for higher QP, the threshold value should vary according to QP to reflect the quantization effect. Linear equations of QP in Eq. (3) and Eq. (4) are found to give a good performance, and a1, b1, a2, b2 are decided by extensive experiments.

$$Th1=a1*QP+b1 \tag{3}$$

$$Th2=a2*QP+b2 \tag{4}$$

3.3 Intra Mode Decision for 4x4 and 16x16 Luma Blocks

We have also observed that the pixels along the direction of the local edge normally have a similar value. This provides a clue for obtaining the directional information from the pixel values along the edge direction. Therefore, we can predict the best candidate mode roughly by checking NSAD (the normalized sum of absolute differences) for some selected pixel positions in the original block. Table 1 shows equations for calculating NSAD. In those equations, the pixel positions of “a” to “p” can refer to Fig.1 (a).

Table 1. NSAD for each intra prediction direction

Mode	Direction	NSAD
0	vertical	(|a-m|+|b-n|+|c-o|+|d-p|)/4
1	horizontal	(|a-d|+|e-h|+|i-l|+|m-p|)/4
3	diagonal down-left	(|b-e|+|d-m|+|l-o|)/3
4	diagonal down-right	(|a-p|+|i-n|+|c-h|)/3
5	vertical- right	(|a-j|+|b-k|+|c-l|)/3
6	horizontal- down	(|a-g|+|e-k|+|i-o|)/3
7	vertical-left	(|b-i|+|c-j|+|d-k|)/3
8	horizontal-up	(|e-c|+|i-g|+|m-k|)/3

Since Mode2 (DC mode) has no direction and are predicted by the mean of sample A-M, we apply Eq. (5) to deal with DC mode.

$$DDC = \sum_{i=0}^{3}\sum_{j=0}^{3}\left|p(x+i, y+j) - mean\right| \tag{5}$$

In Eq.(5), *mean* is the mean value of the samples A-M. If *DDC* (difference of the DC mode) is less than a threshold, the DC mode is selected as the best candidate mode for the current block and the mode with the smallest NSAD among the other eight modes is chosen as the second best mode. Otherwise, the mode with the smallest NSAD and the second smallest NSAD are selected as the best candidate mode and the second best candidate mode, respectively. Here we denote the best candidate mode by Mode C, and the second best candidate mode by Mode S.

Using NSAD and DDC, we can roughly predict the best mode. Further consideration of the spatial correlation information helps to evaluate the reliability of the best candidate. Observations show the best mode of the current block is highly correlated to its neighboring blocks. The most probable mode can be obtained from the left and above blocks. Fig. 2(a) shows the neighboring modes of the current block.

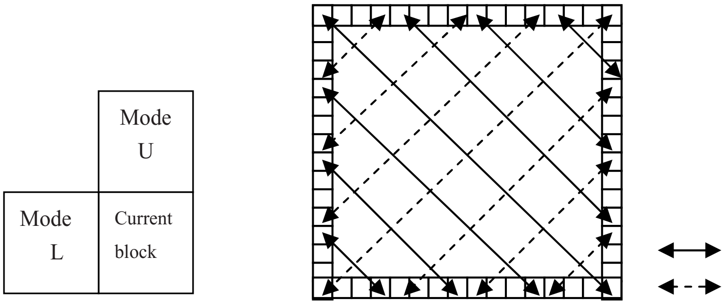
Through extensive experiments on various video sequences with different textures, we find the average probability of the current mode =L or the current mode=U is 80.86%. Under the condition of U=L, the probability of the current mode=U=L is up to 87.5%, which means that when U=L, the current mode has a very high probability to choose the same mode as U and L. Therefore, Mode U and L can be used to check the reliability of pre-predicted best candidate mode C. According to the reliability test we decide the number of candidate modes using Table 2.

Table 2. Candidate mode decision table

Condition	Reliability of mode C	Candidate modes
U=L=C	reliable	C
C=U&&C! =L	unreliable	C, S, L
C=L&&C! =U	unreliable	C, S, U
L=U&&C! =L	unreliable	C, S, L
C=!U&&C!=L&& L!=U	totally unreliable	C, S, L, U
U not available&& C=L	unreliable	C, S
U not available&& C! =L	totally unreliable	C, S, L
L not available&& C=U	unreliable	C, S
L not available&& C!=U	totally unreliable	C, S,U

The same idea is applied to the Intra16x16 luma block, except the different block size and the plane mode prediction. The plane prediction estimates a bilinear function from the neighboring pixels to the 16x16 block. It is not mathematically correct to associate the plane prediction to any directional edge. According to the plane prediction method used in the reference software [8], we use Eq. (6) to calculate the NSAD for plane prediction. In Eq. (6), Org (A_{diff}) and Org (B_{diff}) indicate the SAD of the original pixel values whose positions are pointed out by arrow A and arrow B (Fig. 2(b)), respectively. Est (A_{diff}) and Est (B_{diff}) indicate the SAD of estimated pixel values whose positions are pointed out by arrow A and arrow B (Fig. 2(b)), respectively.

$$\text{NSAD} = (A+B)/2$$
$$A = (\text{Org}(A_{diff}) - \text{Est}(A_{diff}))/7 \qquad B = (\text{Org}(B_{diff}) - \text{Est}(B_{diff}))/7 \tag{6}$$



(a) (b)

Fig. 2. (a) Neighboring blocks of the current block (b) NSAD for plane mode

3.4 Procedure of the Proposed Algorithm

Fig. 3 shows the overall structure of the proposed fast intra mode decision algorithm.

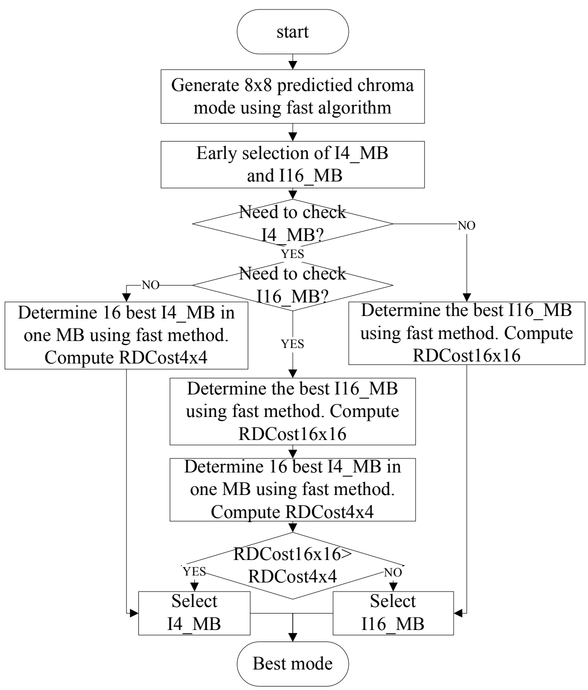


Fig. 3. Flowchart for the fast intra mode decision algorithm

4 Fast Inter Mode Decision Algorithm

In natural video sequences, we have lots of homogenous regions and when objects move, most parts of the object move in the similar direction. As we mentioned before, these areas are suitable for larger size inter mode coding. If we can detect these areas in the early stage, a significant time could be saved for the motion estimation and RDO computations of small size modes.

In our algorithm, we differentiate the SKIP mode from other block types and give it the highest priority. As mentioned before, SATD is close to the Lagrangian function while the computational load is much lower. Firstly, we calculate the SATD cost for the SKIP mode. The only information we need is the motion vector of the SKIP mode. Then, we compare the SKIP cost to the threshold value Th. If the SKIP cost is less than Th, we choose the SKIP mode as the best mode and terminate the following mode decision procedure. Since the SKIP mode is preferred for larger QP, the threshold value should vary with QP to reflect the quantization effect.

$$Th=a*QP+b \tag{7}$$

We also borrow the early 16x16 mode decision scheme from Ref. [6]. After performing motion estimation of the 16x16 block, calculating J_{mode} (16x16) and finding the SKIP motion vector (MV), we determine the best mode as the 16x16 mode when the following conditions are satisfied: (1) J_{motion} (16x16) is the smallest among J_{motion} (16x16), J_{motion} (16x8) and J_{motion} (8x16). (2) CBP (16x16) is zero. (3) SKIP MV is the same as 16x16 MV. Even though MB is not determined as the 16x16 mode, if both Condition (1) and Condition (2) are satisfied, we exclude the P8x8 mode for the best mode decision process.

Since the P8x8 mode is the most complex mode among all the modes and the frequency of the P8x8 mode increases at small QPs, it is necessary to consider fast decision for the P8x8 mode. We observe that the best prediction mode of a block is most likely to have the minimum SATD value. We simply select the P8x8 mode with the smallest SATD and inactivate the other P8x8 modes. Fig. 4 shows the flowchart of the fast inter mode decision algorithm.

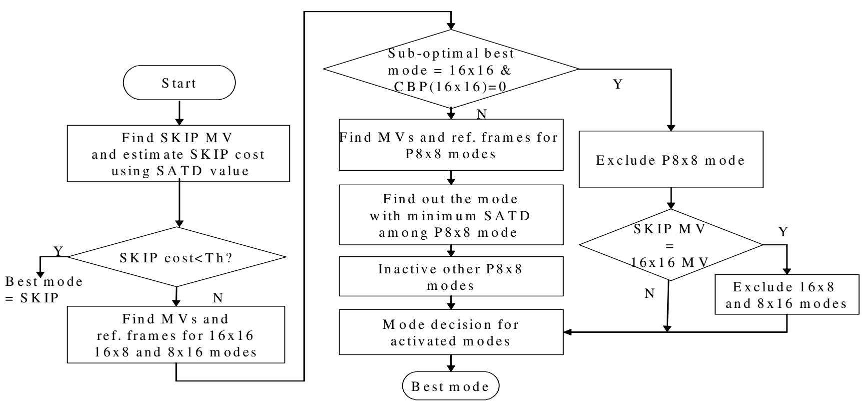


Fig. 4. Flowchart of fast inter mode decision algorithm

5 Simulation Results

The proposed algorithms are implemented on JM 11.0. We have tested several CIF (352x288) video sequences: Foreman, Bus, Coastguard, Mobile, City, Crew, Akiyo and Soccer. For each sequence, 100 frames are encoded. The frame rate is 30fps. Other simulation conditions [9] for fast intra mode decision algorithm and fast inter mode decision algorithm are shown in Table 3 and Table 4, respectively. For performance comparison, we have used the Bjonteggard delta PSNR values and Bjonteggard delta bit rates [10].

Table 3. Encoding parameters for fast intra mode decision algorithm

GOP Structure	IIII…	CABAC	Enable
Hadamard Transform	Used	QP	28, 32, 36, 40

Table 4. Encoding parameters for fast inter mode decision algorithm

GOP Structure	IPPP…	Reference Frames	5
Hadamard Transform	Used	QP	28, 32, 36, 40
Search Range	±16	FME Algorithms	UMHexagonS, CBFPS

Table 5 and Table 6 show the performance comparison of the fast mode decision algorithm for intra and inter prediction, respectively. They are relative to results by the H.264 standard. From Table 5 and Table 6, we observe that the proposed methods provide significant timesaving at the cost of negligible loss in PSNR values and a small increment in bit rates.

Fig. 5 and Fig. 6 display rate-distortion curves of “Foreman” and “Coastguard” sequences for the fast intra mode decision method and fast inter mode decision method separately. From Fig. 5 and Fig. 6, we note that the proposed methods provide a similar RD performance to the H.264 standard.

Table 5. Performance comparison for fast intra mode decision algorithm

Sequence	△PSNR_Y (dB)	△PSNR_UV (dB)	△Bits (%)	△Time (%)
Foreman	-0.03	0.18	3.645	-81.79
Bus	-0.08	0.15	2.93	-82.46
Coastguard	-0.05	0.30	2.055	-81.71
Mobile	-0.14	0.01	2.885	-83.31
City	-0.04	0.26	3.612	-80.91
Crew	-0.02	0.10	3.417	-81.57
Average	-0.06	0.17	3.091	-81.96

Table 6. Performance comparison for fast inter mode decision algorithm

Sequence	△PSNR_Y (dB)	△Bits(%)	△Time(%)
Foreman	-0.17	1.255	-80.95
Akiyo	-0.12	0.01	-91.86
Mobile	-0.12	0.397	-70.88
City	-0.08	0.456	-78.55
Crew	-0.12	0.838	-74.68
Bus	-0.11	1.256	-71.43
Soccer	-0.09	1.771	-78.40
Coastguard	-0.08	-0.68	-71.11
Average	-0.11	0.663	-77.23

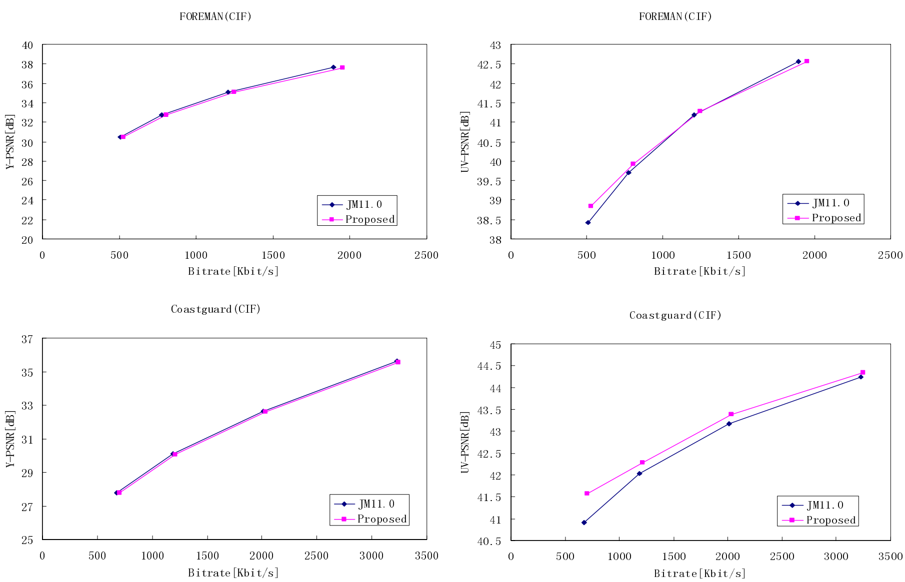


Fig. 5. RD performance of "Foreman" and "Coastguard" for intra mode decision

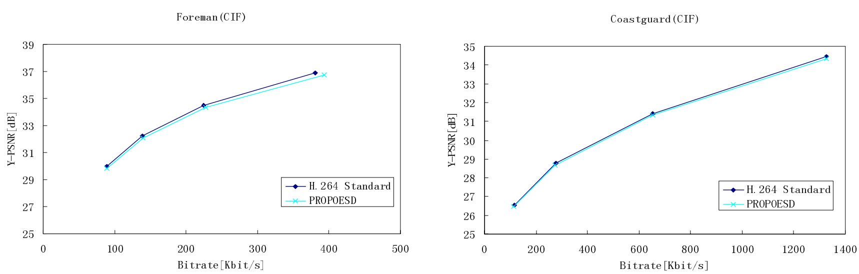


Fig. 6. RD performance of "Foreman" and "Coastguard" for inter mode decision

6 Conclusion

In this paper, we have proposed fast mode decision algorithms for both intra–frame and inter-frame to reduce the encoding complexity of the H.264 coding scheme. In the fast intra mode decision algorithm, we use the spatial correlation information and the directional information to reduce the candidate modes. We also apply an early block size selection method to reduce the computation complexity further. For the fast inter mode decision, an early SKIP mode decision method is applied in the first stage of mode decision by using SATD values. No additional information is required for the early selection. The early 16x16 mode decision and fast P8x8 mode decision methods are also developed to achieve further speedup. Results of extensive computer simulations demonstrate that the proposed algorithms can achieve time saving up to 82% and 77% in intra-coding and inter-coding, respectively, compared to the H.264 standard reference software. These considerable encoding time reductions are achieved at the cost of negligible loss in PSNR values and a small increase in bit rates.

Acknowledgments. This work was supported in part by the Institute for Information Technology Advancement (IITA), in part by the Ministry of Information and Communication (MIC) through the Realistic Broadcasting Research Center (RBRC) at Gwangju Institute of Science and Technology (GIST), and in part by the Ministry of Education (MOE) through the Brain Korea 21(BK21) project.

References

1. Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A.: Overview of the H.264/AVC video coding standard. IEEE Trans. Circuits and Systems for Video Technology 13(7), 560–576 (2003)
2. Sullivan, G.J., Wiegand, T.: Rate-distortion optimization for video compression. IEEE Signal Processing Magazine 15, 74–90 (2003)
3. Pan, F., Lin, X., Rahardja, S., Lim, K.P., Li, Z.G., Wu, D., Wu, S.: Fast mode decision algorithm for intraprediction in H.264/AVC video coding. IEEE Trans. Circuits and Systems for Video Technology 15(7), 813–822 (2005)
4. Ri, S.H., Ostermann, J.: Intra-Prediction Mode Decision for H.264 in Two Steps. In: Picture Coding Symposium 2006 (April 2006)
5. Choi, I.C., Choi, W.I., Lee, J.Y., Jeon, B.W.: The Fast Mode Decision with Fast Motion Estimation. JVT Doc. JVT-N013
6. Kim, G.Y., Yoon, B.Y., Ho, Y.S.: A Fast Inter Mode Decision Algorithm in H.264/AVC for IPTV Broadcasting Services. In: Proc. SPIE, vol. 6508 (2007)
7. Lim, K.P., Sullivan, G.J., Wiegand, T.: Text description of joint model reference encoding methods and decoding concealment methods. JVT-N046, JVT of ISO/IEC MPEG and ITU-T VCEG (January 2005)
8. JVT Reference Software version JM11.0, http://iphome.hhi.de/suehring/tml/download/
9. Sullivan, G.J., Bjontegaard, G.: Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive scan Source Material. ITU-T Q.6/16, Doc. VCEG-N81 (2001)
10. Bjontegaard, G.: Calculation of Average PSNR Difference between RD-Curve. ITU-T Q.6/16, Doc. VCEG-M33 (2001)