A Fast Block Mode Decision Scheme for P- Slices of High profile in H.264/AVC

Jongho Kim#1, Unkyung Pahk*2, Munchurl Kim*3, Jinsoo Choi#4
#Broadcasting and Telecommunications Media Research Department,
Electronics and Telecommunications Research Institute
138 Gajeongno, Yuseong-gu, Daejeon, 305-700, Korea
**Laboratory for Multimedia Computing, Communications and Broadcasting*
School of Engineering, Information and Communications University
119 Munjiro, Yuseong-gu, Daejeon, 305-732, Korea
E-mail: 1pooney@etri.re.kr, 2hwihwang@icu.ac.kr, 3mkim@icu.ac.kr, 4jschoi@etri.re.kr

ABSTRACT

The recent H.264/AVC video coding standard provides a higher coding efficiency than previous standards. H.264/AVC achieves a bit rate saving of more than 50 % with many new technologies, but it is computationally complex. Most of fast mode decision algorithms have focused on Baseline profile of H.264/AVC. In this paper, a fast block mode decision scheme for P- slices in High profile is proposed to reduce the computational complexity for H.264/AVC because the High profile is useful for broadcasting and storage applications. To reduce the block mode decision complexity in P- pictures of High profile, we use the SAD value after 16 × 16 block motion estimation. This SAD value is used for the classification feature to divide all block modes into some proper candidate block modes. The proposed algorithm shows average speed-up factors of 47.42 ~ 67.04% for IPPP sequences.

Keywords: image compression, H.264/AVC, fast block mode decision, video procession, image coding, inter mode decision

I. INTRODUCTION

H.264/AVC is a recent video coding standard [1] which has been standardized by the joint video team of ISO/IEC MPEG and ITU-T VCEG. Several techniques have been adopted for high compression performance, including variable block-size motion compensation, variable intra mode prediction modes based on spatial correlation, multiple reference pictures, context-based adaptive binary arithmetic coding (CABAC) for entropy coding, and quarter-pel motion vectors. H.264/AVC achieves excellent performance with these coding techniques, but it is also computationally very expensive. Because of very high computational complexity, there are some limitations in using the H.264/AVC codec in the real-time application part or embedded systems.

Especially, the motion estimation with variable block sizes and the intra mode prediction especially entail high computational complexity during encoding process.

To reduce the computational complexity of H.264/AVC, a number of fast algorithms have been proposed. In the fast motion estimation algorithm for early determination of the best matching point [2]-[3], the proper block modes among the various block modes that are determined by an inter-mode [3]-[10] and intra prediction [11]-[13].

Problems with H.264/AVC remain still, including a complex and time-consuming block mode decision process with motion estimation of variable block sizes. The block modes include SKIP (Mode 0), 16 × 16 (Mode 1), 16 × 8 (Mode 2), 8 × 16 (Mode 3), 8x8 (Mode 4), 8x4 (Mode 5), 4x8 (Mode 6), and 4x4 (Mode 7). Generally, SKIP, 16 × 16, 16 × 8, and 8 × 16 are called large block size modes and 8 × 8, 8 × 4, 4 × 8, and 4 × 4 are called small size blocks or sub block modes (p8 × 8). A variable block size for the block mode decision maximizes the coding efficiency based on rate-distortion optimization (RDO) where all modes should be checked for every macroblock (MB). Therefore, it is essential to reduce the inter mode decision complexity.

Lee et al. [3] suggested a two-path method based on the rate-distortion cost difference ratio. Grecos at al. [4] suggested a skip detection algorithm based on the mode information of neighbourhood MBs. The spatial-temporal predictor was also used to determine the skip condition in this algorithm. A threshold value based on the ratio is used to select the proper search mode. Wu et al. [5] proposed an algorithm based on the spatial homogeneity of a video object's texture and the temporal stationary characteristics inherent in a video sequence. Feng et al. [6] used mode groups with similar distributed characteristics to predict the most probable mode of an MB on the basis of the modes of neighbouring MBs. A method based on contextual prediction for a fast mode decision was proposed by Kim et al. [7] in which the conditional probability was analysed to obtain contextual mode information.

Herein, we introduce a fast block mode decision scheme using the SAD value after 16 × 16 block motion estimation for P-picture coding in High profile of H.264/AVC. We define several ranges using the average SAD values of block modes. Then, we find the proper range for the SAD value of the 16 × 16 block mode, and consider the mode of the range as a candidate search mode.

From the experiment results, we show that the best mode is similar to the mode of range with high probability. Briefly, if the range is between the average SAD value of the SKIP mode and the average SAD value of the 16× 16 block mode, the probability that the current mode could be selected as the SKIP mode is very high. In this case, the SKIP mode is just added to the candidate search mode.

II. FAST BLOCK MODE DECISION ALGORITHM

The block mode decision procedure usually occupies over 50% of the total encoding time. To efficiently reduce the complexity of the block mode decision, the candidate search mode grouping method has been adopted in most of the block mode decision algorithms. How to select the accurate candidate mode group is very important. We suggest a candidate grouping method based on the SAD value distribution characteristic of each block mode to reduce the computational complexity.

A. Observation for Statistical Analysis

The SAD value can represent the block mode characteristic. Generally, we can confirm that SKIP mode or large block mode has smaller SAD value than p8× 8 sub-block modes in most cases. The final block mode may be selected as one of the p8× 8 sub-block mode, when the motion or boundary of the block is complex in most cases. Therefore, the SAD value between current block and reference block increases.

If there are some specified ranges of SAD values for each block mode, we can easily determine the proper candidate search modes. We plot the average SAD value of each block mode for many sequences to prove whether each mode has its own SAD value range.

The average SAD value of SKIP mode is smallest while the average SAD value of the p8× 8 sub block mode is largest. The average SAD values of the 16× 16, 16× 8 and 8× 16 are similar. It is observed that the distributions of SAD values for each block mode are similar to Gaussian distribution. This means that each block mode can be selected as the best mode more frequently when the SAD value of the block is near the mean of each SAD distribution.

In this paper, we design adaptive ranges with the average SAD values for block mode decision by grouping the modes which have similar average SAD values. Some types of average SAD values are shown in Table 1. Table 2 shows the adaptive ranges we obtained. Range 1 has the smallest SAD value and the range 5 has the largest SAD value.

Table 1. Types of average SAD values

Index	Meaning (N-1: previous frame)
$SAD_{Mean\_SKIP}^{N-1}$	The average SAD of the SKIP mode
$SAD_{Mean\_largeblock}^{N-1}$	The average SAD of the large block mode
$SAD_{Mean\_frame}^{N-1}$	The average SAD of the previous frames
$SAD_{Mean\_p8\times8}^{N-1}$	The average SAD of the P8x8 sub-block mode

Table 2. Adaptive range set-up with the average SAD values

Index	Ranges
Range 1	0 ~ $SAD_{Mean\_SKIP}^{N-1}$
Range 2	$SAD_{Mean\_SKIP}^{N-1}$ ~ $SAD_{Mean\_largeblock}^{N-1}$
Range 3	$SAD_{Mean\_largeblock}^{N-1}$ ~ $SAD_{Mean\_frame}^{N-1}$
Range 4	$SAD_{Mean\_frame}^{N-1}$ ~ $SAD_{Mean\_p8\times8}^{N-1}$
Range 5	Over $SAD_{Mean\_p8\times8}^{N-1}$

Experiments with many sequences and various quantization parameters (QP) show that if the SAD value after motion estimation is within any range, the final mode is highly correlated with the block mode of the upper boundary of each range. Table 3 shows that the occupation ratio of the best block mode when the inter best SAD is confined in each range. For example, if the best SAD value is in range 2, the average SAD value of the boundary is SKP and large block modes. Thus, the final mode can be selected as the SKIP mode or one of the large block modes. Therefore, we can consider the modes are related with the boundary of the range as the candidate search modes.

The SKIP mode is the most probable mode in range 1 (SKIP: 71%) and one of the large block or SKIP mode is the most frequent mode in range 2 (SKIP: 46%, large block mode: 42%). The occurrence ratio of the large block mode is the highest in range 3. The large block mode and p8× 8 sub-block mode is distributed with similar ratios in range 4. Even though the boundaries of the range 4 is the average SAD value of the large block mode and the mean SAD of the p8× 8 sub-block mode, the ratio of the p8× 8 sub-block mode is relatively small. Because both large block mode and SKIP mode are more frequently occurring modes in most of the sequences. Also we can see that the p8× 8 sub-block mode occupies the largest portion in range 5.

Table 3. Occupation ratio of each block mode in various ranges of average SAD values

Index	SKIP	16x16	16x8	8x16	p8x8	Intra
Range 1	71.61	13.39	3.16	3.13	1.96	6.72
Range 2	46.53	26.72	7.55	7.58	10.36	1.24
Range 3	28.13	29.63	8.9	8.56	23.34	1.4
Range 4	17.76	26.61	10	9.31	34.84	1.45
Range 5	6.29	19.71	8.58	7.87	54.5	3

From the results, we can notice that the SAD values of each block mode have a specified range. We use the SAD value of the 16× 16 mode as a feature to classify the proper candidate search mode.

B. Proposed Fast Mode Decision Algorithm for P-Slices

We propose a fast mode decision method for P-slices based on the characteristics of the observation. We also adopt the technique based on spatial-temporal correlation. There is strong correlation between adjacent inter frames Therefore, the probability that the best mode of the current MB and collocated MB of the previous frame belongs to the same block mode group (SKIP, large block mode, sub-block mode) is very high [8]. The current block is also highly affected by its neighbour block [6], because the current block and its neighbour blocks may have similar motion and texture characteristics.

The proposed method based on this characteristic is designed as the following:

STEP 1: SKIP mode detection.

- The best mode of the collocated block is SKIP mode.
- The SAD value of the current block is smaller than that of the collocated block.

STEP 2: Calculate the predicted SAD

- The SAD value of the 16× 16 block mode

STEP 3: Check the range

- Which range does predicted SAD value is existed in.

STEP 3.1: Range 1 (Boundary: SKIP mode)

- If the best mode of the collocated block is a SKIP mode.
  Candidate Search Mode: SKIP, 16× 16
- Else
  Candidate Search Mode: SKIP, 16× 16, 16× 8, 8× 16

STEP 3.2: Range 2 (Boundary: SKIP, large block mode)

- If the best mode of the collocated block is SKIP mode or 16× 16 mode.
- If the best mode of the upper and left block is SKIP mode or 16× 16 mode.
  Candidate Search Mode: SKIP, 16× 16
- Else
  Candidate Search Mode: SKIP, 16× 16, 16× 8, 8× 16

STEP 3.3: Range 3 & 4 (Boundary: large block mode, p8× 8 sub-block mode)

- If the best mode of the collocated block is one of the large block modes.
- If the best mode of the upper and left block is one of the large block modes.
  Candidate Search Mode: SKIP, 16× 16, 16× 8, 8× 16
- Else
  Candidate Search Mode: Full Search

STEP 3.4: Range 5 (Boundary: 8× 8 sub-block mode)

- If the best mode of the collocated block is one of the large block modes.
- If the best mode of the upper and left block is one of the large block modes.
  Candidate Search Mode: SKIP, 16× 16, 16× 8, 8× 16
- Else
  Candidate Search Mode: Full Search

STEP 4: Sub-block Mode Detection

- If the best mode of the collocated block is one of the p8× 8 sub-block modes.
- If the best mode of the upper and left block is one of the p8× 8 sub-block modes.
- The p8× 8 sub-block modes are added for candidate search mode.

We only consider large block modes under the range 1 and 2 in IPPP GOP structure. From the Table 3, we can notice that the portion of the p8× 8 sub-block mode is quite a big in the range 2 and 3. The portion of the p8× 8 sub-block mode is quite small in the range 1 but it can be occurred in range 1. If we don’t consider sub-block mode in range 1, 2 and 3, the misclassification error can be increased. Thus, a sub-block mode detection method using the spatial-temporal correlation is added to increase the accuracy.

From the experiment, we can find that when the collocated block, left and upper block are encoded as the one of the p8× 8 sub-block modes, the probability that the current block is encoded as the one of the p8× 8 sub-block modes is approximately 60~80% in most sequences. The sub-block mode detection method using this correlation is mentioned in STEP 4.

III. EXPERIMENTAL RESULTS

Various MPEG standard sequences (CIF, HD) were used to see the system performance. The analysis was performed with encoding frames = 150, reference picture = 1, sequence type = IPPP, QP = 20, 24, 28, 32, and 36, CABAC enabled, and the High profile. Transform 8× 8 mode is also enabled in the experiment. It means that the intra 8× 8 search mode and the DCT8x8 mode is added for the experiments.

For the experiment, we used JM11.0 reference software by JVT. All algorithms for comparison run on same condition.

The H/W specification is Intel core 2 6600 2.40GHz CPU with 2.0Gbyte of SDRAM and a 200Gbyte HDD

We defined several measures for evaluating the performance of the proposed scheme, including the average ΔPSNR average ΔBits, and ΔTime.

The average ΔPSNR was defined for image quality comparison as:

$$\Delta PSNR = PSNR_{Proposed} - PSNR_{Full} \tag{1}$$

The average ΔPSNR represents the difference in quality variations between the average PSNR of the proposed method and the corresponding values of the full mode search.

Also, the ΔBit value, which shows bit increases between the full intra mode search and the proposed algorithm, is defined as:

$$\Delta Bits = \frac{Bit_{Proposed} - Bits_{Full}}{Bits_{Full}} \times 100 \tag{2}$$

A positive value for ΔBit indicates bit increase and performance degradation, compared with the full mode search.

ΔTime represents the amount of the encoding time saving and is defined as:

$$\Delta Time = \frac{Time_{Full} - Time_{Proposed}}{Time_{Full}} \times 100 \tag{3}$$

The average ΔTime is a complexity comparison factor to indicate the amount of total encoding time saving. Grecos's own algorithm without Jeon's method was also used for objective comparisons of the encoding performance [4].

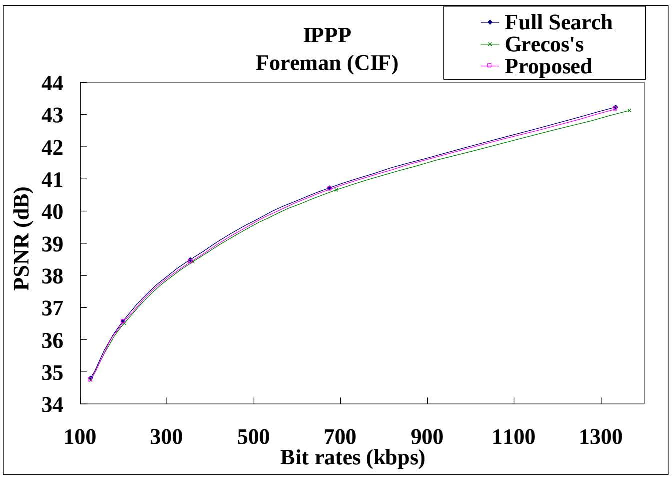
A. **Results for the IPPP GOP Structure.**

Fig. 1 shows the rate-distortion (RD) curves derived from the full mode search, the proposed fast algorithm and Grecos's algorithm. The rate-distortion performance of the proposed method is similar to the original full JM 11.0 original encoder and the rate-distortion curve of the proposed method was in a little bit upper boundary compare with Grecos's method.

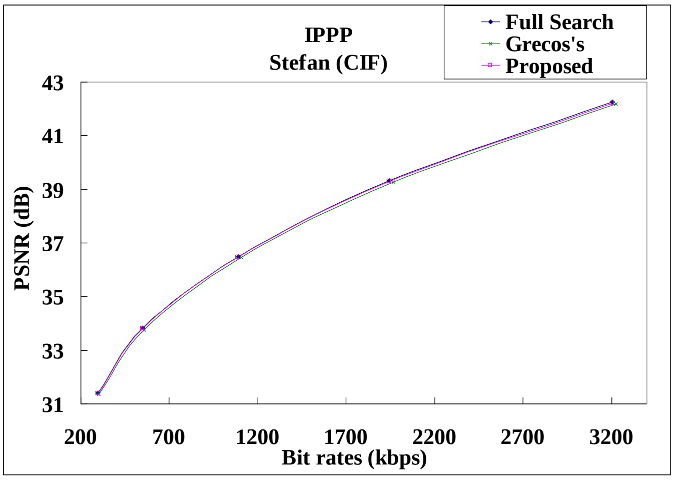
Table 4 shows overall performance of the proposed algorithm. The average loss in PSNR was -0.007dB ~ -0.081dB and total bit increment the total bit increment was approximately 0.708% ~ -1.13%, compared with the full mode search.

The proposed scheme achieved an improvement of 47.42%~ 67.04% in total encoding time. Our results are good as the △PSNR was within -0.1 dB and the increased bit requirement was approximately 1% in the most sequences. The bit requirement was only decreased in some sequences, especially high QP (low bit rates) range in CIF sequences and most of the QP range in HD sequences with a large reduction in complexity.

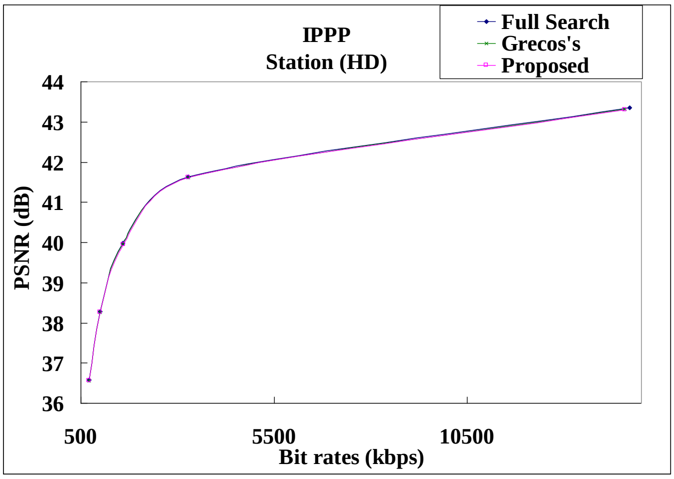
Grecos' method has a quite good performance in stationary sequences with little motion. However, it has a relatively large bit increments and PSNR loss in fast motion sequences. In a viewpoint of time reduction, the proposed method has better speed-up factor for the most sequences with a negligible loss of quality.



(a) Foreman (CIF)



(b) Stefan (CIF)



(c) Station (CIF)

Fig. 1. Rate-Distortion performance curves of IPPP sequences.

Table 4. Performance comparison of the proposed on the JM 11.0 reference encoder for IPPP structure.

IPPP GOP Structure		QP = 20			QP=24			QP=28			QP=32			QP=36		
		∆ PSNR	∆ Bits	∆ Time	∆ PSNR	∆ Bits	∆ Time	∆ PSNR	∆ Bits	∆ Time	∆ PSNR	∆ Bits	∆ Time	∆ PSNR	∆ Bits	∆ Time
Foreman (CIF)	Grecos	-0.099	2.405	29.17	-0.07	2.329	28.79	-0.055	1.695	28.39	-0.056	1.04	27.98	-0.055	0.48	27.84
	Proposed	-0.045	0.008	52.15	-0.034	0.259	54.31	-0.055	0.146	56.38	-0.034	0.127	57.31	-0.065	0.219	56.71
Container (CIF)	Grecos	-0.043	0.387	32.88	-0.034	0.6	32.39	-0.02	0.415	32.19	-0.019	-0.053	31.71	-0.027	-0.087	31
	Proposed	-0.044	-0.053	65.48	-0.053	0.284	65.75	-0.024	0.424	63.92	-0.031	-0.107	63.48	-0.029	-0.59	61.93
Akiyo (CIF)	Grecos	-0.033	1.194	32	-0.038	1.067	31.47	-0.017	0.266	30.75	-0.008	-0.038	30.02	-0.046	-0.054	29.64
	Proposed	-0.066	0.556	65.25	-0.045	0.358	66.51	-0.014	0.078	67.04	-0.019	0.094	66.43	-0.059	-0.845	64.99
Mobile (CIF)	Grecos	-0.102	1.109	32.28	-0.091	1.079	31.58	-0.082	0.804	30.62	-0.081	0.073	29.57	-0.063	-0.015	29.03
	Proposed	-0.063	0.513	54.22	-0.052	0.312	55.09	-0.049	0.015	55.46	-0.051	-0.633	54.57	-0.055	-1.13	55.81
Coastguard (CIF)	Grecos	-0.061	0.051	30.64	-0.046	0.271	29.51	-0.053	0.195	28.63	0.003	-0.151	27.65	0.02	0.145	27.77
	Proposed	-0.052	-0.131	55.37	-0.033	-0.04	55.93	-0.05	-0.367	56.71	-0.029	-0.473	57.09	-0.051	-0.576	55.81
Bus (CIF)	Grecos	-0.068	1.218	30.81	-0.057	1.883	30.74	-0.056	1.898	29.32	-0.055	1.701	28.63	-0.052	1.877	28.02
	Proposed	-0.057	0.245	58.79	-0.042	0.458	57.53	-0.043	0.233	56.08	-0.037	-0.229	56.27	-0.024	0.025	55.74
Stefan (CIF)	Grecos	-0.064	0.605	30.82	-0.044	1.209	30.19	-0.047	1.472	29.37	-0.062	1.124	28.39	-0.053	0.783	27.8
	Proposed	-0.054	-0.001	58.22	-0.025	0.042	58.96	-0.035	-0.111	57.92	-0.032	-0.138	56.81	-0.031	-0.411	57.96
Flower garden (CIF)	Grecos	-0.053	1.025	30.26	-0.037	1.652	29.48	-0.033	1.965	28.72	-0.045	1.62	28.19	-0.041	2.079	28.07
	Proposed	-0.025	0.136	47.42	-0.017	0.258	48.55	-0.012	0.221	48.55	-0.012	0.236	51.05	-0.007	0.304	52.86
Blue Sky (HD)	Grecos	-0.055	-0.531	29.62	-0.035	-0.088	29.82	-0.036	-0.236	28.78	-0.024	-0.305	28.46	-0.014	0.846	27.94
	Proposed	-0.061	-0.55	64.82	-0.043	-0.095	66.52	-0.037	-0.076	65.57	-0.028	0.08	63.95	-0.029	0.534	61.85
Station (HD)	Grecos	-0.039	-0.801	28.55	-0.011	-0.075	27.74	-0.014	0.019	28.21	-0.019	0.095	27.35	-0.022	0.206	28.13
	Proposed	-0.044	-0.84	65.1	-0.021	-0.098	64.38	-0.022	0.553	61.88	-0.028	0.144	60.55	-0.028	0.519	59.39
Sunflower (HD)	Grecos	-0.031	0.405	27.72	-0.028	0.669	27.85	-0.044	0.708	28.95	-0.063	0.655	28.1	-0.048	0.002	28.48
	Proposed	-0.074	-1.062	62	-0.069	-0.488	62.53	-0.077	-0.305	62.56	-0.081	-0.416	61.69	-0.065	-0.64	61.75

IV. CONCLUSION

We proposed a fast block mode decision scheme for P- and B-slice coding of High profile in H.264|AVC. Based on the SAD value distribution characteristic of each block mode, we make a proper candidate search mode group. We also proposed a differential mode allocation technique for lists of the IBBPBBP structure. Based on this, we can estimate more probable list and add more candidate search modes into the probable list. Through comparative analysis, 47.42 ~ 67.04% of speed-up for IPPP sequences and 41.9 ~ 60% for IBBPBB sequences were achieved in average, respectively, with negligible amounts of bit increments and negligible amounts of PSNR drops.

ACKNOWLEDGMENTS

This work was supported by MKE/IITA [2008-F-011-01, Development of Next-Generation DTV Core Technology]

REFERENCES

[1] ISO/IEC 14496-10, “Information technology–coding of aud io-visual objects–Part 10: Advanced video coding,” December 2003.

[2] B.-G. Kim, S.-T. Kim, S.-K. Song and P.-S. Mah, “Fast adaptive rood pattern search for block motion estimation,” Electronics Letters 4th, vol. 41 no. 16, August 2005.

[3] Jeyun Lee and Byungwoo Jeon, “Fast Mode Decision for H.264,”Proc. of IEEE International Conference on Multimedia and Expo (ICME), Vol. 2, pp. 1131 - 1134, 2004.

[4] C. Grecos and M. Y. Yang, “Fast Mode Prediction for the Baseline and Main Profiles in the H.264 Video Coding Standard,” IEEE Trans. multimedia, vol. 8, no. 6, pp. 1125 - 1134, Dec. 2006.

[5] D. Wu, S. Wu, K. P. Lim, F. Pan, and X. Lin, “Block inter mode decision for fast encoding of H.264", *IEEE Int'l Conference on Speech, Acoustics, and Signal Processing*, May 2004, vol. III, p. 181-184

[6] Bin Feng, Guang-xi Zhu, Wen-yu Liu, “Fast adaptive inter mode decision method in H.264 based on spatial correlation”, *IEEE International Symposium on Multimedia*, 12-14 Dec. 2005

[7] C. Kim, H.-H. Shih, C.-C. J. Kuo, “Feature-based intra prediction mode decision for H.264,” Proc. of

International Conference on Image Processing (ICIP), Vol. 2, pp. 769 - 772, 2004.

[8] Byung-Gyu Kim, "Novel Inter Mode Decision Algorithm Based on Macro-block (MB) Tracking For the P-Slice in H.264/AVC Video Coding," IEEE Tr. Circuits and Systems for Video Technology, Vol. 18, No. 2, pp. 273-279, Feb. 2008.

[9] X. Jing and L.-P. Chau, “Fast approach for H.264 inter mode decision,” Electroincs Letters, vol. 40, no. 17, September 2004.

[10] Bumshik Lee, Munchurl Kim, Sanjin Hahm, Changseob Park and Keunsoo Park, "An Efficient Block Mode Decision for Temporal Scalability in Scalable Video Coding," *Visual Communications and Image Processing (VCIP) 2008*, 27~31 January 2008, San Jose, CAL, USA

[11] F. Fu, X. Lin, and L. Xu, “Fast Intra Prediction Algorithm in H.264/AVC,” Proc. of the 7th International Conference on Signal Processing (ICSP), Vol. 2, pp. 1191 -1194, 2004.

[12] C.-C. Cheng and T.-S. Chang, “Fast Three Step Intra Prediction Algorithm for 4×4 blocks in H.264,” Proc. Of IEEE International Symposium on Circuits and Systems (ISCAS), Vol. 2, pp. 1509 - 1512, 2005.

[13] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, “Fast Mode Decision Algorithm for Intra prediction in H.264/AVC Video Coding,” IEEE Trans. Circuit Syst. Video Technol., Vol. 15, no. 7, pp. 813 - 822, July 2005