Probability-Based Mode Decision Algorithm for Scalable Video Coding

Yu-CheWey[1], Mei-Juan Chen[1] and Chia-Hung Yeh[2]
[1]Dept. of Electrical Engineering, National Dong Hwa University, Hualien, Taiwan
E-mail: 610023037@ems.ndhu.edu.tw,cmj@mail.ndhu.edu.tw
[2]Dept. of Electrical Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan
E-mail:yeh@mail.ee.nsysu.edu.tw

***Abstract*—To reduce the computational complexity of the encoding process in Scalable Video Coding, we utilize the information of motion vector predictor (MVP) and the number of non-zero coefficients(NZC) to propose a fast mode decision algorithm. The probability models of motion vector predictor and the number of non-zero coefficients are built to predict the partition mode in the enhancement layer. In addition, the search range of motion estimation is adaptively adjusted to further reduce computational complexity. Experiment results show that the proposed algorithm can reduce coding time by up to 76% in average and provide higher time saving and better performance than previous work.**

I. INTRODUCTION

Scalable Video Coding (SVC) [1-2] is the extended version of H.264/AVC, providing multi-layer for various devices. SVC supports spatial, temporal and qualities scalabilities which provide various picture sizes, frame rates and qualities for different layers, respectively. Because of the high correlation between base layer(BL) and enhancement layer(EL),SVC has three kinds of inter-layer predictions: inter-layer motion prediction, inter-layer residual prediction and inter-layer intra prediction to improve the rate-distortion efficiency of the ELs.

In H.264/SVC, the modes of macroblock(MB) which follows H.264/AVC have eight kinds of inter frame prediction modes, including Mode SKIP, Mode16×6,Mode 16×8, Mode 8×16, Mode 8×8, Mode 8×4,Mode 4×8 and Mode 4×4, and two kinds of intra frame prediction modes, i.e. INTRA 16×16 and INTRA 4×4. In order to select the best mode of the current MB, the rate-distortion cost(RD-cost) of each mode should be calculated, and the mode that has the minimum RD-cost is chosen as the best mode. Many fast mode decision algorithms have been proposed to reduce computational complexity of mode decision. Because Mode SKIP has the lowest complexity in all inter modes,[3] uses the coding information of the co-located MB in BL and the neighboring MBs of the EL to predict the SKIP mode. Wang et al.'s method [4] is based on the priority-based mode decision (PBMD) and uses the RD-cost correlation between BL and EL to decide the mode priority in EL. In [5], the algorithm employs the correlation between BL and EL to predict the best mode of the EL and uses motion vector difference (MVD)of the BL to decide the search range of the EL to accelerate the encoding time. These methods are based on just one parameter to decide the mode decision early terminating or not. The accuracy could not be so high. Kuo et al. [6] propose a method based the motion field distribution to efficiently determine the block mode for complexity reduction by likelihood.

Search range has significant impact on the coding time of motion estimation. Therefore, the way to define search range is also an important research issue in video coding. In [7], the authors propose an efficient motion re-estimation schemes for H.264 B-frame and P-frame transcoding by utilizing maximum likelihood to measure the candidates of motion vector to predict the best one. Lee et al. [8] propose a fast motion estimation scheme by applying the adaptive search range(SR). This method is based on the heuristic rule observed from experiments of motion vector characteristics. The results show that it can reduce lots of coding time while keeping good coding performance. Kim et al. [9] propose an efficient learning method to control the EL's search range according to block modes of BL.

This paper proposes an algorithm which utilizes the probability models generated by motion vector predictor (MVP) and the number of non-zero coefficients (NZC) of Mode 16×16 in EL to find the most probable modes. The proposed algorithm can reduce the number of modes required to be tested and find the suitable mode rapidly so that the encoding process of SVC can be accelerated significantly.

II. OBSERVATION AND ANALYSIS

First, we observe the motion vector predictor (MVP) and number of non-zero coefficients(NZC) of Mode 16×16 in EL. We use eight CIF sequences (Akiyo, Foreman, Football, Soccer, Stefan, Coastguard, Table and News) to record their MVP and NZC. Then we analyze their features by using Laplacian distributions. Finally, we combine this method with mode priority decision to early decide best mode.

A. Analysis of Motion Vector Predictor (MVP)

MVP is calculated by the median of motion vectors of neighboring left, top and right-top of the current MB to get the starting point for search in motion estimation. We use the

MVP to represent the motion characteristics of the current block. Usually, static blocks which mean the blocks with lower motion use larger size mode like Mode 16×16, on the contrary, high-motion blocks use smaller size mode like Mode 8×8.So we analyze the MVP of Mode 16×16 in EL and try to figure out the correlation between the MVP of Mode 16×16 and the best mode of the MB. Fig.1 shows the probability of horizontal component of MVP in each final selected mode. $Mode_1$, $Mode_2$, $Mode_3$ and $Mode_4$ represent Mode 16×16, Mode 16×8, Mode8×16 and Mode8×8, respectively. According to Fig.1, we can find that it has higher probability to choose Mode 16×16 when the MVP is small, and the MVP is more dispersed when the best mode is Mode 8×8.

B. Analysis of number of non-zero of coefficient

The number of non-zero quantized coefficients can represent the complexity of block. From this viewpoint, we also analyze the correlation between the best mode and the number of non-zero coefficients (NZC) of Mode 16×16 in EL. Fig.2 shows the probabilities of the number of non-zero coefficients in each mode. Clearly, the number of non-zero coefficients in Mode 16×16 is often fewer than other modes. When the selected best mode has smaller size, its number of non-zero coefficients will have higher probability to be larger values.

C. Mode priority decision by base layer

As mentioned in [4], Wang *et al.* analyze RD-cost correlation between BL and EL, and find out that it can efficiently decide the mode priority of EL by using the RD-costs of each $sub\text{-}MB_{BL}$. Eq. (1) shows the RD-costs of all $sub\text{-}MB_{BL}$ including Mode 8×8, Mode 8×4,Mode 4×8 and Mode 4×4after sorting. The minimum J in J_{BL} is set to be the first priority and the maximum J is set to be the last one. We use the RD-Cost order of the BL sub-MB as EL's execution order for their corresponding mode. Mode 8×8, Mode 8×4, Mode 4×8 and Mode 4×4 in BL correspond to Mode16×16, Mode16×8,Mode8×16 and Mode8×8 in EL, respectively. For example, if the first mode priority in BL is Mode 8×4, then we will execute the Mode16×8 first in EL. The mode priority of EL is represented by $Mode_{BL\text{-}p}$ (p = 1~4). $Mode_{BL\text{-}1}$ means the first priority mode and $Mode_{BL\text{-}4}$ represents the last one.

$$J_{BL} = \{ J_{BL\_p} \mid J_{BL\_1} < J_{BL\_2} < J_{BL\_3} < J_{BL\_4},\ p = 1\ to\ 4 \} \quad (1)$$

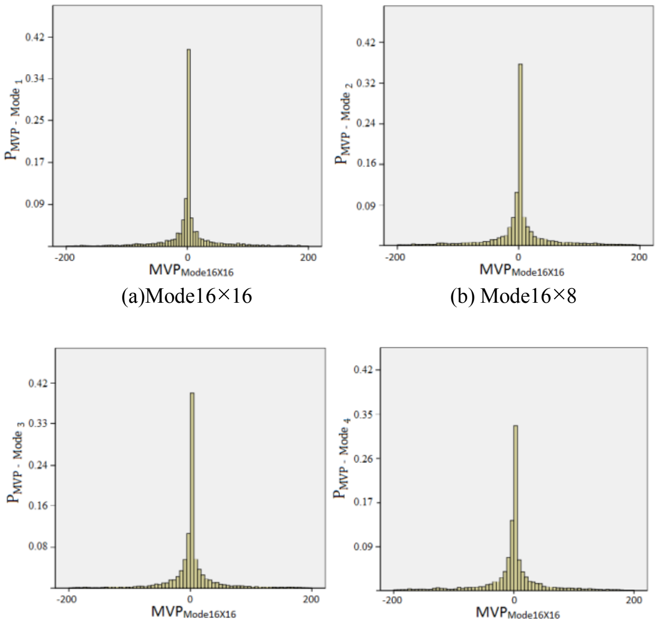
D. Adaptive search range

We also analyze the correlation between SR and mode by experiments. Through the experiments, we propose an adaptive search range algorithm to shrink the SR by the following conditions:

(a) If an MB is only coded with Mode SKIP and Mode 16×16, the search range is set as 10.

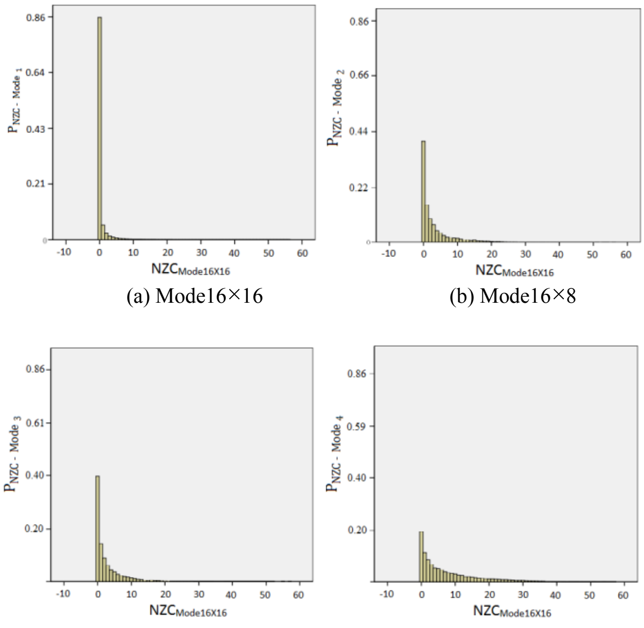
(b) Otherwise, the search range is set by $2\times MVD_{BL}$. If MVD_{BL} is less than 2, then the SR is set as 4 as shown in Eq.(2).

$$SR_{EL} = max\{2\times MVD_{BL},\ 4\} \quad (2)$$



(a)Mode16×16 (b) Mode16×8 (c) Mode8×16 (d) Mode8×8

Fig.1 Probabilities of Mode16×16's MVP in EL for final selected modes



(a) Mode16×16 (b) Mode16×8 (c) Mode8×16 (d) Mode8×8

Fig.2 Probabilities of Mode16×16's NZC in EL for final selected modes

III. PROPOSED ALGORITHM

Our proposed algorithm is using statistic models to find the most probable candidate modes. The purpose is to calculate the best estimator from known probability density function. Because of the previous analyses of MVP and NZC, we can use the probability density function to help us to predict the candidate modes of the current block. By experiment results, these two probability distributions are similar and can be approximated as Laplacian distributions, so we fit the probability distribution of MVP of Mode 16×16 as shown in eq.(3) and NZC as shown in (4), i_{MVP} and i_{NZC} represent the mode number of the most probable candidate mode predicted by MVP and NZC.

$$P_{MVP-Mode_i}(MVP) = P_{MVP-Mode_i}(MVP_x) * P_{MVP-Mode_i}(MVP_y)$$
$$= \frac{1}{2\sigma_{i,mvp_x}} exp\left[\frac{-|MVP_x-\theta_{i,mvp_x}|}{\sigma_{i,mvp_x}}\right]$$
$$* \frac{1}{2\sigma_{i,mvp_y}} exp\left[\frac{-|MVP_y-\theta_{i,mvp_y}|}{\sigma_{i,mvp_y}}\right], \ i = 1,2,3,4 \qquad (3)$$

$$P_{NZC-Mode_i}(NZC)$$
$$= \frac{1}{2\sigma_{i,NZC}} exp\left[\frac{-|NZC-\theta_{i,NZC}|}{\sigma_{i,NZC}}\right], \ i = 1,2,3,4 \qquad (4)$$

where σ_i and θ_i are the standard deviation and the median of all MVP and NZC of Mode 16×16 when $Mode_i$ is selected as the best mode.

$$i_{MVP} = {}^{arg}_{\ \ i}MaxP_{MVP-Mode_i}(MVP)\ , \ i = 1,2,3,4 \qquad (5)$$

$$i_{NZC} = {}^{arg}_{\ \ i}MaxP_{NZC-Mode_i}(NZC)\ , \ i = 1,2,3,4 \qquad (6)$$

We utilize the information of Mode 16×16 to calculate the most probable mode as the candidate mode in our algorithm. Table I shows the hit rate of selecting correct mode by using different estimators. When only one of these two candidate modes which are generated by probability models of using MVP($Mode_{i_{MVP}}$) and NZC ($Mode_{i_{NZC}}$) is used to predict the best mode, the hit rate would be under 66%. If we consider both of two candidate modes to be the best mode, the hit rate will upgrade to 77%.

Table I. Mode hit rate by probability models

Mode	$Mode_{i_{MVP}}$	$Mode_{i_{NZC}}$	$Mode_{i_{MVP}}$ and $Mode_{i_{NZC}}$
Hit Rate	58.76%	65.23%	77.56%

The flowchart of the proposed fast encoding algorithm is shown in Fig.3. We use the information of co-located MB in BL and the Mode 16×16 in EL to early select the best mode of EL in order to reduce the number of the candidate modes. The details of our algorithm are described as follows:

step 1 Check the co-located MB in BL. If it is coded by Mode SKIP, the current MB in EL will only encode with Mode SKIP and Mode 16×16. The search range will be set as 10. Otherwise the search range will be set as eq. (2), then go to step 2.

step 2 Evaluate Mode SKIP and Mode 16×16 first, and calculate the probabilities of candidate modes which use motion vector predictor according to Eq.(3) and the number of non-zero coefficients according to Eq.(4) to generate two most possible candidate modes. If two generated candidate modes including both Mode 16×16 and Mode8×8, it means the prediction results are not very consistent, go to step 3. Otherwise, go to step 4.

step 3 Evaluate all modes.

step 4 Use mode priority defined in Sec.II.C. as an order of mode checking. If candidate modes which are generated from step 2are all evaluated, then early terminate the mode decision

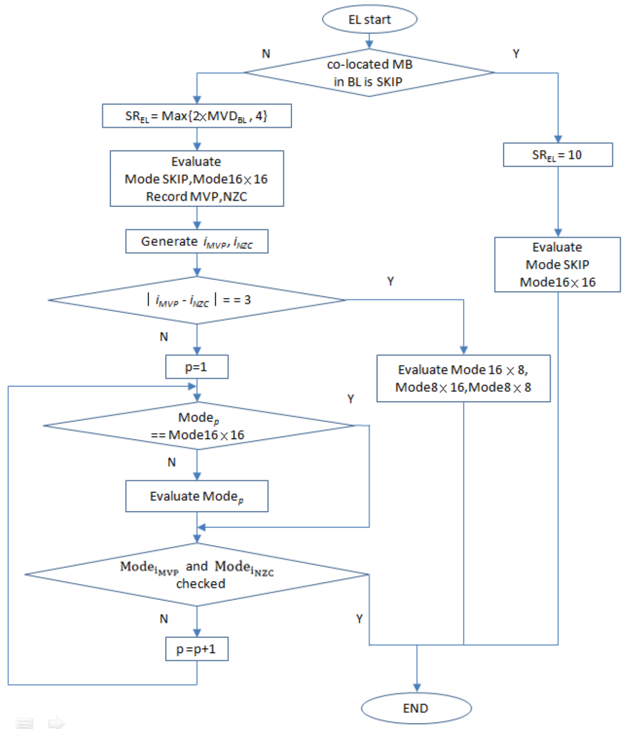


Figure3 Flowchart of the proposed algorithm

IV. EXPERIMENT RESULT

The proposed algorithm is implemented with the JSVM 9.18 reference software [10].Table II shows the details of the experimental setting. We compare the performance of our algorithm with [5].

Five measurements TS, ΔPSNR, ΔBR, BDPSNR and BDBR [11] are used the evaluate the performance of the proposed algorithm. TS, ΔPSNR andΔBR are metrics defined in Eq.(7-9).

$$Time\ saving\ (TS) = \frac{Time_{JSVM}-Time_{proposed}}{Time_{JSVM}} \times 100\% \quad (7)$$

$$\Delta PSNR = PSNR_{proposed} - PSNR_{JSVM} \quad (8)$$

$$\Delta BR = \frac{BR_{proposed}-BR_{JSVM}}{BR_{JSVM}} \times 100\% \quad (9)$$

Table II. Experimental environment

JSVMreference software	JSVM. 9.18
PC-CPU	Intel(R) Core(TM) i5-3570 3.4GHZ
Encoding format	Hierarchical B
PC-RAM	4.00GB
GOP	16
Frame encoded	150
Search range	32
Motion estimation	Full search
Entropy coding	CABAC
Frame size	QCIF/CIF , CIF/4CIF

Table III. Performance comparisons of the proposed algorithm with Lu's algorithm in different QPs (QP_{BL}/QP_{EL} = 30/25, 30/30, 30/35) for QCIF/CIF spatial scalability

Seq.	QP	Lu[5]			Proposed		
		ΔPSNR	ΔBR	TS	ΔPSNR	ΔBR	TS
Bus	30/25	-0.12	1.51	62.25%	-0.05	0.54	65.86%
	30/30	-0.09	1.08	62.86%	-0.06	0.24	67.25%
	30/35	-0.05	0.45	63.77%	-0.05	-0.09	68.13%
City	30/25	-0.04	1.86	67.84%	-0.10	0.30	76.45%
	30/30	-0.03	1.03	69.04%	-0.12	0.10	78.02%
	30/35	-0.02	0.12	70.47%	-0.08	-0.73	78.03%
Crew	30/25	-0.03	1.96	60.58%	-0.07	0.23	68.15%
	30/30	-0.01	1.02	61.29%	-0.08	-0.42	69.62%
	30/35	-0.03	0.68	62.25%	-0.06	-0.74	71.38%
Football	30/25	-0.05	0.92	57.79%	-0.03	0.27	56.80%
	30/30	-0.05	0.48	58.29%	-0.05	-0.06	58.72%
	30/35	-0.03	0.17	59.22%	-0.02	-0.16	60.45%
Foreman	30/25	-0.14	1.87	63.00%	-0.12	0.41	73.56%
	30/30	-0.07	0.61	64.23%	-0.12	-0.53	74.47%
	30/35	-0.03	-0.24	65.22%	-0.06	-0.42	75.23%
Harbour	30/25	-0.16	0.81	68.19%	-0.04	0.43	75.77%
	30/30	-0.07	0.60	68.77%	-0.05	0.00	75.94%
	30/35	-0.04	0.14	69.95%	-0.06	-0.29	76.25%
Average		-0.06	0.84	64.17%	-0.07	-0.05	70.56%

Table IV. Performance comparisons of the proposed algorithm with Lu's algorithm in the same QPs (QPBL/QPEL = 28/28, 32/32, 36/36, 40/40) for QCIF/CIF spatial scalability

Seq.	Lu[5]			Proposed		
	BDPSNR	BDBR	TS	BDPSNR	BDBR	TS
Bus	-0.10	1.96%	63.83%	-0.10	1.94%	69.62%
Foreman	-0.07	1.44%	67.79%	-0.10	1.94%	76.34%
Mobile	-0.06	1.09%	69.88%	-0.08	1.94%	76.25%
M.D	-0.02	0.43%	80.19%	-0.01	0.29%	81.85%
Average	-0.06	1.23%	70.42%	-0.07	1.53%	76.02%

Table V. Performance comparisons of the proposed algorithm with Lu's algorithm for CIF/4CIF spatial scalability

Seq. CIF/4CIF	Lu[5]			Proposed		
	BDPSNR	BDBR	TS	BDPSNR	BDBR	TS
City	-0.02	0.38%	59.13%	-0.03	0.97%	76.93%
Crew	-0.02	0.60%	59.39%	-0.04	1.25%	72.28%
Harbour	-0.04	0.97%	57.21%	-0.04	1.17%	73.77%
Soccer	-0.04	1.05%	59.85%	-0.05	1.24%	72.48%
Average	-0.03	0.75%	58.90%	-0.04	1.16%	73.87%

The performance of our proposed algorithm is compared with Lu et al.'s method shown in Table III-V. Regardless of the QPs between BL and EL, higher coding efficiency with lower bitrates can be achieved by our algorithm. It means our proposed algorithm can efficiently select the best mode precisely. In high motion sequences such as Bus, Football, or Soccer, the abundance in block information leads to better prediction and better performance. For higher resolution sequences, it is clear that our algorithm can achieve higher time saving than Lu's.

V. CONCLUSIONS

This paper proposes a fast mode decision by using the statistic models. Our algorithm not only decides best mode but also adaptively reduce search range of motion estimation. The proposed method has up to 76% time saving in average and also has better coding performance in terms of coding time than the previous work.

VI. REFERENCES

[1] T. Wiegand, G. Sullivan, J. Reichel, H. Schwarz and M. Wien, "Joint Draft ITU-T Rec. H.264 | ISO/IEC 14496-10 / Amd.3 Scalable video coding," *Joint Video Team (JVT)JVT-X20*1, 2007.

[2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp.1103-1120,Sept. 2007.

[3] L. Shen, Y. Sun, Z. Liu, and Z. Zhang, "Efficient SKIP mode detection for coarse grain quality scalable video coding," *IEEE Signal Processing Letters*, vol. 17, no. 10, pp.887-890, October 2010.

[4] T.H. Wang, M.J. Chen, M.C. Chi, S.F. Huang and C.H. Yeh, "Computation-scalable algorithm for scalable video coding," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 3, pp.1194-1202, August 2011.

[5] X. Lu, and G.R. Martin, "Fast mode decision algorithm for the H.264/AVC scalable video coding extension," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 5, pp. 846-855, May 2013.

[6] T.Y. Kuo and C.H. Chan, "Fast variable block size motion estimation for H.264 using likelihood and correlation of motion field," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16,no. 10, pp.1185-1195, October 2006.

[7] S.C. Huang, C.T. Hsu, M.J. Chen, "Efficient motion re-estimation for H.264 B- and P-Frame transcoding by using maximum likelihood," *in Proceedings of International Conference on Electronics and Information Engineering*, pp.V1-557 - V1-561, 2010.

[8] J. Lee, M. Choi, Y. Cho, J. Kim and W.K. Cho, "Fast H.264/AVC motion estimation algorithm using adaptive search range," *in Proceedings of the12thInternational Symposium on Integrated Circuits,* pp. 336-339, 2009.

[9] B.G. Kim, K. Reddy, and K.W. Lim, "Dynamic search range control algorithm for inter-frame coding in scalable video coding," *in Proceedings of IEEE International Conference on Multimedia and Expo*, pp. 225-228, 2009.

[10] JSVM (Joint Scalable Video Model) reference software for SVC. [Online].Available:http://ip.hhi.de/imagecom/G1/savce/downloads/SVC Reference-Software.htm

[11] G. Bjϕntegaard, "Calculation of average PSNR differences between RD curves," ITU-T VCEG-M33, VCEG 13th meeting, Austin, Texas, USA.