

An Efficient Mode Preselection Algorithm for Fractional Motion Estimation in H.264/AVC Scalable Video Extension

Gwo-Long Li and Tian-Sheuan Chang, *Senior Member IEEE*

Abstract—The video coding standard, H.264/AVC scalable video extension (SVC), adopts various advanced interlayer prediction modes to explore the data redundancies between layers for better coding efficiency but at the expense of significantly increased computational complexity and data access bandwidth, especially for hardware realization of fractional motion estimation mode decision. To deal with this problem, this paper proposes a mode preselection algorithm for fractional motion estimation in scalable video coding. We first analyze the rate distortion cost relationship between different prediction modes. With the statistical results, several mode preselection rules are proposed to filter out the potentially skippable prediction modes. Simulation results show that our proposed algorithm reduces up to 65.97% prediction modes and 79.79% coding time on average with only 0.036 dB and 0.496% BD-peak signal-to-noise ratio (PSNR) degradation and BD-rate increase, respectively. Furthermore, the proposed mode preselection algorithm has been implemented in hardware and it costs only 9k gate counts, when synthesized by 90nm CMOS technology.

Index Terms—Fractional motion estimation, scalable video coding.

I. INTRODUCTION

FRACTIONAL motion estimation (FME) has been widely adopted in the existing video coding standards [1]–[3] to improve the coding efficiency, and [4] reported that up to over 4 dB peak signal-to-noise ratio (PSNR) improvement can be achieved. Fig. 1 illustrates the concept of FME used in H.264/AVC reference software [5] with half pixel and quarter pixel position motion estimation. These subpixel values are interpolated by a six-tap filter for half pixels and a bilinear filter for quarter pixels. Although FME only checks several positions around the best motion vector from integer motion estimation (IME), the computational complexity of integer motion estimation and FME are almost equal to each other in the hardware realization perspective due to the complex

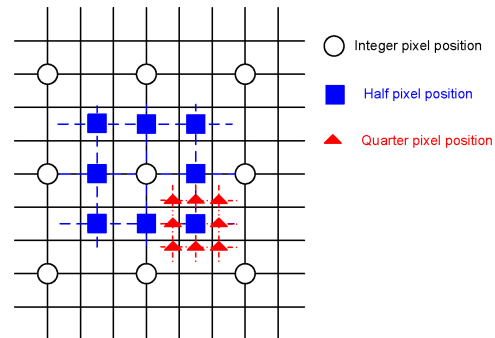


Fig. 1. Fractional motion estimation.

interpolation process, and lots of prediction modes needed to be checked [4], [6]. As a result, the operations of IME and FME are usually divided into two distinct pipeline stages in hardware design [7] to balance the computational complexity of pipeline stage.

Lots of prediction mode numbers of FME cause the serious computational bottleneck. Fig. 2 shows the mode selection in H.264/AVC video coding standard [1]. It adopts variable block size motion estimation in Inter prediction mode and each partition size has to be checked by IME and FME one by one. Thus, 41 blocks have to be examined by IME and FME operation. However, in addition to the intrinsic prediction modes supported in H.264/AVC, H.264/AVC scalable extension (SVC) [3] supports several extra interlayer prediction modes including interlayer motion (ILM), interlayer residual (Residual), InterBL, and interlayer texture (IntraBL) to further improve the coding performance, as shown in Fig. 3. Although the interlayer prediction modes can improve the coding performance of SVC, the computational complexity of FME is increased significantly, which needs to examine $41 \times 4 = 164$ blocks in SVC.

Several researches were proposed to solve the above problem to simplify the hardware design complexity. In [7], the small blocks, ranging from 8×8 to 4×4 , are decided early in IME stage to derive a submode, and thus only partition sizes of 16×16 , 16×8 , 8×16 , and submode have to be examined by FME operation, as shown in Fig. 4(a). Although the early decision method for submode can efficiently reduce the overheads of FME, the computational complexity of FME is still high. Several works [8]–[11] have been proposed to

Manuscript received August 20, 2012; revised November 27, 2012; accepted January 16, 2013, January 21, 2013. Date of publication February 22, 2013; date of current version November 1, 2013. This paper was recommended by Associate Editor F. Wu.

G.-L. Li is with the Industrial Technology Research Institute, Hsinchu, Taiwan (e-mail: glli@itri.org.tw).

T.-S. Chang is with the Department of Electronics Engineering, National Chiao-Tung University, Hsinchu, Taiwan (e-mail: tschang@dragons.ee.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2013.2248496

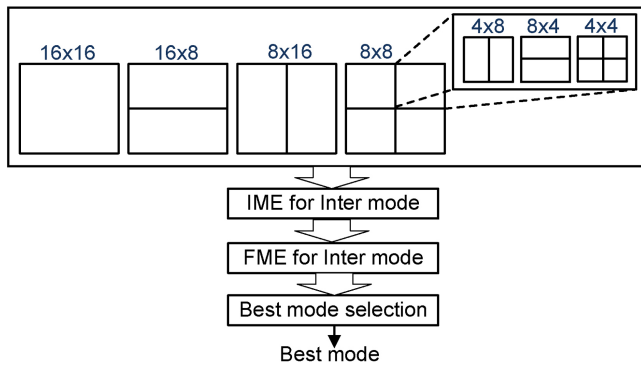


Fig. 2. Mode selection process of H.264/AVC.

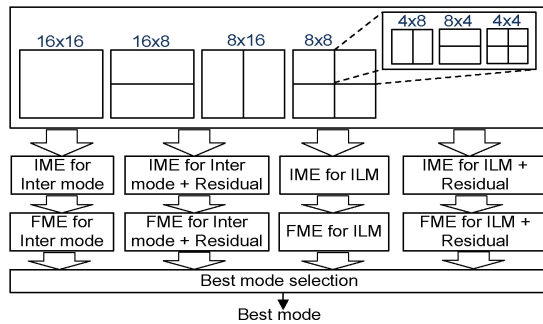


Fig. 3. Mode selection process of SVC.

increase the coding speed of FME through hardware implementation. In contrast to check all prediction modes, [12] and [13] proposed the mode preselection method, as shown in Fig. 4(b), to preselect the potentially skippable prediction modes before entering FME prediction process in H.264/AVC, but they cannot be applied to SVC directly. Several literatures [14]–[18] have been proposed to select best mode in SVC. However, these works did not take the information of FME into account for mode decision. [19] proposed a two-dimensional mode predecision algorithm for SVC by using the mode relationship between spatial layers. However, this literature did not consider the rate distortion cost relationship between IME and FME, which has a higher correlation between each other. In addition, the information of interlayer prediction has not been considered in this literature as well.

To solve this high-computational complexity problem, this paper proposes an efficient mode preselection algorithm to lighten the computational complexity of FME for SVC by using the concept of mode preselection. The proposed mode preselection is based on the statistical analysis of the rate distortion cost relationship between different prediction modes. With this analysis, we can preselect the potentially ignorable prediction modes and reduce the complexity significantly.

The rest of this paper is organized as follows. In Section II, the supported interlayer prediction modes in SVC will be briefly introduced. Some observations are introduced in Section III to show the rate distortion cost relationship between different prediction modes. Afterwards, the mode preselection algorithms are proposed in Section IV. Simulation results are shown in Section V to demonstrate the efficiency

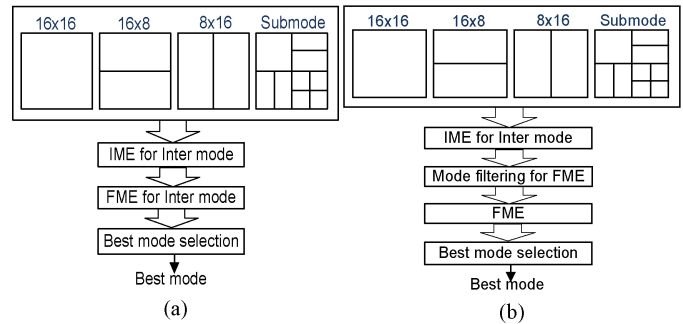


Fig. 4. (a) Mode selection process for H.264/AVC. (b) Mode preselection concept for H.264/AVC.

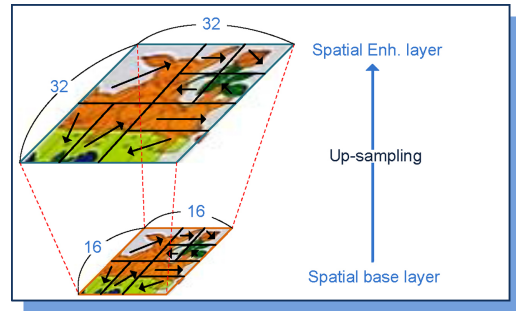


Fig. 5. Interlayer motion prediction.

of our proposed algorithms. The hardware architecture design is presented in Section VI. Finally, the conclusions are given in Section VII.

II. INTRODUCTION TO INTERLAYER PREDICTION MODES IN H.264/AVC SCALABLE EXTENSION

In addition to the inherent prediction modes in H.264/AVC, SVC also supports Interlayer motion, Interlayer residual, InterBL, and Interlayer texture prediction mode to encode the macroblocks in enhancement layers. In these Interlayer prediction modes, the base layer information is used as a reference for the prediction purpose in spatial enhancement layers to further increase the coding performance. In the following subsections, the Interlayer motion, Interlayer residual, and InterBL prediction modes are briefly described.

A. Interlayer Motion Prediction

In this prediction mode, the motion information of base layers is used as a reference for the prediction in the enhancement layer, as shown in Fig. 5, when both modes of enhancement layer and base layer are Inter prediction mode. The motion vectors of the enhancement layer are obtained by multiplying the motion vectors of corresponding block in the base layer by the frame resolution ratio between spatial layers. Furthermore, the up-sampled motion information is used to refine the search results.

B. Interlayer Residual Prediction

Fig. 6 shows the concept of Interlayer residual prediction mode. For Interlayer residual prediction, the residual data is

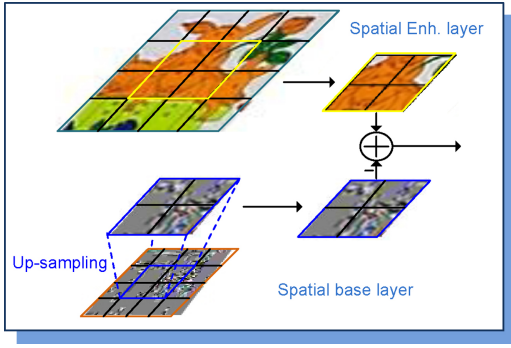


Fig. 6. Interlayer residual prediction.

up-sampled from corresponding block of the base layer by bilinear interpolation. Afterwards, the up-sampled residuals are used to predict the residuals of the current macroblock in the enhancement layer.

C. InterBL Prediction Mode

Similar to the Interlayer motion prediction mode, InterBL uses the motion information up-sampled from the base layer to predict the macroblocks of enhancement layers. However, the main difference between ILM and InterBL is that the up-sampled motion information will not be further refined in the InterBL prediction mode.

III. ANALYSIS OF RATE DISTORTION COST BETWEEN PREDICTION MODES

In this section, we conduct several analyses for the rate distortion cost of IME and FME between different prediction modes. Afterwards, several simulations are derived to confirm the observed results.

A. Relationship of Rate Distortion Cost between Different Prediction Modes

In this subsection, we analyze the relationship of rate distortion cost (RDC) of IME and FME for four different prediction modes combination. These four combinations are as follows:

- Type1* : Inter versus Interlayer motion;
- Type2* : Inter+residual versus Interlayer motion + residual;
- Type3* : Inter versus Inter+residual;
- Type4* : Interlayer motion versus Interlayer motion + residual.

In our simulation, the test conditions are listed as follows:

- 1) Reference software JSVM9.17 [5];
- 2) Two spatial layers;
- 3) Search range: ± 8 ;
- 4) Quantization parameter both in base layer and enhancement layer: 18, 28, and 38;
- 5) Group of picture: 8;
- 6) Quality scalability: Off;
- 7) Adaptive Inter-layer prediction: On.

Figs. 7–11 show the RDC relationship of IME and FME for MB (including 16×16 , 16×8 , and 8×16 block size) and submode prediction modes of different types. In these figures,

the vertical axis indicates the RDCs and the horizontal axis is the index of macroblocks. The terms of *I*, *M*, and *R* individually stand for the *Inter*, *ILM*, and *Interlayer Residual* prediction mode.

From these figures, we can observe one property, cost similarity, between IME and FME for a prediction mode. Here, we define the cost similarity as the RDCs variation between IME and FME can be determined by $[(RDC_{IME} - RDC_{FME}) / RDC_{IME}] < th$. The threshold *th* is experimentally derived and has been set to 0.01 in this paper. If the cost similarity property is satisfied, it means that the RDC of IME is very close to the RDC of FME for Inter prediction mode as an example. Similar observation can also be found for ILM prediction mode in *Type1*. That is, if IME RDC of Inter mode is sufficiently larger than that of IME RDC of ILM mode, it has a high probability that the FME RDC of Inter mode will be larger than FME RDC of ILM mode and vice versa due to the cost similarity property. From these figures, we can observe that the most of macroblocks have the cost similarity property for MB size modes. However, for the block size of submode, such property does not exist.

From the above observations, we can summarize the following selection rules. First, the mode preselection process should take the block prediction size into consideration. That is, the block sizes of 16×16 , 16×8 , and 8×16 are considered together for deriving FME mode preselection algorithm. For smaller blocks, they should be treated individually for better prediction performance as it reveals different behavior compared to block sizes of 16×16 , 16×8 , and 8×16 . Second, the macroblocks should be dealt separately depending on what property they belong to. More concretely, the macroblocks that have the cost similarity property should be processed by selection rules designed while considering the cost similarity property, and the other macroblocks have to be processed by another mode preselection algorithm without considering the cost similarity property.

To confirm the validity of our observation, several simulations are conducted by using the conditional probability of $P(A|E)$ listed as

$$P(A|E) = \frac{P(A \cap E)}{P(E)}. \quad (1)$$

Here, the probability of even *E* is defined as follows:

$$P(E) = P(RDC_{IME}(x)_{Mode+w} \leq RDC_{IME}(y)_{Mode}) | x \neq y. \quad (2)$$

And the probability of even *A* is defined as follows:

$$P(A) = P(RDC_{FME}(x)_{Mode} \leq RDC_{FME}(y)_{Mode}) | x \neq y \quad (3)$$

where $x, y \in \{I, M, R + I, R + M\}$, $Mode \in \{16 \times 8, 8 \times 16, 16 \times 8, Submode\}$, *w* is a weighting value and it can be adjusted dynamically, and RDC_{IME} and RDC_{FME} are the random variable of rate distortion cost of IME and FME, respectively.

For different prediction mode combinations, the probability of each event is individually defined as follows:

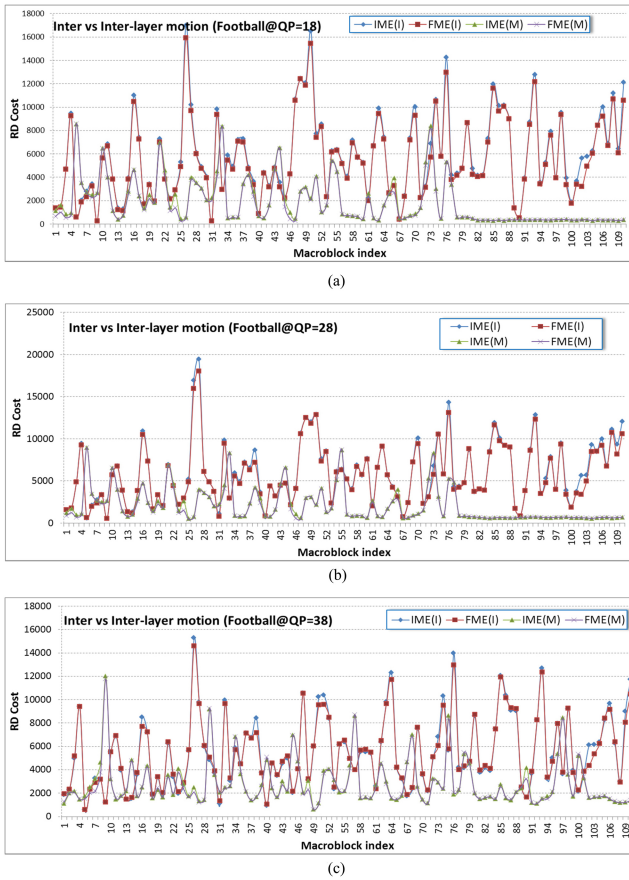


Fig. 7. Relationship between RDC s of IME and FME of Football sequence for MB prediction mode combination for *Type1*. (a) QP = 18. (b) QP = 28. (c) QP = 38.

For *Type1*

$$P(E) = P(RDC_{IME}(I)_{Mode} + w \leq RDC_{IME}(M)_{Mode}) \quad (4)$$

$$P(A) = P(RDC_{FME}(I)_{Mode} \leq RDC_{FME}(M)_{Mode}) \quad (5)$$

or

$$P(E) = P(RDC_{IME}(M)_{Mode} + w \leq RDC_{IME}(I)_{Mode}) \quad (6)$$

$$P(A) = P(RDC_{FME}(M)_{Mode} \leq RDC_{FME}(I)_{Mode}) \quad (7)$$

$$w = \begin{cases} w_1 = AvRDC, Mode\{16 \times 16, 16 \times 8, 8 \times 16\} \\ w_2 = 0, Mode \in \{Submode\} \end{cases}$$

where

$$AvRDC = \frac{1}{3} \sum_{m \in \{16 \times 16, 16 \times 8, 8 \times 16\}} |RDC_{IME}(I)_m - RDC_{IME}(M)_m|. \quad (8)$$

For *Type2*

$$P(E) = P(RDC_{IME}(R+I)_{Mode} + w \leq RDC_{IME}(R+M)_{Mode}) \quad (9)$$

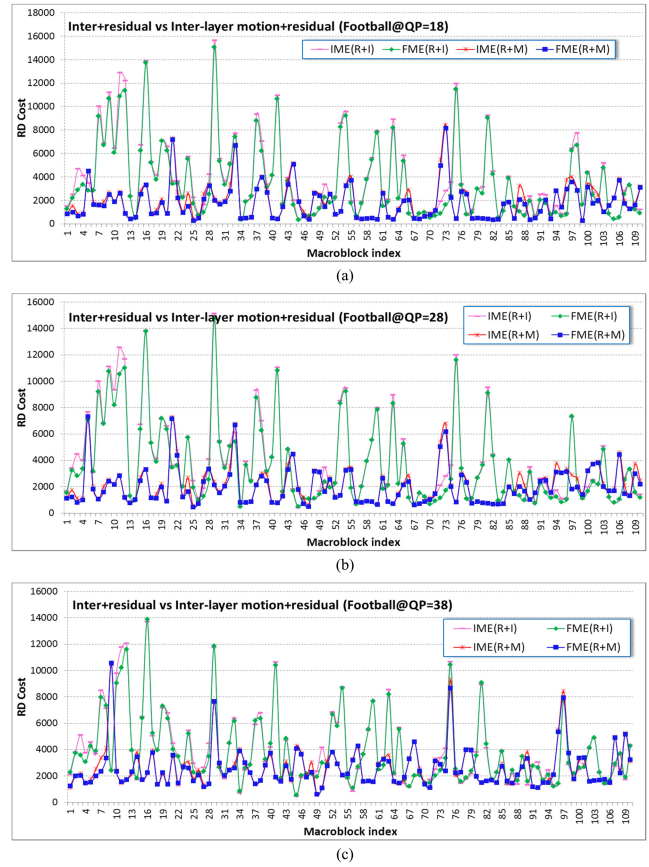


Fig. 8. Relationship between RDC s of IME and FME of Football sequence for MB prediction mode combination *Type2*. (a) QP = 18. (b) QP = 28. (c) QP = 38.

$$P(A) = P(RDC_{FME}(R+I)_{Mode} \leq RDC_{FME}(R+M)_{Mode}) \quad (10)$$

or

$$P(E) = P(RDC_{IME}(R+M)_{Mode} + w \leq RDC_{IME}(R+I)_{Mode}) \quad (11)$$

$$P(A) = P(RDC_{FME}(R+M)_{Mode} \leq RDC_{FME}(R+I)_{Mode}) \quad (12)$$

$$w = \begin{cases} w_1 = AvRDC, Mode\{16 \times 16, 16 \times 8, 8 \times 16\} \\ w_2 = 0, Mode \in \{Submode\} \end{cases}$$

where

$$AvRDC = \frac{1}{3} \sum_{m \in \{16 \times 16, 16 \times 8, 8 \times 16\}} |RDC_{IME}(R+I)_m - RDC_{IME}(R+M)_m|. \quad (13)$$

For *Type3*

$$P(E) = P(RDC_{IME}(I)_{Mode} + w \leq RDC_{IME}(R+I)_{Mode}) \quad (14)$$

$$P(A) = P(RDC_{FME}(I)_{Mode} \leq RDC_{FME}(R+I)_{Mode}) \quad (15)$$

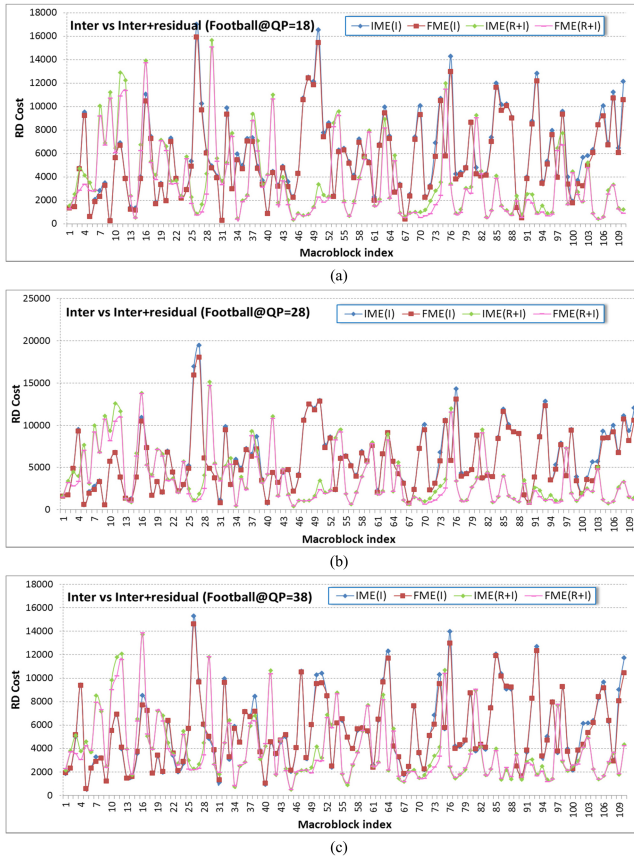


Fig. 9. Relationship between RDC s of IME and FME of Football sequence for MB prediction mode combination *Type3*. (a) QP = 18. (b) QP = 28. (c) QP = 38.

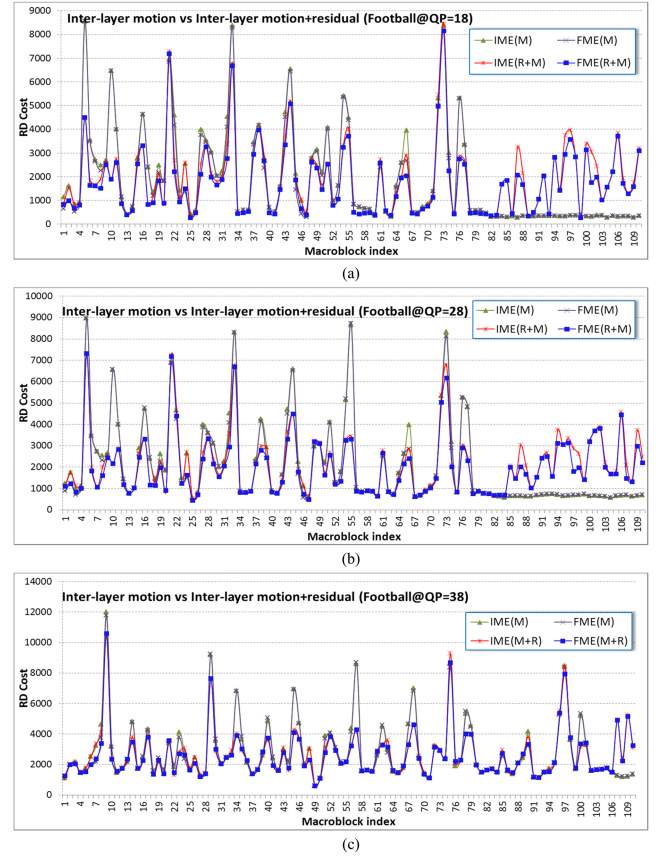


Fig. 10. Relationship between RDC s of IME and FME of Football sequence for MB prediction mode combination *Type4*. (a) QP = 18. (b) QP = 28. (c) QP = 38.

or

$$P(E) = P(RDC_{IME}(R+I)_{Mode} + w \leq RDC_{IME}(I)_{Mode}) \quad (16)$$

$$P(A) = P(RDC_{FME}(R+I)_{Mode} \leq RDC_{FME}(I)_{Mode}) \quad (17)$$

$$w = \begin{cases} w_1 = AvRDC, Mode \{16 \times 16, 16 \times 8, 8 \times 16\} \\ w_2 = 0, Mode \in \{Submode\} \end{cases}$$

where

$$AvRDC = \frac{1}{3} \sum_{m \in \{16 \times 16, 16 \times 8, 8 \times 16\}} |RDC_{IME}(I)_m - RDC_{IME}(R+I)_m|. \quad (18)$$

For Type4

$$P(E) = P(RDC_{IME}(M)_{Mode} + w \leq RDC_{IME}(R+M)_{Mode}) \quad (19)$$

$$P(A) = P(RDC_{FME}(M)_{Mode} \leq RDC_{FME}(R+M)_{Mode}) \quad (20)$$

or

$$P(E) = P(RDC_{IME}(R+M)_{Mode} + w \leq RDC_{IME}(M)_{Mode}) \quad (21)$$

$$P(A) = P(RDC_{FME}(R+M)_{Mode} \leq RDC_{FME}(M)_{Mode}) \quad (22)$$

$$w = \begin{cases} w_1 = AvRDC, Mode \{16 \times 16, 16 \times 8, 8 \times 16\} \\ w_2 = 0, Mode \in \{Submode\} \end{cases}$$

where

$$AvRDC = \frac{1}{3} \sum_{m \in \{16 \times 16, 16 \times 8, 8 \times 16\}} |RDC_{IME}(M)_m - RDC_{IME}(R+M)_m| \quad (23)$$

B. Statistical Results

The statistical results are shown in Table I–IV. For *Type1*, *Type2*, *Type3*, and *Type4* prediction mode combination, the conditional probability can achieve 82.39%, 72.27%, 96.21%, and 95.93% on an average, respectively. Therefore, from these tables, we can make sure that our observations work well and the conditions listed above can be used to derive our FME mode preselection algorithm to skip the potentially ignorable prediction modes, and thus achieve computational complexity savings.

IV. PROPOSED FME MODE PRESELECTION ALGORITHM

Fig. 12 shows the concept of proposed FME mode preselection for SVC. In our proposed FME mode preselection algorithm, IME is executed for the prediction modes of Inter, *Inter+residual*, *Interlayer motion*, and *Interlayer motion+residual* first. Afterwards, the proposed FME mode

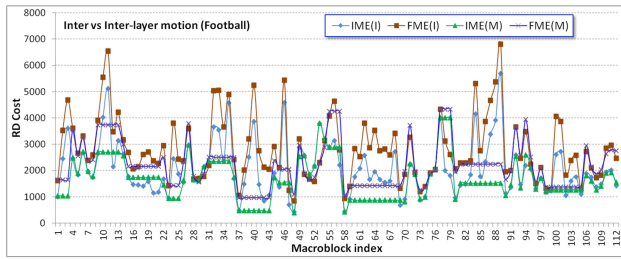


Fig.11(a)

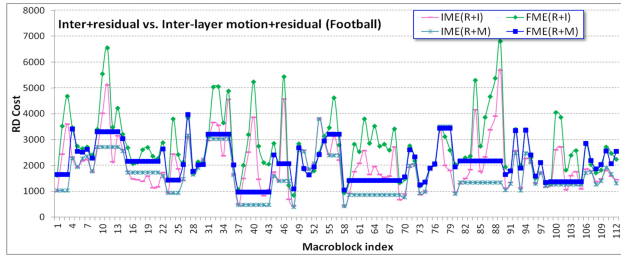


Fig.11(b)

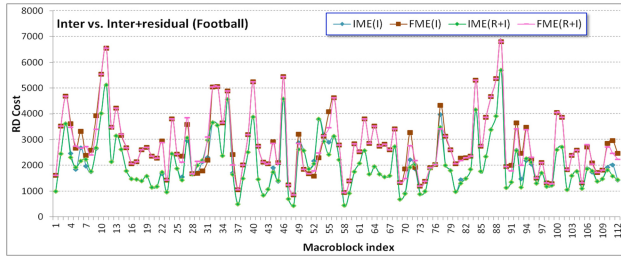


Fig.11(c)

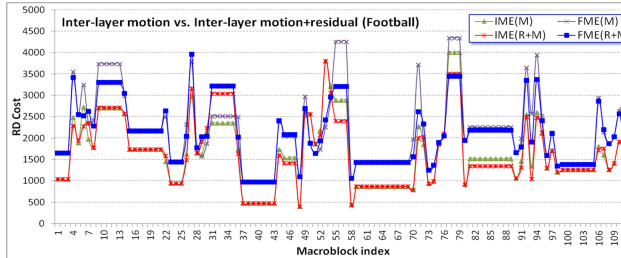


Fig.11(d)

Fig. 11. Relationship between $RDCs$ of IME and FME of Football sequence for submode prediction mode (a) *Type1*, (b) *Type2* (c) *Type3*, and (d) *Type4*.

preselection algorithm is applied for all prediction modes coming from the results of IME to skip the potentially ignorable prediction modes before entering FME operation. Once the candidate prediction modes have been decided by the proposed FME mode preselection algorithm, the selected candidate modes will be fed into FME module to choose the best prediction mode.

Fig. 13 shows the flowchart of our proposed mode preselection algorithm that includes four types of mode preselection algorithm. In this flowchart, the candidate set of prediction modes Φ is defined as follows:

$$\Phi = \{\Phi_{ij} | i \in \{Inter, ILM, InterR, ILMR\}, j \in \{16 \times 16, 16 \times 8, 8 \times 16, Submode\}\} \quad (24)$$

Fig. 14 shows the detailed flowchart of proposed FME mode preselection algorithm of *Type1*. In this flowchart, the weightings of w_1 and w_2 are computed first. Afterwards, IME

TABLE I
STATISTICAL RESULTS OF *Type1*

Sequences	16×16	16×8	8×16	Submode
<i>Akiyo</i>	96.75	98.22	96.39	96.02
<i>Table</i>	76.48	79.59	75.63	82.10
<i>News</i>	95.45	97.93	95.10	94.45
<i>Tempete</i>	78.53	82.68	76.43	74.46
<i>Football</i>	71.02	85.31	70.18	72.31
<i>Foreman</i>	74.11	81.55	73.05	74.84
<i>M&D</i>	95.58	97.10	94.75	95.00
<i>Soccer</i>	69.27	76.77	69.10	75.72
<i>Stefan</i>	71.25	80.43	71.31	70.96
Average	80.94	86.62	80.22	81.76

TABLE II
STATISTICAL RESULTS OF *Type2*

Sequences	16×16	16×8	8×16	Submode
<i>Akiyo</i>	83.31	99.31	86.99	94.78
<i>Table</i>	62.24	81.27	64.80	58.01
<i>News</i>	70.84	93.03	66.29	64.67
<i>Tempete</i>	75.56	79.32	73.23	58.73
<i>Football</i>	63.35	86.66	66.12	61.26
<i>Foreman</i>	64.77	72.21	73.41	72.59
<i>M&D</i>	76.66	81.71	71.01	64.02
<i>Soccer</i>	56.74	76.30	61.73	60.40
<i>Stefan</i>	67.41	79.23	65.71	67.94
Average	68.99	83.23	69.92	66.93

TABLE III
STATISTICAL RESULTS OF *Type3*

Sequences	16×16	16×8	8×16	Submode
<i>Akiyo</i>	93.55	94.46	91.60	88.37
<i>Table</i>	99.47	98.97	98.72	96.50
<i>News</i>	99.77	99.88	99.96	98.24
<i>Tempete</i>	95.20	93.96	92.73	87.57
<i>Football</i>	99.63	99.25	98.94	92.41
<i>Foreman</i>	99.25	99.35	99.59	96.93
<i>M&D</i>	99.18	98.33	97.30	74.39
<i>Soccer</i>	99.63	98.97	99.04	96.91
<i>Stefan</i>	99.12	96.88	96.88	92.56
Average	98.31	97.78	97.20	91.54

TABLE IV
STATISTICAL RESULTS OF *Type4*

Sequences	16×16	16×8	8×16	Submode
<i>Akiyo</i>	89.35	100.00	93.62	88.07
<i>Table</i>	99.08	99.10	99.14	95.59
<i>News</i>	99.83	99.94	99.77	98.14
<i>Tempete</i>	94.63	93.74	92.69	89.39
<i>Football</i>	99.21	97.81	97.46	90.05
<i>Foreman</i>	99.70	99.11	99.17	93.22
<i>M&D</i>	95.79	99.88	97.21	78.01
<i>Soccer</i>	99.50	99.31	99.35	96.52
<i>Stefan</i>	97.51	95.29	95.94	91.46
Average	97.18	98.24	97.15	91.16

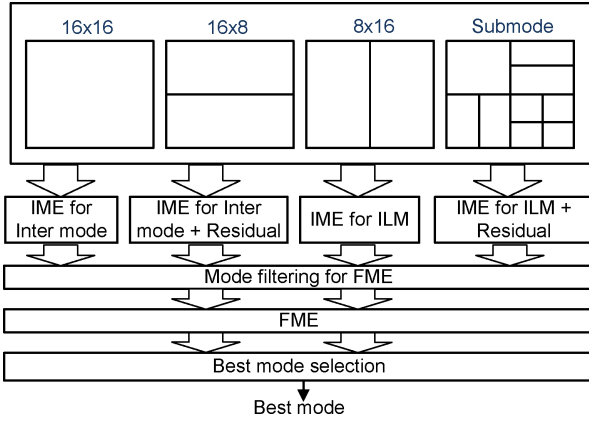


Fig. 12. FME mode preselection for SVC.

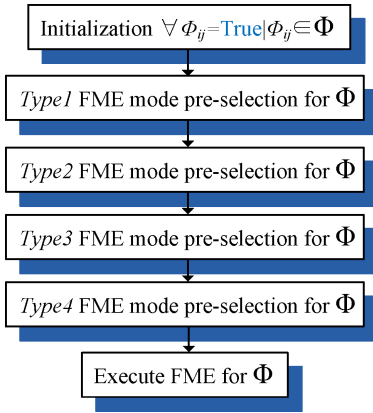


Fig. 13. Flowchart of proposed FME mode preselection algorithm for SVC.

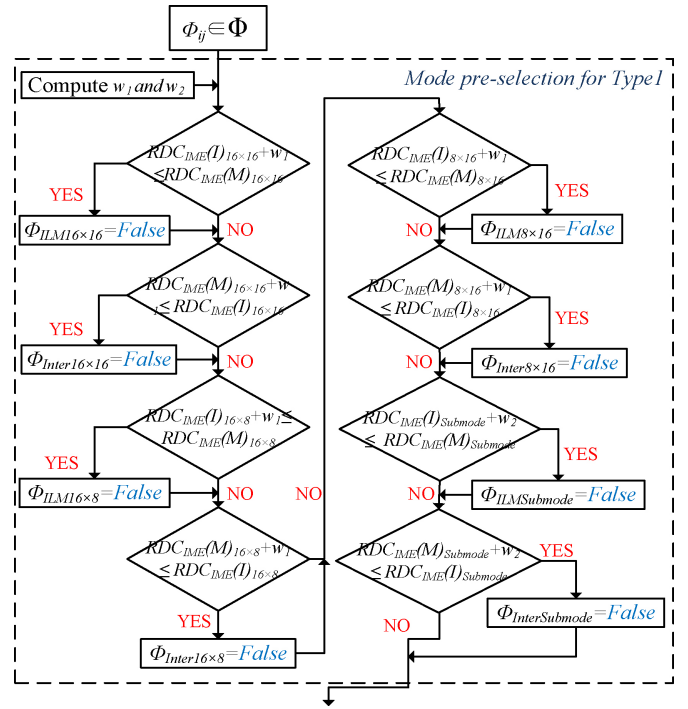
TABLE V
SIMULATION SETTINGS

Reference software		JSVM9.17 [5]
QP for spatial base layer		18, 28, 33, 38
QP for Spatial enhancement layer		12, 22, 27, 32
Frame size in spatial base layer		QCIF and 540P
Frame size in spatial enhancement layer		CIF and 1080P
Frames to be encoded		300 and 150
Frame rate		30 and 15
Adaptive inter-layer prediction		ON
Search range size		± 8
GOP		8
Test sequences	QCIF and CIF	Akiyo, Dancer, Coastguard, Table tennis, Tempete, Football, Foreman, MD, Mobile, News, Soccer, Stefan
	540P and 1080P	Blue_sky, Pedestrian, Riverbed, Station2, Sunflower, Tractor

rate distortion cost relationship between *Inter* and *Interlayer motion* is compared for all block size one by one to skip the potentially ignorable prediction modes. Once the *Type1* mode preselection algorithm has been done, the candidate set of Φ will be fed into the next mode preselection algorithm to further preselect possible modes. Fig. 15 exhibits the detailed flowchart of proposed FME mode preselection algorithm for *Type2*. The flowchart of *Type2* mode preselection algorithm

TABLE VI
BD-PSNR AND BD-RATE COMPARISONS FOR PROPOSED ALGORITHM
SUBJECT TO FULL MODE FME

Resolution	Sequences	BD-PSNR (dB)	BD-rate (%)
BL: QCIF EL: CIF	Akiyo	-0.025	+0.774
	Coastguard	-0.030	+0.469
	Dancer	-0.056	+0.754
	Football	-0.041	+0.733
	Foreman	-0.052	+1.619
	Mobile	-0.037	+0.636
	MD	-0.031	+1.141
	News	+0.056	-0.783
	Table tennis	-0.045	+0.920
	Tempete	-0.032	+0.586
	Stefan	-0.051	+1.112
	Soccer	-0.069	+1.364
Average		-0.034	+0.777
BL: 540P EL: 1080P	Blue_sky	-0.051	+1.216
	Pedestrian	-0.017	+0.038
	Riverbed	-0.006	+0.006
	Station2	-0.043	+1.520
	Sunflower	-0.067	+2.547
	Tractor	-0.035	+0.156
Average		-0.037	+0.914

Fig. 14. Detailed flowchart of proposed *Type1* mode preselection algorithm.

is very similar to the mode preselection algorithm of *Type1* except that the rate distortion cost relationship is compared between *Inter+residual* and *Interlayer motion+residual*.

Figs. 16 and 17 show the detailed flowcharts of proposed *Type3* and *Type4* mode preselection algorithm, respectively, which are similar to those of *Type1* and *Type2* mode preselection algorithms. Extra processes are highlighted by bold line for clarity. The extra processes here are to avoid unnecessary decision operations since some candidate prediction

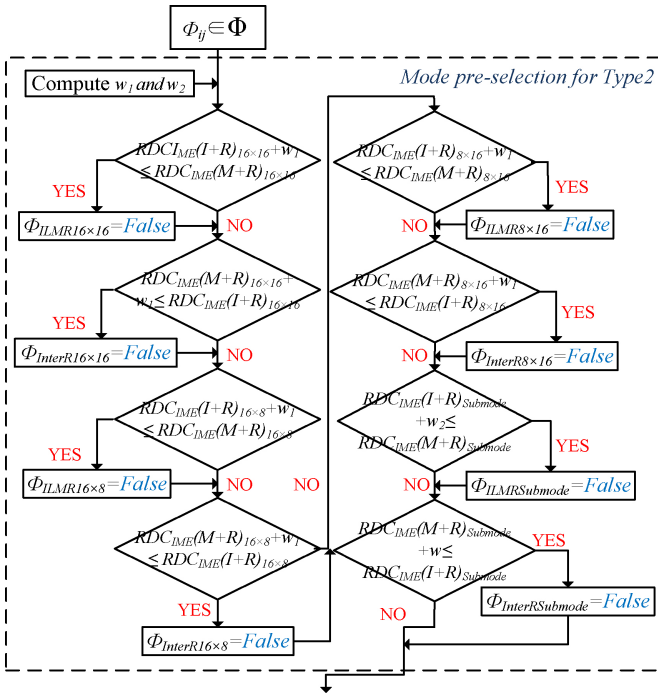


Fig. 15. Detailed flowchart of proposed Type2 mode preselection algorithm.

modes might have been already disabled in the previous Type1 and Type2 mode preselection algorithm.

V. SIMULATION RESULTS

In this section, several simulation results are shown to demonstrate the performance of our proposed FME mode preselection algorithm. The simulation settings are summarized in V.

Table VI shows the BD-PSNR and BD-rate comparisons for our proposed FME mode preselection algorithm and the full mode FME. In this table, the quantization parameter values of 12, 22, 27, and 32 are adopted to derive the results. It should be mentioned that since the only highest quality (including spatial, temporal, SNR) layers would be decoded from the SVC bitstream, we only show the QP values for highest spatial layers. For QCIF and CIF case, the average BD-PSNR degradation and BD-rate increasing is 0.034dB and 0.0777%, respectively. In addition, for the 540P and 1080P case, the average BD-PSNR degradation and BD-rate increasing is 0.037dB and 0.914%. From this table, it is obvious that our proposed FME mode preselection algorithm results in negligible rate distortion performance loss when compared to the full mode FME. Table VII further shows the detailed PSNR degradation and bitrate increasing for different QP values. On average, our proposed algorithm only results in 0.005dB PSNR degradation and 0.89% bitrate increasing.

Table VIII shows the mode reductions of our proposed algorithm. On average, our proposed algorithm can achieve 65.97% mode reduction whatever the quantization parameter is.

Table IX tabulates the execute time reduction of our proposed FME mode preselection algorithm compared to full

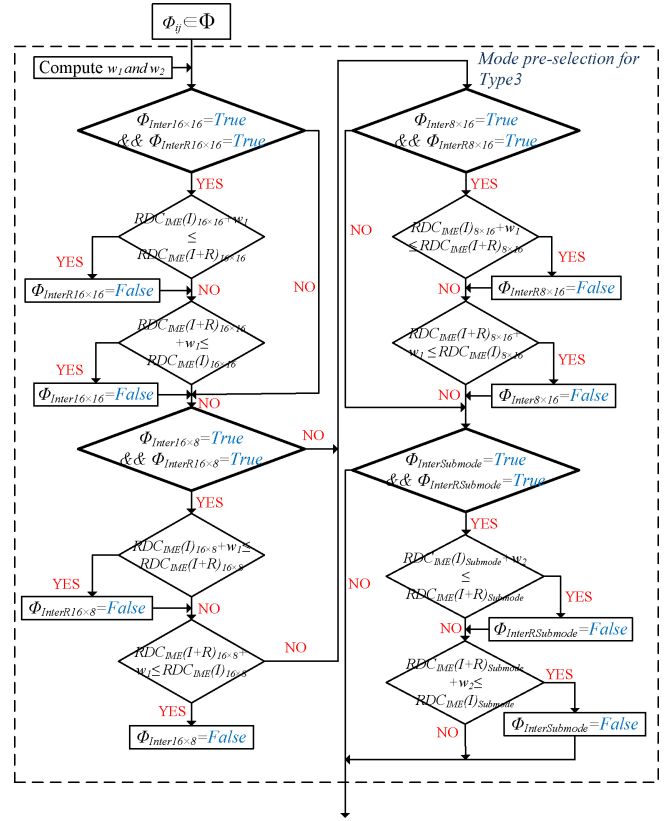


Fig. 16. Detailed flowchart of proposed Type3 mode preselection algorithm.

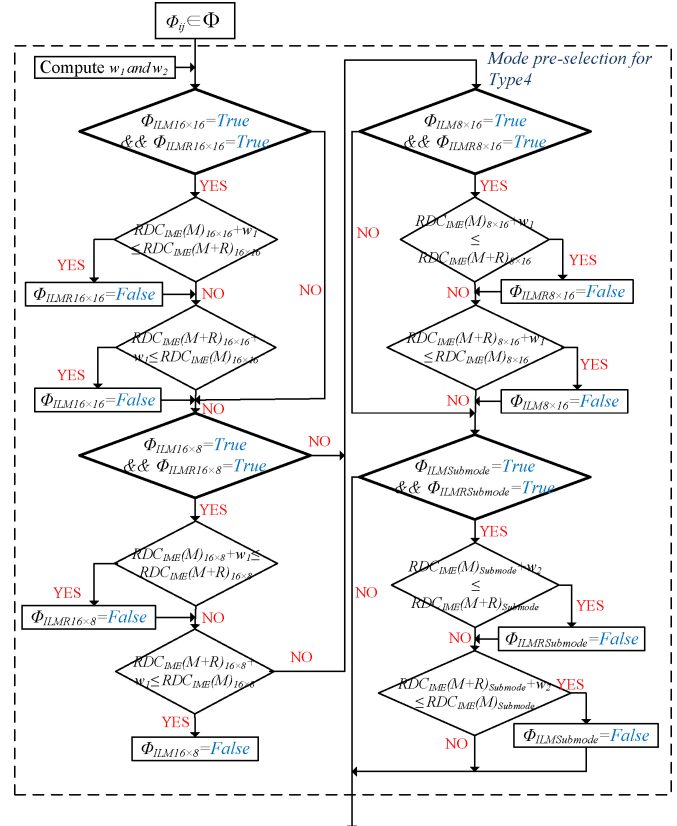


Fig. 17. Detailed flowchart of proposed Type4 mode preselection algorithm.

TABLE VII
RATE DISTORTION PERFORMANCE COMPARISONS FOR PROPOSED ALGORITHM

	PSNR degradation (dB)			Bitrate increase (%)		
	QP _{BL} = 18 QP _{EL} = 12	QP _{BL} = 28 QP _{EL} = 22	QP _{BL} = 38 QP _{EL} = 32	QP _{BL} = 18 QP _{EL} = 12	QP _{BL} = 28 QP _{EL} = 22	QP _{BL} = 38 QP _{EL} = 32
Akiyo	0.00	0.00	0.01	0.28	0.94	1.26
Dancer	0.01	0.00	-0.01	0.47	0.68	1.34
Coastguard	0.00	-0.01	-0.01	0.22	0.37	0.66
Table	0.00	0.00	0.00	0.41	0.92	1.72
Tempete	0.00	0.00	-0.01	0.31	0.55	1.07
Football	0.00	0.00	0.00	0.34	0.45	1.58
Foreman	0.00	-0.01	-0.02	0.51	1.17	1.90
MD	0.00	0.00	0.02	0.42	1.25	2.22
Mobile	0.00	-0.01	0.00	0.33	0.40	0.86
News	-0.01	-0.01	-0.02	0.23	0.65	1.06
Soccer	-0.01	0.00	-0.01	0.33	1.67	0.91
Stefan	0.00	-0.01	-0.01	0.35	1.14	0.59
Blue_sky	0.01	-0.01	-0.03	0.79	0.67	1.27
Pedestrian	0.01	-0.01	-0.01	0.28	0.49	0.90
Riverbed	0.01	0.01	0.01	0.05	0.17	0.79
Station2	0.02	-0.01	-0.02	0.43	1.04	1.92
Sunflower	0.02	-0.02	0.00	0.56	1.16	2.72
Tractor	0.01	-0.02	-0.03	0.50	0.30	3.17
Average	0.003	-0.006	-0.008	0.38	0.78	1.44

TABLE VIII
MODE REDUCTIONS FOR PROPOSED ALGORITHM SUBJECT TO FULL MODE FME

(%)		QP _{BL} = 18 QP _{EL} =12	QP _{BL} = 28 QP _{EL} =22	QP _{BL} = 33 QP _{EL} =27	QP _{BL} = 38 QP _{EL} =32
BL:QCIF EL:CIF	Akiyo	-75.00	-75.00	-75.00	-75.00
	Dancer	-75.00	-75.00	-75.00	-75.00
	Coastguard	-62.50	-62.50	-62.50	-62.50
	Table	-62.50	-62.50	-62.50	-62.50
	Tempete	-75.00	-75.00	-75.00	-75.00
	Football	-62.50	-62.50	-62.50	-62.50
	Foreman	-62.50	-62.50	-62.50	-62.50
	MD	-75.00	-75.00	-75.00	-75.00
	Mobile	-62.50	-62.50	-62.50	-62.50
	News	-75.00	-75.00	-75.00	-75.00
	Soccer	-62.50	-62.50	-62.50	-62.50
Stefan	-62.50	-62.50	-62.50	-62.50	
BL:540P EL:1080P	Blue_sky	-62.50	-62.50	-62.50	-62.50
	Pedestrian	-62.50	-62.50	-62.50	-62.50
	Riverbed	-62.50	-62.50	-62.50	-62.50
	Station2	-62.50	-62.50	-62.50	-62.50
	Sunflower	-62.50	-62.50	-62.50	-62.50
	Tractor	-62.50	-62.50	-62.50	-62.50
Average		-65.97	-65.97	-65.97	-65.97

mode FME. Our simulation was running on the Microsoft Windows Server 2003 operating system with Inter Xeon 2.5 GHz CPU and 4GB RAM. From this table, we can observe that our proposed algorithm can achieve about 79% execution time reduction on an average. However, compared to the mode reduction shown in Table VIII, it can be found that the execution time reduction is higher than that of the mode reduction due to different execution time of each mode. More precisely, the reduced number of submodes in our proposed algorithm can significantly save the overall execution time with negligible efforts on calculating the preselection rules.

Table X shows the performance comparison of different algorithms. For BD-PSNR and BD-rate comparison, it can be found that our proposed algorithm outperforms [4] and

[19]. However, although our proposed algorithm has slight BD-PSNR decrease and BD-rate increase compared to [18], the complexity saving of our proposed algorithm is much higher than [18].

VI. HARDWARE ARCHITECTURE DESIGN

Fig. 18 reveals the hardware architecture design of our proposed FME more preselection algorithm. When designing a modern video encoder system, IME and FME are usually separated into two distinct pipeline stages, as shown Fig. 18(a), due to the reason of balancing the computational loads for each stage [7], [20]-[22]. Therefore, our proposed FME mode preselection algorithm is arranged at the same stage of IME due to the hardware cost consideration.

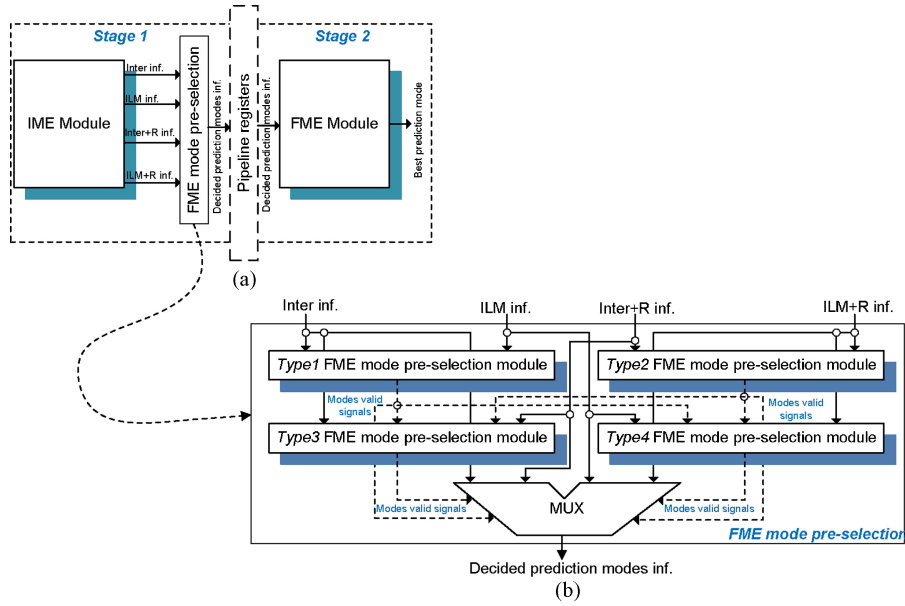


Fig. 18. Hardware architecture of our proposed FME mode preselection algorithm. (a) Combination with IME module. (b) Detailed architecture of FME mode preselection.

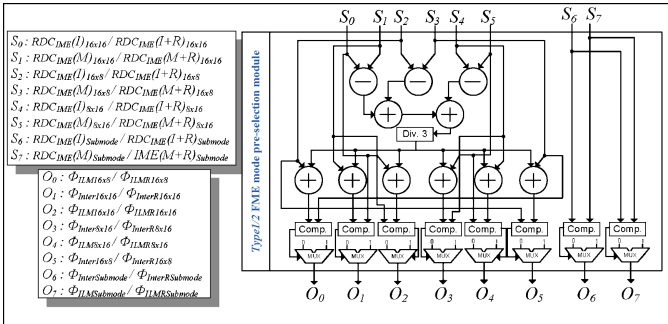


Fig. 19. Detailed hardware architecture of Type1 and Type2 FME mode preselection algorithm.

Fig. 18(b) shows the overall hardware architecture of our proposed FME mode preselection algorithm that is mainly composed by four major modules, Type1, Type2, Type3, and Type4 FME mode preselection module. The four major modules corresponding to four major mode preselection algorithms are mentioned in Section III. For Type1 and Type2 FME mode preselection modules, the mode preselection rules are applied to valid the possible modes for the following process. Once the Type1 and Type2 modules have finished their mode preselection tasks, some mode validation signals will be generated and passed to the other modules. Similarly, the Type3 and Type4 modules have the same operations like Type1 and Type2 modules do, but they further take the mode validation signals received from Type1 and Type2 modules into consideration. Once all modules have finished their mode preselection operations, all generated mode validation signals will be sent to a multiplexer to select the mode prediction information for output.

The detailed hardware architecture design of proposed Type1/Type2 and Type3/Type4 FME mode preselection algorithm is shown in Fig. 19 and Fig. 20, respectively.

From the detailed hardware architecture figures shown in Fig. 19 and Fig. 20, we observed that our design needs a division to calculate the weighting of w_1 . In the hardware design perspective, the operations of multiplication and division are usually avoided to reduce the hardware implementation costs. Therefore, we use a mathematical way to approximate the result of division. Mathematically, our division operations can be expressed as

$$x = \frac{s}{3} \approx \sum_{i=1}^n \frac{s}{2^{2i}} \approx \frac{s}{2^2} + \frac{s}{2^4} + \frac{s}{2^6} + \dots + \frac{s}{2^{2n}} \quad (25)$$

where x and s stand for the computed result and the dividend, respectively. From this equation, it can be seen that although the division operation still existed in the equation, the division operations can be easily implemented by the right shift operations since the divisors are all the square of two. As a result, the hardware cost of division can be saved. However, we can further observe that a variable n existed in (25). This variable n is used to define the precision of the computed result. In other words, the larger the variable n is, the higher the computed precision can be achieved. However, the larger n also results in the increasing of the computations and hardware costs. Therefore, the n is set to 10 in our design since $n=10$ is sufficient. When integrating our proposed FME mode preselection design into SVC hardware implementation, the number of required FME processing blocks in 4×4 processing unit is 138 and 92 for worse and best case, respectively. Compared to the 366 required FME processing blocks without incorporating our proposed FME mode preselection algorithm, our proposal can reduce the computation complexity of FME significantly. Implementation results show that our proposed FME mode preselection algorithm only consumes 9k gate counts when synthesized by 90nm technology.

TABLE IX
EXECUTION TIME REDUCTIONS FOR PROPOSED ALGORITHM SUBJECT TO FULL MODE FME

(%)		QP _{BL} = 18 QP _{EL} =12	QP _{BL} = 28 QP _{EL} =22	QP _{BL} = 33 QP _{EL} =27	QP _{BL} = 38 QP _{EL} =32
BL:QCIF EL: CIF	Akiyo	79.97	79.99	79.99	79.98
	Dancer	79.35	79.93	79.96	79.94
	Coastguard	78.93	79.85	79.96	79.98
	Table	79.43	79.92	79.99	79.99
	Tempete	79.28	79.98	79.97	79.98
	Football	77.32	79.63	79.89	79.95
	Foreman	79.63	79.96	79.95	79.94
	MD	79.92	79.97	79.98	79.98
	Mobile	78.43	79.97	79.97	79.97
	News	79.78	79.97	79.99	79.97
	Soccer	79.29	79.96	79.95	79.99
EL:1080P BL:540P	Stefan	78.38	79.82	79.96	79.98
	Blue_sky	79.82	79.89	79.95	79.92
	Pedestrian	79.87	79.95	79.94	79.85
	Riverbed	79.95	79.98	79.97	79.99
	Station2	79.85	79.97	79.35	79.93
	Sunflower	79.92	79.99	78.93	79.85
	Tractor	79.98	79.97	79.95	79.94
Average		79.39	79.93	79.87	79.95

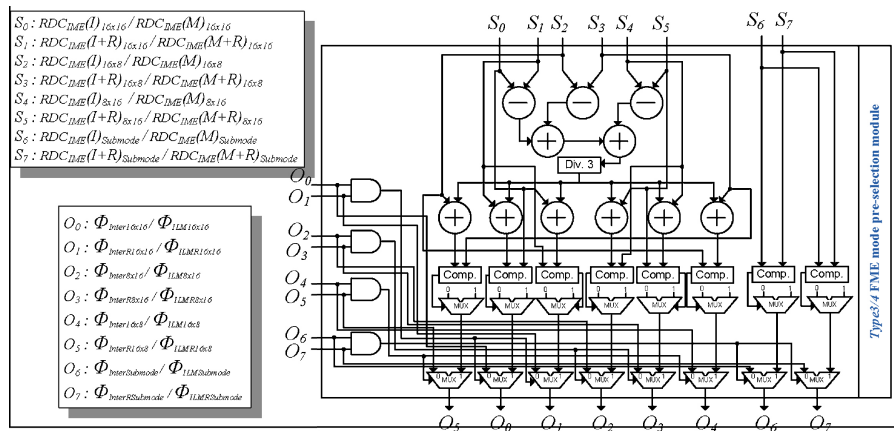


Fig. 20. Detailed hardware architecture of Type3 and Type4 FME mode preselection algorithm.

TABLE X
BD-PSNR AND BD-RATE COMPARISONS FOR DIFFERENT ALGORITHMS

BD-PSNR (dB)			
Proposed	[4]	[18]	[19]
-0.037	-0.114	-0.034	-0.05
BD-rate (%)			
Proposed	[4]	[18]	[19]
+0.914	+1.14	+0.37	+1.64
Complexity saving (%)			
Proposed	[4]	[18]	[19]
79.79*	64**	44.25*	79**

*: execution time savings
 **: clock cycle savings

VII. CONCLUSION

In this paper, we proposed an FME mode preselection algorithm to skip potentially ignorable prediction modes before FME to lighten the computation complexity. The rate distortion cost relationship between different prediction modes

was analyzed first to explore the possibility of ignoring some prediction modes. Afterward, several statistical results were conducted to confirm our observation and the results proved the high possibility that some prediction modes could be skipped before FME by using observed properties. Based on the observations and statistical results, we proposed several FME mode preselection algorithms to preselect modes before entering FME operation. Simulations results showed that our proposed FME mode preselection algorithm only results in 0.036dB and 0.496% BD-PSNR degradation and BD-rate increase, respectively. On average, our proposed algorithm could achieve 65.97% mode reduction. The resulted hardware only cost 9K gate count.

REFERENCES

[1] *Advanced Video Coding for Generic Audiovisual Services*, ITU-T Rec. H.264, Mar. 2010.
 [2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, Jul. 2003.

- [3] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of scalable extension of H.264/MPEG-4 AVC video coding standard," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 17, no. 9, pp. 103–112, Sep. 2007.
- [4] T.-C. Chen, Y.-W. Huang, and L.-G. Chen, "Fully utilized and reusable architecture for fractional motion estimation of H.264/AVC," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2004, pp. V-9–V-12.
- [5] *JSVM Software Version JSVM 9.17*, ITU-T and I. JTC1, 2008.
- [6] T.-C. Wang, Y.-W. Huang, H.-C. Fang, and L.-G. Chen, "Performance analysis of hardware oriented algorithm modifications in H.264," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 2, Apr. 2003, pp. 493–496.
- [7] Y.-K. Lin, D.-W. Li, C.-C. Lin, T.-Y. Kuo, S.-J. Wu, W.-C. Tai, W.-C. Chang, and T.-S. Chang, "A 242mW, 10 mm² 1080p H.264/AVC high profile encoder chip," in *Proc. IEEE Int. Solid State Circuits Conf.*, Feb. 2008, pp. 314–315.
- [8] H. Nisar and T.-S. Choi, "Fast and efficient fractional pixel motion estimation for H.264/AVC video coding," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp.1561–1564.
- [9] C.-Y. Kao, C.-L. Wu, and Y.-L. Lin, "A high-performance three-engine architecture for H.264/AVC fractional motion estimation," *IEEE Trans. Very Large Scale Integr. Sys.*, vol. 18, no. 4, pp. 662–666, Apr. 2010.
- [10] Y.-J. Wang, C.-C. Cheng, and T.-S. Chang, "A fast algorithm and its VLSI architecture for fractional motion estimation for H.264/MPEG-4/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 5, pp. 578–583, May 2007.
- [11] G. Kim, J. Kim, and C.-M. Kyung, "A low cost single-pass fractional motion estimation architecture using bit clipping for H.264 video codec," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2010, pp. 661–662.
- [12] C.-C. Yang, K.-J. Tan, Y.-C. Yang, and J.-I. Guo, "Low complexity fractional motion estimation with adaptive mode selection for H.264/AVC," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2010, pp. 673–678.
- [13] C.-C. Lin, Y.-K. Lin, and T.-S. Chang, "A fast algorithm and its architecture for motion estimation in MPEG-4 AVC/H.264," in *Proc. Asia Pacific Conf. Circuits Syst.*, Dec. 2006, pp. 1250–1253.
- [14] C. H. Yeh, K. J. Fan, M. J. Chen, and G. L. Li, "Fast mode decision algorithm for scalable video coding using Bayesian theorem detection and Markov process," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 4, pp. 563–574, Apr. 2010.
- [15] P. C. Wang, G. L. Li, S. F. Huang, M. J. Chen, and S. C. Lin, "Efficient mode decision algorithm based on spatial, temporal and inter-layer rate distortion correlation coefficients for scalable video coding," *ETRI J.*, vol. 32, no. 4, pp. 577–587, Aug. 2010.
- [16] H.-C. Lin, W.-H. Peng, and H.-M. Hang, "Fast context-adaptive mode decision algorithm for scalable video coding with combined coarse-grain quality scalable (CGS) and temporal scalability," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 5, pp. 732–748, May 2010.
- [17] B.-G. Kim, G.-S. Hong, and K.-W. Rim, "Fast mode decision algorithm for inter-frame coding in H.264/AVC extended scalable video coding," *Int. J. Soft Comput.*, vol. 6, no. 4, pp. 102–110, 2011.
- [18] Y. Chun, F. Min, and Z. Yan, "Fast mode decision algorithm for enhancement layer of spatial and CGS scalable video coding," in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jul. 2011, pp. 1–5.
- [19] K. Lee, C. E. Rhee, H.-J. Lee, and J. W. Kang, "Memory and computation efficient hardware design for a 3 spatial and temporal layers SVC encoder," *IEEE Trans. Consumer Electron.*, vol. 57, no. 4, pp. 1921–1928, Nov. 2011.
- [20] L.-F. ding, W.-Y. Chen, P.-K. Tsung, T.-D. Chuang, H.-K. Chiu, Y.-H. Chen, P.-H. Hsiao, S.-Y. Chien, T.-C. Chen, P.-C. Lin, C.-Y. Chang, and L.-G. Chen, "A 212MPixels/s 4096x2160p multiview video encoder chip for 3D/quad HDTV applications," in *Proc. IEEE Int. Solid State Circuits Conf.*, Feb. 2009, pp. 153–155.
- [21] H.-C. Chang, J.-W. Chen, C.-L. Su, Y.-C. Yang, Y. Li, C.-H. Chang, Z.-M. Chen, W.-S. Yang, C.-C. Lin, C.-W. Chen, J.-S. Wang, J.-I. Guo, "A 7mW-to-183mW dynamic quality-scalable H.264 video encoder chip," in *Proc. IEEE Int. Solid State Circuits Conf.*, Feb. 2007, pp. 279–281.
- [22] Y.-W. Huang, T.-C. chen, C.-H. Tsai, C.-Y. Chen, T.-W. chen, C.-S. Chen, C.-F. Shen, S.-Y. Ma, T.-C. Wang, B.-Y. Hsieh, H.-C. Fang, and L.-G. Chen, "A 1.3TOPS H.264/AVC single-chip encoder for HDTV applications," in *Proc. IEEE Int. Solid State Circuits Conf.*, Feb. 2005, pp. 128–588.



Gwo-Long Li received the B.S. degree from the Department of Computer Science and Information Engineering, Shu-Te University, Kaohsiung, Taiwan, in 2004, the M.S. degree from the Department of Electrical Engineering, National Dong-Hwa University, Hualien, Taiwan, in 2006, and the Ph.D. degree from the Department of Electronics Engineering, National Chiao-Tung University, Hsinchu, Taiwan, in 2011.

He is currently an Engineer at the Industrial Technology Research Institute, Hsinchu. His current research interests include the video signal processing and its very large scale integration architecture design.

Dr. Li was a recipient of the Excellent Master Thesis Award from the Institute of Information and Computer Machinery in 2006.



Tian-Sheuan Chang (S'93–M'06–SM'07) received the B.S., M.S., and Ph.D. degrees in electronic engineering from National Chiao-Tung University (NCTU), Hsinchu, Taiwan, in 1993, 1995, and 1999, respectively.

He is currently a Professor with the Department of Electronics Engineering, NCTU, Hsinchu. From 2000 to 2004, he was a Deputy Manager with Global Unichip Corporation, Hsinchu. His current research interests include (silicon) intellectual property and system-on-a-chip design, very large-scale integration signal processing, and computer architecture.