An Efficient Mode Decision Algorithm Based on Dynamic Grouping and Adaptive Adjustment for H.264/AVC

Bin Zhan, Baochun Hou and Reza Sotudeh

School of Electronic, Communications and Electrical Engineering
University of Hertfordshire
Hatfield, Hertfordshire, AL10 9AB, United Kingdom
E-mail: {b.zhan, b.hou, r.sotudeh}@herts.ac.uk

Abstract- The rate distortion optimization (RDO) enabled mode decision (MD) is one of the most important techniques introduced by H.264/AVC. By adopting the exhaustive calculation of rate distortion, the optimal MD enhances the video encoding quality. However, the computational complexity is significantly increased, which is a key challenge for real-time and low power consumption applications. This paper presents a new fast MD algorithm for highly efficient H.264/AVC encoder. The proposed algorithm employs a dynamic group of candidate inter/intra modes to reduce the computational cost. In order to minimize the performance loss incurred by improper mode selection for the previously encoded frames, an adaptive adjustment scheme based on the undulation of bitrate and PSNR is suggested. Experimental results show that the proposed algorithm reduces the encoding time by 35% on average, and the loss of PSNR is usually limited in 0.1 dB with less than 1% increase of bitrate.

Keywords- fast mode decision, H.264/AVC, rate distortion optimization, video coding

I. Introduction

H.264/AVC [1], the latest video coding standard jointly developed by ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG), introduces a number of distinct features compared with the former video coding standards to notably improve the compression performance in terms of encoding efficiency [2]. However, the high efficiency is achieved at the expense of significantly increased computational complexity. One of the most important techniques for H.264/AVC is variable block sizes based mode decision (MD) with the rate distortion optimization (RDO). In contrast to the traditionally unique 16×16 macroblock, seven block sizes are supported in the standard to more accurately predict the details in a video frame and reduce the bitrate with improved video quality. RDO enabled MD is done by minimizing the Lagrangian cost function as described in (1):

$$J(s,c,MODE/QP,\lambda_{MODE}) =$$
$$SSD(s,c,MODE/QP) + \lambda_{MODE} \cdot R(s,c,MODE/QP) \quad (1)$$

where *MODE* is one of the candidate modes. *QP* stands for the quantization parameter. λ_{MODE} is a Lagrangian multiplier for MD as defined in (2), *SSD* gives the sum of the squared differences between *s* (original block) and *c* (reconstructed

block), *R(s, c, MODE/QP)* is contributed by the number of bits based on the *MODE* and *QP*.

$$\lambda_{MODE} = 0.85 \times 2^{(QP-12)/3} \quad (2)$$

The above exhaustive computation of rate and distortion of all modes drastically increases the encoding time and the power consumption of encoder. It is also one of the major challenges for real-time and low power applications such as video conferences and mobile video communications. Several fast MD algorithms have been developed in recent years to reduce the complexity of computations. In [3], Grecos and Yang proposed a fast inter MD scheme which exploits neighbourhood information and a set of constraints to enhance the Skip MD and other MD separately. Pan et al. introduced an approach to reduce the intra candidate modes according to a pre-established local edge direction histogram in [4]. J. F. Wang et al. in [5] used the dominant edge strength to estimate the possible directions to improve the intra MD efficiency. H. Wang et al. optimized the MD by early terminating the unnecessary modes according to the thresholds based prediction in [6]. These algorithms reduced the computational cost incurred by conventional RDO enabled MD, however a large amount of computations for predictions were still involved in their algorithms.

In this paper, we proposed a simple but efficient MD algorithm by using a dynamic group to reduce the candidate modes in order to decrease the computational cost. The video contents have high temporal and spatial correlation between the current and the previously encoded frames, which can be exploited to group the modes. Meanwhile, the change of video scenes in a video sequence could lead to the decrease of performance if continuously utilizing the improperly selected modes of formerly encoded frames to create the dynamic groups. Therefore, adaptive adjustment schemes are necessary to compliment the algorithm. The scheme based on the undulation of bitrate [7] and PSNR can effectively detect the tendency of degradation of encoding performance and adaptively adjust the algorithm to minimize the degradation of performance. The proposed algorithm is simple and efficient, which reduces up to 55.89% of whole encoding time. The decreases of encoding performance are limited in an acceptable level according to the extensive simulation results.

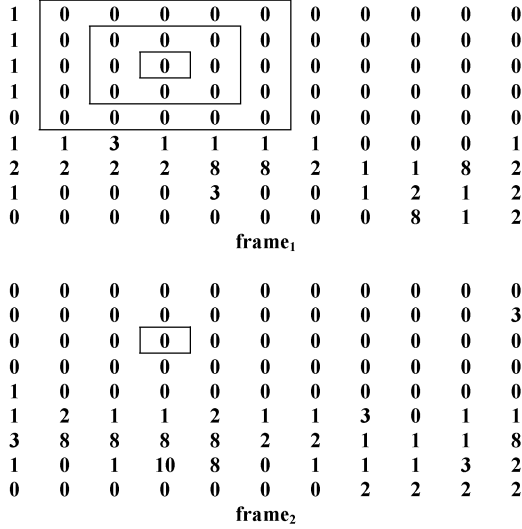The rest of this paper is structured as follows: In Section II,

$$
\begin{array}{l|ccccc|ccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 3 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
2 & 2 & 2 & 2 & 8 & 8 & 2 & 1 & 1 & 8 & 2 \\
1 & 0 & 0 & 0 & 3 & 0 & 0 & 1 & 2 & 1 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 8 & 1 & 2 \\
\end{array}
$$

$$\text{frame}_1$$

$$
\begin{array}{l|ccccc|ccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 1 & 1 & 2 & 1 & 1 & 3 & 0 & 1 & 1 \\
3 & 8 & 8 & 8 & 8 & 2 & 2 & 1 & 1 & 1 & 8 \\
1 & 0 & 1 & 10 & 8 & 0 & 1 & 1 & 1 & 3 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \\
\end{array}
$$

$$\text{frame}_2$$

Fig.1. Macroblock mode selection for Highway

■	co-located macroblock
●	first neighbouring macroblock
×	second neighbouring macroblock
(light)	1st edge
(dark)	2nd edge

Fig.2. Macroblocks in a QCIF frame

the principles of proposed algorithm are analyzed. The proposed algorithm is summarized in details in Section III. Section IV presents the experimental results. The conclusion of this paper follows in the last section.

II. PRINCIPLES OF PROPOSED FAST MD ALGORITHM

A. The Correlation of Macroblocks

Usually, the successive frames have strong correlation in temporal domain. Smaller sampling interval (normally, 1/30 second per sampling for television quality requirement) enables two even more adjacent frames to possess very similar objects and backgrounds. Removing the same contents (temporal redundancy) extremely decreases the bitrate, which satisfies the limited storage capacity and the transmission bandwidth. That is the main motivation of motion estimation (ME), which is also known as inter prediction process. Furthermore, the correlation between current macroblock and its neighbouring macroblocks attracts the intra prediction to exploit the spatial redundancy within the same frame. The bitrate is therefore further decreased. According to the experiments, the correlation between macroblocks within one frame and several adjacent frames can be exploited for improving MD. Fig. 1 gives an example that illustrates the best macroblock modes used to encode two frames in the Highway QCIF video sequence. The frame_1 is previously encoded frame and used as the reference for frame_2. In the example, the standard JM reference software was used with full search ME scheme. The number in the figure indicates the adopted best mode. From the figure, it is clearly shown that the macroblocks in the frame_2 (current frame) are highly correlated with their co-located or neighbouring macroblocks in the frame_1 (reference frame). Based on the correlation, we defined a dynamic modes group $A_{(x, y)}$ according to (3):
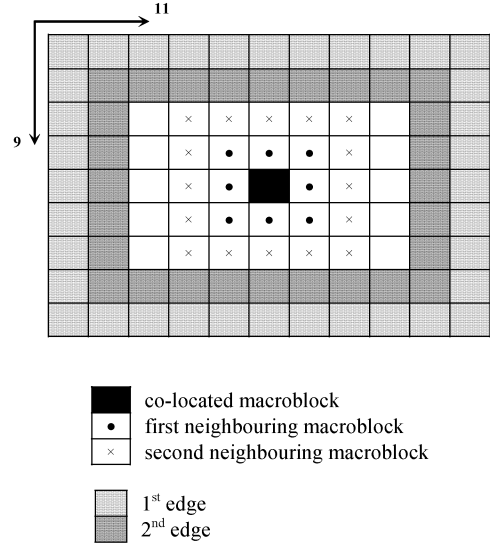
$$
mode_i \in A_{(x,y)} \begin{cases} if\ mode_i\ is\ the\ best\ for\ co\text{-}located\ MB \\ if\ mode_i\ is\ the\ best\ for\ any\ first\ neighbouring\ MB \\ if\ mode_i\ is\ the\ best\ for\ any\ second\ neighbouring\ MB \end{cases} \tag{3}
$$

where $mode_i$ is one of the seven macroblock modes. For most macroblocks, group $A_{(x, y)}$ comprises modes according to the definition in (3). However, if current macroblock is on the first edge of frame, $A_{(x, y)}$ consists of all candidate modes. If current macroblock is on the second edge of frame, only the first two conditions of (3) are satisfied. The edges of frame and the neighbouring macroblocks are indicated in Fig. 2.

According to the correlation between macroblocks, the best mode for macroblock (x, y) should be one member of group $A_{(x, y)}$. Sufficient simulation results indicated that the modes in $A_{(x, y)}$ are usually a small subset of entire modes. Fig. 3 and Fig. 4 are examples to show the number of candidate modes in $A_{(x, y)}$ for Foreman and Bridge-close two sequences in QCIF. A large amount of quiescent backgrounds and the minor motions of objects are contained in Bridge-close, the number of modes in each candidate group is therefore very small (mostly equaling to 1 or 2). This characteristic is very obvious for video sequences with low motion or smooth background. Compared with the entire modes specified in the standard, the dynamic group, $A_{(x, y)}$, is composed of less candidate macroblock modes. Using the modes in a dynamic group, the encoding time can be reduced significantly with negligible performance degradation in terms of decreased PSNR and increased bitrate.

B. Adjustment Schemes

In the previous subsection, a fast MD scheme was discussed and analyzed by introducing a dynamic group containing the selected candidate modes. As stated before, a
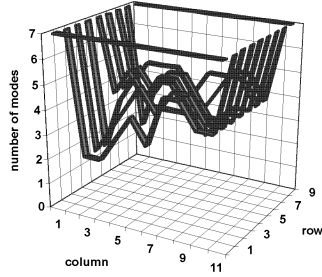
786

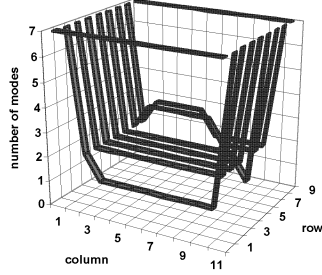Fig.3. Number of modes in each $A_{(x,y)}$ for Foreman

Fig.4. Number of modes in each $A_{(x,y)}$ for Bridge-close

current macroblock usually can find the best mode from the neighbouring macroblocks. However, it is important to keep the best mode through out the whole encoding stage. Particularly, if there is sudden change in video scene, the correlation will become poor. Therefore, an updated dynamic group of modes is necessary. In order to achieve this, an adaptive strategy is introduced to avoid the degradation of performance due to the improper modes in $A_{(x,y)}$.
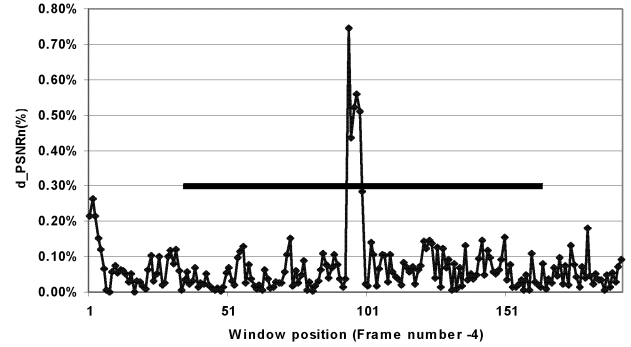
Two adjustment schemes are suggested to optimize the fast MD processing: namely, Fixed-Step adjustment and adaptive adjustment. For the Fixed-Step adjustment, the first of every 30 frames (for frame-rate at 30fps) will be encoded with all modes enabled, and that frame will be used as extra reference of modes for the following 29 frames.

An adaptive adjustment scheme, which can detect the sudden change occurrence in the video scene by monitoring the variation of bitrate, was introduced by authors in [7]. However, the bitrate would remain steady if the rate control (RC) is enabled. In this case, the bitrate based adjustment could not work properly. Considering this restriction, a PSNR based adjustment has been proposed and added into the new fast MD algorithm. The details of two adaptive adjustment schemes are as follows:

For bitrate based adjustment, the bitrate of the most recently encoded 30 frames (depending on the frame rate) will be recorded to form a moving window to monitor the undulation of bitrate. If the change rate of bitrate exceeds the threshold λ_1 (λ_1=3%) for 5 consecutive frames (five fames can avoid the inaccurate adjustment according to the casual change of video scenes) [7], it indicates that sudden change could exist in the video stream and current MD scheme needs to be re-evaluated. The PSNR based adjustment is similar to the bitrate based adjustment. Since the variation in PSNR is

(a) Original PSNR values

(b) d_PSNR$_n$

Fig.5. An example for threshold λ_2

small, the average PSNR value is relatively flat. Therefore, the window size is set to 5 frames. The average value of PSNR, X_PSNR_n, is defined in (4), and the change rate, d_PSNR_n, is defined in (5),

$$X\_PSNR_n = \frac{SNR\_Y_n + SNR\_Y_{n-1} + \cdots + SNR\_Y_{n-4}}{5} \quad (4)$$

$$d\_PSNR_n = ABS(\frac{X\_PSNR_n - X\_PSNR_{n-1}}{X\_PSNR_{n-1}}) \times 100\% \quad (5)$$

where SNR_Y_n is the PSNR value of frame$_n$. Similarly, a decision of sudden change in the video stream could be made if d_PSNR_n exceeds threshold λ_2 for 5 continuous frames. From the intensive simulations, λ_2 is best set as 0.3%, which is then used to effectively adjust the MD scheme. Fig. 5 gives an example that shows the change of video contents occurred at the 100th frame.

III. DETAILS OF PROPOSED FAST MD ALGORITHM

Based on the analysis in the previous section, the proposed fast MD algorithm is summarized here. The best modes selected for the macroblocks in the most previously encoded frame(s) are recorded as the reference to create the dynamic mode groups for current frame. Only the modes contained in group $A_{(x,y)}$ will be considered, and the mode with minimum rate distortion is chosen as the most suitable candidate for

787

current macroblock. Two adaptive adjustment methods are deployed to keep the encoding performance in case of a sudden video scene change. The detailed algorithm is outlined as follows:

Step 1. Encode the I-frame.
Step 2. Set a frame counter C, and $C = 1$.
Step 3. If RC is enabled, skip to Step 4. Otherwise, initialize a moving window to record the bitrates of the most recently encoded 30 frames for bitrate based adaptive adjustment.
Step 4. Initialize a moving window to record the PSNR values of the most recently encoded 5 frames for video quality based adaptive adjustment.
Step 5. If ($C == 1$), current frame is named as f_0. All candidate modes are enabled, encode f_0 and record the best mode selected for each macroblock in a data table T_0 as the reference modes.
Step 6. Encode the coming frames:
  Step 6.1. If current macroblock is on the first edge of the frame, all candidate modes are contained in group $A_{(x, y)}$. Skip to Step 6.4.
  Step 6.2. Check the recorded modes of the co-located and first neighbouring macroblocks in f_0 and the previously encoded P-frame to generate a candidate mode group $A_{(x, y)}$.
  Step 6.3. If current macroblock is not on the first edge and the second edge of frame, check the recorded modes of the second neighbouring macroblocks to update $A_{(x, y)}$.
  Step 6.4. Compare the rate distortion for each mode in $A_{(x, y)}$, the mode leading to the minimum rate distortion is selected as the best mode, and it is recorded as the reference for the next frame.
Step 7. If current macroblock is not the last macroblock of current frame, go back to Step 6.1.
Else if C does not equal to 30, $C += 1$.
Else, reset $C = 1$.
Step 8. Monitor the undulations of bitrate and PSNR:
  Step 8.1. If RC is enabled, skip to Step 8.2.
Else if the bitrate fluctuation exceeds the threshold λ_1 for five consecutive frames, configure the next frame as I-frame and go back to Step 1.
  Step 8.2. If the PSNR fluctuation exceeds the threshold λ_2 for five consecutive frames, configure the next frame as I-frame and go back to Step 1.
Step 9. If current frame is not the last frame of video sequence, go back to Step 5. Otherwise, the processing is finished.

IV. EXPERIMENTAL RESULTS

In our experiments, the reference software JM 10.1 [8] was chosen, and the proposed fast MD algorithm was integrated into it. The profile was set as baseline. The search range was ±16 pixels, and RDO was enabled. Each video sequence was encoded following the pattern of IPPP. The Full Search scheme was deployed in ME, and one frame was used for ME reference. The experiments were performed on an Intel Pentium IV-2.6 GHz computer with 2 GB memory. Two groups of video sequences were selected for the experiments, and they consisted of QCIF and CIF sequences, separately. The length of each sequence is 300 frames. In order to compare the performance, the evaluation of the proposed algorithm was focused on the changing rate of encoding time, PSNR and bitrate: $\triangle Time(\%)$, $\triangle PSNR(dB)$ and $\triangle Bitrate(\%)$ as defined in (6), (7) and (8).

$$\Delta Time = \frac{Time_B - Time_A}{Time_A} \times 100\% \qquad (6)$$

$$\Delta PSNR = PSNR_B - PSNR_A \qquad (7)$$

$$\Delta Bitrate = \frac{Bitrate_B - Bitrate_A}{Bitrate_A} \times 100\% \qquad (8)$$

where subscript A is based on the JM10.1 MD algorithm, and subscript B is based on the proposed MD algorithm.

A. Experiments on QCIF Video Sequences

Eight video sequences were included in this section of experiments: {bridge-far, bridge-close, foreman, carphone, Claire, coastguard, highway and mobile}. They were divided into 3 categories according to characteristics of contents: distance, background and motion. Bridge-far and Bridge-close are two sequences in the first category. They contain similar contents, but the distance between objects and cameras is different. Foreman, carphone and Claire belong to the second category, and they all display people with medium motion under the variable backgrounds. For Foreman, the background is comparably steady. Carphone involves the fast motion of background. Homochromous and steady are the main characteristics of background in Claire. The rest three sequences were classified as the last category by motion. Dual-direction motions of two boats are in Coastguard. Highway shows a car running fast. Although the motion of objects is slow, the moving background is arresting for Mobile.

The experiments were carried out with RC disabled and enabled, respectively. When RC was disabled, four different QP values 28, 32, 36 and 40 were used in the experiments, and Table 1 shows the experimental results. When RC was enabled, QP value was variable in order to keep the bitrate at predefined value. The corresponding results are listed in Table 2. Fig. 6 shows the comparisons of PSNR values versus bitrates between standard JM10.1 MD algorithm and the proposed algorithm. From the experimental results, the encoding time can be saved 38.31% on average, while the loss of PSNR is less than 0.10 dB, and the increase

788

Table 1. QCIF sequences under different quantization parameters

Sequences	QP	28	32	36	40
bridge-far (QCIF)	ΔTime (%)	-54.08	-53.86	-53.58	-54.35
	ΔPSNR (dB)	-0.02	0	0	0
	ΔBitrate (%)	-0.27	0	0	0
bridge-close (QCIF)	ΔTime (%)	-54.22	-55.48	-55.26	-54.04
	ΔPSNR(dB)	-0.02	-0.02	-0.01	-0.02
	ΔBitrate (%)	0.78	0.46	0.99	0.21
foreman (QCIF)	ΔTime (%)	-21.74	-22.05	-23.71	-33.15
	ΔPSNR(dB)	-0.02	-0.02	-0.05	-0.06
	ΔBitrate (%)	0.43	0.34	0.81	0.12
carphone (QCIF)	ΔTime (%)	-23.58	-24.75	-29.65	-38.16
	ΔPSNR(dB)	-0.02	-0.06	-0.06	-0.14
	ΔBitrate (%)	0.18	0.04	0.65	1.18
Claire (QCIF)	ΔTime (%)	-45.09	-46.35	-48.35	-50.91
	ΔPSNR(dB)	-0.06	-0.10	-0.18	-0.12
	ΔBitrate (%)	0.75	0.60	-0.38	-0.99
coastguard (QCIF)	ΔTime (%)	-24.36	-23.77	-27.54	-31.54
	ΔPSNR(dB)	-0.01	-0.03	-0.02	-0.05
	ΔBitrate (%)	0.14	0.16	0.57	-0.20
highway (QCIF)	ΔTime (%)	-30.18	-37.06	-41.45	-46.01
	ΔPSNR(dB)	-0.03	-0.07	-0.02	-0.14
	ΔBitrate (%)	0.69	1.09	0.97	-0.21
mobile (QCIF)	ΔTime (%)	-29.18	-28.07	-25.53	-26.15
	ΔPSNR(dB)	0	-0.01	-0.01	-0.04
	ΔBitrate (%)	0.04	-0.08	0.72	0.65

Table 3. CIF under different quantization parameters

Sequences	QP	28	32	36	40
bridge-far (CIF)	ΔTime (%)	-26.02	-22.05	-26.32	-28.42
	ΔPSNR (dB)	-0.01	-0.01	0	0
	ΔBitrate (%)	-0.04	-0.39	0	0
bridge-close (CIF)	ΔTime (%)	-29.97	-30.31	-30.67	-31.60
	ΔPSNR(dB)	-0.01	-0.01	-0.02	-0.05
	ΔBitrate (%)	0.18	0.55	0.22	0.46
foreman (CIF)	ΔTime (%)	-12.43	-13.29	-16.02	-21.80
	ΔPSNR(dB)	-0.02	-0.04	-0.06	-0.05
	ΔBitrate (%)	0.22	0.47	-0.06	0.82
Stefan (CIF)	ΔTime (%)	-15.54	-14.65	-12.64	-10.96
	ΔPSNR(dB)	-0.02	-0.02	-0.02	-0.06
	ΔBitrate (%)	0.26	0.33	0.51	0.91
m_d (CIF)	ΔTime (%)	-25.33	-27.03	-29.08	-29.74
	ΔPSNR(dB)	-0.03	-0.03	-0.07	-0.06
	ΔBitrate (%)	0.89	0.90	0.38	0.90
Paris (CIF)	ΔTime (%)	-24.00	-25.55	-26.32	-27.31
	ΔPSNR(dB)	-0.02	-0.07	-0.09	-0.07
	ΔBitrate (%)	0.65	0.84	0.40	1.41
container (CIF)	ΔTime (%)	-25.08	-23.96	-24.38	-26.36
	ΔPSNR(dB)	-0.02	-0.04	-0.05	-0.05
	ΔBitrate (%)	0.72	1.36	0.80	0.78
highway (CIF)	ΔTime (%)	-24.57	-26.43	-42.43	-28.51
	ΔPSNR(dB)	-0.02	-0.05	-0.10	-0.10
	ΔBitrate (%)	-0.35	0.62	0.16	0

Table 2. QCIF sequences under rate control (45 Kbps)

Sequences	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
bridge-far	-49.80	0.02	-0.09
bridge-close	-55.89	-0.04	0.15
foreman	-30.03	-0.06	-0.02
carphone	-30.74	-0.10	0.02
Claire	-46.91	-0.05	0.24
coastguard	-27.67	-0.04	0.04
highway	-37.62	-0.04	-0.07
mobile	-40.70	-0.10	0

Table 4. CIF sequences under rate control (45 Kbps)

Sequences	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
bridge-far	-28.23	0.01	0
bridge-close	-32.65	0	-0.57
foreman	-26.83	0	0
m-d	-29.70	-0.12	-0.02
Stefan	-27.58	-0.06	0.42
Paris	-30.89	-0.07	-0.02
container	-30.19	-0.09	0
highway	-29.70	-0.04	-0.06

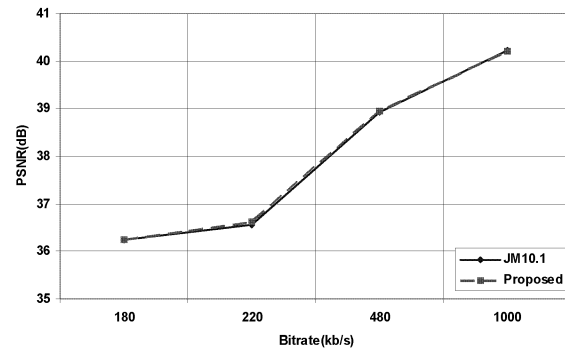Fig.6. Bitrate vs. PSNR for Foreman (QCIF)

Fig.7. Bitrate vs. PSNR for Highway (CIF)

of bitrate is less than 1%. For different bitrates, the quality of reconstructed video is very close to the video encoded by the exhaustive MD algorithm.

B. Experiments on CIF Video Sequences

In this section of experiments, there were eight sequences

in CIF format: {bridge-far, bridge-close, foreman, Steven, mother_daughter (m_d), Paris, container and highway}. It is similar to the experiments in Section IV-A, we also classified sequences into the same three categories. The experimental results are listed in Table 3 and Table 4. From the experimental results, the encoding time can be reduced 25.36% on average, while the loss of PSNR is less than 0.10

789

Table 5. Comparison with Grecos's proposed algorithm [3]

Sequences	Algorithm	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
QP = 28				
bridge-far (QCIF)	Grecos's	-72.82	-0.05	0
	ours	-54.08	-0.02	-0.27
foreman (QCIF)	Grecos's	-12.50	0	0.67
	ours	-21.74	-0.02	0.43
carphone (QCIF)	Grecos's	-13.21	-0.13	-0.11
	ours	-23.58	-0.02	0.18
coastguard (QCIF)	Grecos's	-11.7	-0.01	-0.35
	ours	-24.36	-0.01	0.14
Stefan (CIF)	Grecos's	-10.69	0	0.30
	ours	-15.54	-0.02	0.26
QP = 32				
bridge-far (QCIF)	Grecos's	-78.96	0	0
	ours	-53.86	0	0
foreman (QCIF)	Grecos's	-15.44	-0.03	-0.72
	ours	-22.05	-0.02	0.34
carphone (QCIF)	Grecos's	-17.64	-0.02	-0.33
	ours	-24.75	-0.06	0.04
coastguard (QCIF)	Grecos's	-17.64	-0.03	-0.73
	ours	-23.77	-0.03	0.16
Stefan (CIF)	Grecos's	-15.78	-0.06	-0.12
	ours	-14.65	-0.02	0.33
QP = 40				
bridge-far (QCIF)	Grecos's	-73.76	0	0
	ours	-54.35	0	0
foreman (QCIF)	Grecos's	-22.67	-0.10	0.11
	ours	-33.15	-0.06	0.12
carphone (QCIF)	Grecos's	-28.82	0.13	0.60
	ours	-38.16	-0.14	1.18
coastguard (QCIF)	Grecos's	-31.20	-0.26	-1.96
	ours	-31.54	-0.05	-0.20
Stefan (CIF)	Grecos's	-27.66	-0.26	-0.44
	ours	-10.96	-0.06	0.91

dB, and the increase of bitrate is less than 1%. Fig. 7 illustrates the comparison of the proposed algorithm with standard JM10.1 in terms of PSNR vs. bitrate. Again, it shows that the encoded video quality in PSNR is very similar by using the proposed algorithm and the exhaustive MD algorithm.

Furthermore, Table 5 shows the comparison results using our proposed fast MD algorithm to another fast MD algorithm by Grecos et al. [3]. It can be seen in most cases our algorithm outperforms their one in [3] under similar experimental environment.

V.   CONCLUSION

In this paper, an efficient MD algorithm has been proposed. By exploiting the correlation of macroblocks in both temporal and spatial domains, the proposed algorithm can reduce the computational complexity significantly. Two adjustment schemes to minimize the loss of PSNR and the increase of bitrate were also introduced. The algorithm is relatively simple and efficient to implement. It is therefore very attractive to real-time and low power consumption applications.

REFERENCES

[1]   "Draft ITU-T Rec. and FDIS of Joint Video Spec. (H.264 | ISO/IEC 14496-10 AVC)," JVT of MPEG and VCEG, Doc. JVT-G050rl, May 2003.

[2]   T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, pp.560-576, July 2003.

[3]   C. Grecos and M. Y. Yang, "Fast inter mode prediction for P slices in the H264 video coding standard," *IEEE Trans. Broadcasting*, vol.51, pp. 256-263, June 2005.

[4]   F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol.15, pp. 813-822, July 2005.

[5]   J. F. Wang, J. C. Wang, J. T. Chen, A. C. Tsai, and A. Paul, "A novel fast algorithm for intra mode decision in H.264/AVC encoders," *Proc. ISCAS 2006*, pp. 3498-3501, May 2006.

[6]   H. Wang, S. Kwong, and C. W. Kok, "An efficient mode decision algorithm for H.264/AVC encoding optimization," *IEEE Trans. Multimedia*, vol.9, pp. 882-888, June 2007.

[7]   B. Zhan, B. Hou, and R. Sotudeh, "A novel fast inter mode decision algorithm based on statistic and adaptive adjustment for H.264/AVC," *Proc. IEEE SoftCOM 2007, Sydney*, September 2007.

[8]   JVT reference software unofficial version JM10.1, http://iphome.hhi.de/suehring/tml/download/jm10.1.zip.

790