AN EFFICIENT FAST MODE DECISION ALGORITHM FOR H.264/AVC INTRA/INTER PREDICTIONS

Bin Zhan, Baochun Hou and Reza Sotudeh

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

ABSTRACT

H.264/AVC is the newest video coding standard, which outperforms the former standards in video coding efficiency in terms of improved video quality and decreased bitrate. Variable block size based mode decision (MD) with rate distortion optimization (RDO) is one of the most impressive new techniques employed in H.264/AVC. However, the improvement on performance is achieved at the expense of significantly increased computational complexity, which is a key challenge for real-time applications. An efficient fast mode decision algorithm is then proposed in this paper. By exploiting the correlation between macroblocks and the statistical characteristics of sub-macroblock in MD, the video encoding time can be reduced 52.19% on average. Furthermore, the motion speed based adjustment scheme was introduced to minimize the degradation of performance.

Index Terms— H.264/AVC, fast mode decision, rate distortion optimization, video coding

1. INTRODUCTION

The H.264/AVC [1] video coding standard introduces several new techniques to achieve high performance video compression in terms of improved video quality under low bitrate [2]. In contrast to the unique 16×16 macroblock size employed in previous standards, variable block size enables the most accurate intra/inter predictions. Especially, rate distortion optimization (RDO) based mode decision (MD) balances the residuals and control information (such as motion vectors) to choose the best suitable modes for macroblocks. The Lagrangian cost function defined in (1) is frequently used as the MD criterion, and the mode offering the minimal value is selected as the best mode among all possible modes.

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

In (1) *MODE* is one of the candidate modes; λ_{MODE} is the Lagrangian multiplier related to the quantization parameter

(*QP*). The sum of squared differences (*SSD*) gives the distortions between current macroblock (*s*) and reference macroblock (*c*). *R* denotes the rate for current mode. Obviously, complicated calculation of *J* consumes huge amount of encoding time. For each macroblock, *J* has to be calculated, which is repeated and a major challenge for real-time applications.

Since H.264/AVC was finalized in 2003, several fast MD algorithms have been developed to reduce the complexity of computations. In [3], Grecos and Yang proposed a fast scheme which exploits neighbourhood information and a set of constraints to enhance the MD. Pan et al. introduced an approach to reduce the intra candidate modes according to a pre-established local edge direction histogram in [4]. Using the characteristics of video objects in the fast MD has been proposed by Wu et al. [5], and they also optimized the MD by early terminating the unnecessary modes according to the thresholds based prediction in [6]. Recently, we introduced a fast mode decision algorithm, which adopts the statistical characteristics of MD to simplify the computational complexity [7]. In this paper, the correlation in MD and the statistical characteristics of sub-macroblock are investigated, and a new fast MD algorithm is proposed. The proposed fast MD algorithm is simple and efficient which can save 52.19% encoding time on average, and the degradation of encoding efficiency in terms of video quality and bitrate is negligible by using motion speed based adjustment.

This paper is organized as follows. The next section introduces the principles and details of the proposed MD algorithm. The experimental results are given in the section 3. The conclusion of this paper is drawn thereafter.

2. PROPOSED MODE DECISION ALGORITHM

2.1. Fast Mode Decision

Generally, the adjacent video frames in the same video sequence present very strong correlation, which enables the encoder to remove the temporal and spatial redundancy from the video sequence. Our investigation revealed that the
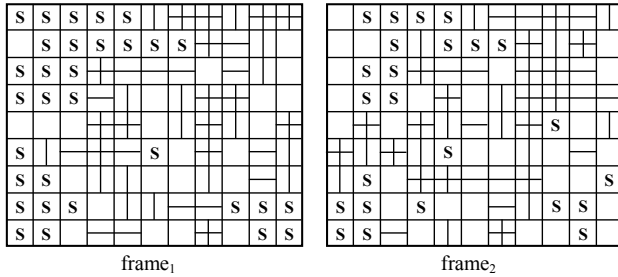
Fig. 1. The best macroblock modes of the first two adjacent frames in Foreman.

Fig. 2. Co-located, inner ring and outer ring macroblocks.

correlation characteristics of macroblocks can be exploited to simplify the best modes selection in contrast to the exhaustive MD.

2.1.1. Macroblocks not on the edge of frame

An example is given in Fig.1, which illustrates the best mode for each macroblock in two successively encoded P-frames of Foreman, where $frame_1$ is previously encoded and used as the reference for $frame_2$, and S indicates the SKIP mode. It is clearly shown that the macroblocks (not on the edge) in $frame_2$ are highly correlated with their co-located or surrounding macroblocks in the $frame_1$. Fig.2 shows the macroblock (x,y) in current frame and its co-located and neighbouring macroblocks in reference frame, where surrounding macroblocks a-h form an inner ring, while macroblocks A-P form an outer ring of the co-located macroblock. Furthermore, the collection of modes consists of the best modes for the co-located and surrounding macroblocks will be a sub-set of the entire modes defined in the standard, and in most cases this number of modes within inner and outer rings is much less than total modes. If we use the best modes from previously encoded frames to build up a candidate mode list $L_{(x, y)}$ instead of the entire modes, the encoding time can be saved considerably. Therefore, the candidate mode list $L_{(x, y)}$ is defined as:

1. $L_{(x, y)}$ contains the best modes of co-located and two rings macroblocks if the outer ring is available.

2. Otherwise, $L_{(x, y)}$ contains the best modes of co-located and inner ring macroblocks

In order to keep the relevant information, the first of every 30 frames (the number depends on the frame rate) will be encoded with exhaustive MD scheme, and the best modes also will be recorded as reference for the following 29 frames to update the dynamic mode lists.

2.1.2. Macroblocks on the edge of frame

Our experiments show that about 35% encoding time can be saved if adopting the scheme introduced in 2.1.1. However, the macroblocks on the edge of frame possess notable percentage of all macroblocks: 36% for QCIF and 19% for CIF. Additional improvement of the MD process on those macroblocks could further reduce the encoding time. Since
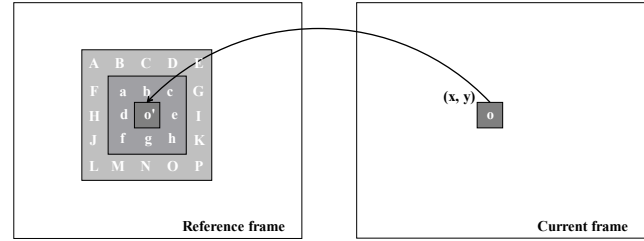
there are insufficient reference macroblocks around the current macroblock on the edge of a frame, the spatial correlation in MD provides less information. The temporal correlation is therefore exploited. The SKIP mode is selected only when the macroblock (16×16) owes zero motion vectors (MV) and the residuals are also zeros, which means the macroblock is the same as its co-located macroblock in reference frame. Taking into account the high temporal correlation between successive frames, we add additional constraint into the dynamic list $L_{(x, y)}$ by modifying the fast MD scheme for macroblock on the edge:

1. If the best mode of the co-located macroblock in the most previously encoded frame is SKIP, $L_{(x, y)}$ consists of only SKIP and inter16×16.

2. Otherwise, all candidate modes are contained in $L_{(x, y)}$.

2.1.3. Fast Sub-macroblock Inter Mode Decision

In [7], we indicated that the best mode for macroblock and sub-macroblock follows certain statistical characteristics: the percentage of each mode and sub-mode will be converge to the steady status after few frames have been encoded, and removing some low-percentage modes can reduce the complexity of MD process without obviously degrade the encoding performance. The strategies introduced in 2.1.1 and 2.1.2 are for macroblock MD, and it can be further enhanced by incorporating the inter sub-macroblock MD based on the statistical characteristics in [7].

2.2. Motion Speed Based Adaptive Adjustment

In general, the values of MVs provide information on the speed and the direction of movement of objects. Smaller values are obtained for relatively static background and smoothly moving objects, and larger values indicate the speedy motion. Taking into consideration the high-speed movement of objects, the above fast MD strategy could exclude the best mode from the dynamic list, and the encoding performance would be degraded if there is no further adjustment scheme deployed. In the proposed algorithm, MVs are exploited to form the criterion to adaptively adjust the current dynamic candidate mode list, $L_{(x, y)}$. We defined the displacement of MVs as the threshold to judge the status of movement:

- If $MV_x >= TH$ or $MV_y >= TH$, the motion speed of objects contained in the current macroblock is regarded as fast;

where subscript x indicates the horizontal direction and y indicates the vertical direction. *TH* is a predefined threshold used to detect the high-speed motion, and the threshold is set to 5 based on large experiments. For simplicity, the best MVs for inter 16×16 mode are checked only. A flag will be set for the macroblock if MV exceeds the threshold. While encoding a macroblock, the fast MD strategy will be available if no flag is set in the co-located macroblock from the reference frame; otherwise, all modes will be enabled for the macroblock.

2.3. Detailed Fast Mode Decision Algorithm

Based on the analysis given in previous two sub-sections, the complete MD algorithm is proposed. Before introduce the algorithm, several parameters will be defined:

- *FLAG*: indicates possible high speed motion.
- T_0: is mode reference table used to record the selected modes and *FLAGs* in the first frame within a group of frames (*FG*).
- T_1: is another mode reference table used to record the selected modes and *FLAGs* in the most previously encoded frame.
- *TH*: is the threshold for motion speed detection.
- λ: is the threshold to eliminate the redundant sub-macroblock modes, and it is set to 5% [7].

Step 1. Initialize T_0, T_1 and inter frame counter C_f=1.

Step 2. Encode the I-frame by using the exhaustive MD.

Step 3. If the current macroblock *(x, y)* is in the first frame of current *FG*, the mode list $L_{(x, y)}$ consists of exhaustive candidate macroblock and sub-macroblock modes, and go to Step 5; otherwise, go to Step 4.

Step 4. If *(x, y)* is on the edge of frame
{
   If the mode of co-located position in T_0 (the second frame of *FG*) or T_1 (the rest frames of *FG* excluding the first and second frames) is *SKIP*, $L_{(x, y)}$ consists of the entire macroblock modes.
}
Else if *FLAG* at the co-located position in T_1 (or T_0 for the second frame of *FG*) is *HIGH*, $L_{(x, y)}$ consists of the entire macroblock modes.
Else
{
   If the outer ring macroblocks of *(x, y)* are not complete, $L_{(x, y)}$ consists of the modes of the co-located and inner ring positions in T_0 and T_1 (T_1 is not available for the second frame of *FG*).
   Otherwise, $L_{(x, y)}$ consists of the co-located,

inner ring and outer ring positions in T_0 and T_1 (T_1 is not available for the second frame of *FG*).
}

Step 5. Encode *(x, y)* based on the current $L_{(x, y)}$, and update T_0 and T_1. Record the number of selection for each sub-macroblock mode. If *(x, y)* is not the last macroblock in the current frame, go back to Step 3. Otherwise, go to Step 6.

Step 6. If (C_f==15), calculate the percentage for each sub-macroblock mode, eliminate the sub mode if its ratio is less than λ. Otherwise, C_f=C_f+1.
If the current frame is the last of video sequence, the encoding process is finished. Otherwise, go back to Step 3.

3. EXPERIMENTAL RESULTS

Experiments have been carried out on an Intel Pentium IV 2.66 GHz PC with 2 GB memory. Ten commonly used video sequences were selected as the test sequences:

QICF   Bridge-far, Bridge-close, Foreman, Carphone, Claire, Mobile and Coastguard

CIF   Highway, Stefan and Container

Each sequence contains 300 frames. The proposed fast MD scheme was integrated into reference software JM10.1 [8], and the configurations are listed as follows:

- Baseline profile
- Search range: [-16, 16]
- Sequence type: IPPP
- Motion estimation scheme: Full Search
- Number of reference frames: One

The evaluation of performance of the proposed algorithm was focused on the change of encoding time, PSNR and bitrate, i.e., *△Time(%)*, *△PSNR(dB)* and *△Bitrate(%)*, and their calculations are defined in (2), (3) and (4):

$$\Delta Time = \frac{Time_B - Time_A}{Time_A} \times 100\% \tag{2}$$

$$\Delta PSNR = PSNR_B - PSNR_A \tag{3}$$

$$\Delta Bitrate = \frac{Bitrate_B - Bitrate_A}{Bitrate_A} \times 100\% \tag{4}$$

where subscript *A* indicates the result under the exhaustive MD algorithm specified in JM10.1, and subscript *B* indicates the result under proposed fast MD algorithm. Table 1 lists the experimental results when different QP values were deployed. According to the results, the encoding time can be saved 52.19% on average. In most cases, the decrease of PSNR is under 0.1dB with less than 2% increase of bitrate. Especially for some test sequences with low-speed motion and steady background, the encoding time is reduced up to 81.40%, and the decrease of encoding performance is negligible. Fig. 3 illustrates the rate distortion (RD) curves for several test sequences, which

Table 1. Experimental results.

Sequence	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
QP = 28			
Bridge-far	-81.13	-0.04	1.08
Bridge-close	-75.34	-0.03	0.90
Foreman	-28.83	-0.03	1.06
Carphone	-34.23	-0.05	0.37
Claire	-70.21	-0.09	1.11
Mobile	-31.20	0	0.14
Coastguard	-30.64	-0.01	0.14
Highway (CIF)	-49.23	-0.06	0.76
Stefan (CIF)	-22.89	-0.02	0.14
Container (CIF)	-63.24	-0.04	0.54
QP = 32			
Bridge-far	-81.06	0	0
Bridge-close	-78.47	-0.03	1.55
Foreman	-30.22	-0.06	1.95
Carphone	-37.83	-0.12	0.51
Claire	-67.98	-0.12	1.33
Mobile	-30.48	-0.02	0.01
Coastguard	-44.42	-0.07	1.00
Highway (CIF)	-56.14	-0.10	1.20
Stefan (CIF)	-22.23	-0.01	0.18
Container (CIF)	-65.35	-0.06	0.68
QP = 36			
Bridge-far	-79.85	0	0
Bridge-close	-81.40	-0.04	2.78
Foreman	-44.27	-0.09	2.67
Carphone	-48.97	-0.14	0.27
Claire	-68.37	-0.17	2.57
Mobile	-34.48	-0.04	0.56
Coastguard	-47.88	-0.08	0.61
Highway (CIF)	-69.98	-0.19	1.18
Stefan (CIF)	-23.24	-0.02	0.26
Container (CIF)	-66.23	-0.06	1.25

Table 2. Comparison with Grecos's fast MD algorithm [3] (QP=28).

| Sequence	Fast MD	ΔTime (%)	ΔPSNR (dB)	ΔBitrate (%)
Bridge-far	Grecos's	-72.82	-0.05	0
	proposed	-81.13	-0.04	1.08
Foreman	Grecos's	-12.50	0	0.67
	proposed	-28.83	-0.03	1.06
Carphone	Grecos's	-13.21	-0.13	-0.11
	proposed	-34.23	-0.05	0.37
Coastguard	Grecos's	-11.70	-0.01	-0.35
	proposed	-30.64	-0.01	0.14
Stefan (CIF)	Grecos's	-10.69	0	0.30
	proposed	-22.89	-0.02	0.14

illustrate that the proposed fast MD algorithm performs close to the exhaustive MD algorithm. Table 2 gives the comparison between the proposed algorithm and Grecos's algorithm [3], which is under the same experimental configurations, and the proposed algorithm can give superior performance.

4. CONCLUSIONS

Correlation and statistical characteristics of macroblocks and sub-macroblocks in MD are investigated, and an efficient fast MD algorithm has been introduced in this paper. By skipping a lot of candidate modes, the encoding
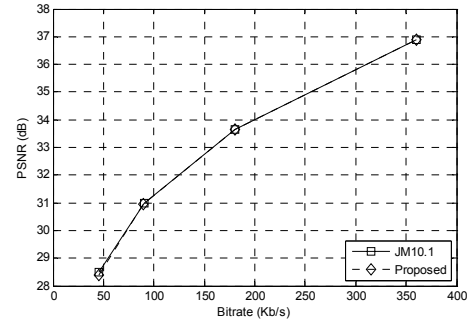
a). Foreman
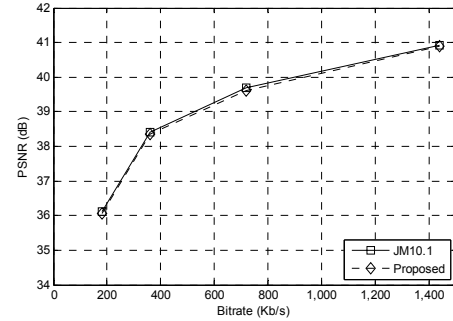
b). Highway (CIF)

Fig. 3. RD curves.

time can be saved up to 81.40%, and in most cases, the loss of video quality is less than 0.1 dB, and the increase of bitrate can be limited in 2%.

5. 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 H.264 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] D. Wu, F. Pan, K.P. Lim, S. Wu, Z.G. Li, X. Lin, S. Rahardja, and, C.C. Ko "Fast intermode decision in H.264/AVC video coding," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 15, pp. 953-958, July, 2005.

[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, "Fast mode decision for inter-prediction in H.264/AVC," *IEEE Proc. 7th ISCIT*, Sydney Australia, pp. 485-490, October 2007.

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