Evaluation of the Image Backtrack-Based Fast Direct Mode Decision Algorithm

Yungho Choi* and Neungsoo Park**

Abstract—B frame bi-directional predictions and the DIRECT mode coding of the H.264 video compression standard necessitate a complex mode decision process, resulting in a long computation time. To make H.264 feasible, this paper proposes an image backtrack-based fast (IBFD) algorithm and evaluates the performances of two promising fast algorithms (i.e., AFDM and IBFD). Evaluation results show that an image backtrack-based fast (IBFD) algorithm can determine DIRECT mode macroblocks with 13% higher accuracy, as compared with the AFDM. Furthermore, IBFD is shown to reduce the motion estimation time of B frames by up to 23% with a negligible quality degradation

Keywords—H.264/AVC, Bi-Directional Prediction, Image Backtracking, Fast Algorithm, Mode Decision

1. INTRODUCTION

In many multimedia services, image compression techniques reduce data size without hampering image quality, which enables the transmission and the storage of image data with fewer resources [1, 2]. From among such techniques, H.264 [3] is one of the recent image compression standards and has been used for many digital image devices, such as monitoring cameras, camcorders, and mobile phones. For high compression efficiency, the H.264 standard employs B frame encoding [4] and its DIRECT mode [5]. B frame encoding enables the reduction of image data redundancy by supporting bi-directional predictions. However, although this scheme reduces image data redundancy, it may suffer from larger side information, which is caused by encoding multiple motion vectors and multiple reference picture indexes. To resolve this problem, the DIRECT mode for B frame encoding is introduced in H.264. The DIRECT mode avoids encoding motion vectors by allowing decoders to predict those vectors that are utilizing previously coded information temporally or spatially.

Generally, B frame encoding is considered to be one of the most time consuming tasks because it needs to select an optimal mode that minimizes the rate distortion (RD) cost of a macroblock encoded currently by calculating complicated motion estimations and bi-directional predictions. Thus, if the DIRECT mode can be determined at an early stage, skipping the complex mode selection processes can save on considerable coding time.

To skip the complex mode selection processes, some fast algorithms have been proposed [6-

8]. One of these algorithms is the Adaptive Fast Direct Mode Decision (AFDM) Algorithm [9]. In terms of optimal modes and rate-distortion costs, this scheme exploits a strong relationship between a macroblock being encoded currently and its co-located macroblocks in previously encoded frames. This enables the determination of the DIRECT mode coding for the current macroblock without a complex mode decision process, which increases the B frame encoding speed significantly. However, the AFDM algorithm leaves out the cases where the co-located macroblock's best mode is neither the DIRECT_16x16 nor the INTER_16x16 mode. Although, even in these cases, the macroblock's optimal mode can be the DIRECT mode. Therefore, this algorithm has a limitation with regard to speeding up the B frame encoding as it suffers from its limited DIRECT mode identification conditions.

To enhance the performance of the AFDM algorithm, this paper proposes an image back-track-based fast DIRECT mode decision (IBFD) algorithm. This algorithm extends the AFDM [9] algorithm and can determine the DIRECT mode even for macroblocks that the AFDM algorithm leaves out. The key idea of IBFD is to locate macroblocks matching the currently coded macroblock in previously encoded frames. According to the coding mode of the located macroblock, IBFD can determine to code the current macroblock in the DIRECT mode. For an extensive evaluation of this algorithm, this paper implements the IBFD and the AFDM using JM 14.2 [10] and evaluates their coding performances by encoding popular video sequences having a variety of image characteristics.

The remainder of the paper is organized as follows. Section 2 describes a previous work, i.e., an adaptive fast direct mode decision algorithm using mode and Lagrangian cost prediction for B frame in H.264/AVC [9]. Section 3 presents an implementation of IBFD. Section 4 evaluates the performances of AFDM and IBFD, and analyzes them. The conclusions are presented in the final section.

2. PREVIOUS WORK

To speed up B frame coding, the AFDM algorithm [9] was proposed. This AFDM algorithm exploits the strong relations of optimal modes and rate-distortion costs (RDcost) between the macroblock (MB) that is currently being encoded and its co-located macroblock. Thus, it determines the DIRECT mode coding for macroblocks without a complex mode decision process, which increases B frame encoding speed significantly.

In this algorithm, the DIRECT mode for a macroblock can be determined by checking the conditions of the AFDM algorithm: (1) $((MD_{PCO.MB} \leq 1$ && $MD_{NCO.MB} \leq 1) \| (MD_{BCO.MB} = 0))$ and (2) $(J_{CURRENT\_MB} < \alpha$ x $J_{PCO\_MB} \| J_{CURRENT\_MB} < \beta$ x $J_{NCO\_MB})$. $MD_{PCO.MB}$, $MD_{NCO.MB}$, and $MD_{BCO.MB}$ are the best mode numbers for the co-located macroblocks (MBs) in the previous P frame, the next P frame, and the previous B frame, respectively. $J_{CURRENT\_MB}$, J_{NCO_MB}, and J_{PCO_MB} are RD costs for the current macroblock and for the co-located macroblocks in the next P frame and in the previous P frame.

The key rationale of this work is that many macroblocks' optimal modes are shown to be DIRECT modes when the optimal modes of the corresponding co-located blocks (i.e., $MD_{PCO.MB}$, $MD_{NCO.MB}$, and $MD_{BCO.MB}$) are either the DIRECT_16x16 mode or the INTER16x16 mode. Therefore, the algorithm can encode macroblocks that satisfy the condition that is given above in the DIRECT mode without spending a lot of time calculating the other coding modes. How-

ever, this is not always true. To overcome this problem, this paper proposes a fast DIRECT mode decision algorithm. The next subsection describes the algorithm and its implementation.

3. IMPLEMENTATION OF AN IMAGE BACKTRACK-BASED FAST DIRECT MODE DECISION ALGORITHM

This section presents an implementation of the image backtrack-based fast DIRECT mode decision (IBFD) algorithm, which might enhance the performance of the Adaptive Fast Direct Mode Decision (AFDM) algorithm [9] by taking care of the macroblocks (MBs) for which the AFDM algorithm needs to run all the complex mode decision processes.

To describe the IBFD algorithm, a simplified pseudo code of the algorithm is given in Fig. 1. As shown in Fig. 1, in the case of macroblocks satisfying the AFDM conditions, the IBFD does the same thing as the AFDM algorithm does. The main difference between AFDM and IBFD algorithms pertains to the macroblocks not satisfying the AFDM conditions. To determine the DIRECT mode coding for such macroblocks, the IBFD locates macroblocks that can match the macroblock that is currently being encoded in the previous P frame, the next P frame, and the previous B frame through an image backtracking process, as shown in Fig. 2. The matched macroblocks are called backtracked macroblocks (BTM) for the currently encoded macroblock, which might be a temporarily displaced image data for the currently encoded macroblock. If the backtracked macroblocks' encoding mode is the DIRECT mode, the possibility that the current macroblock's optimal mode is the DIRECT mode may be high. This allows the IBFD algorithm to code the current macroblock in the DIRECT mode, without a complicated and long mode decision process.

To locate the BTM in previous frames, IBFD can use many methodologies such as SAD (Sum of Absolute Differences), mode pattern-based search, etc. From among the methodologies, this paper is assumed to employ a mode pattern-based search. In this search, the encoding modes of the macroblocks surrounding the current macroblock are used as the pattern to be searched within the previously encoded frames. Namely, the encoding modes of macroblocks, A_{MB}, B_{MB}, C_{MB}, and D_{MB}, in the F frame comprise the encoding mode pattern of the currently encoded mac-

---

For each macroblock M, in B frames:
    1.   if (M satisfy AFDM conditions) then:
        1.1 encode M in the DIRECT mode
    2.   or else
        2.1 locate M's backtracked image macroblocks through image backtracking
        2.2 if (there are backtracked image macroblocks for M) and (M satisfies
            IBFD conditions) then:
            2.2.1   encode M in the DIRECT mode
        2.3 or else
            2.3.1   process all the complicated mode decision calcula-
                    tions.

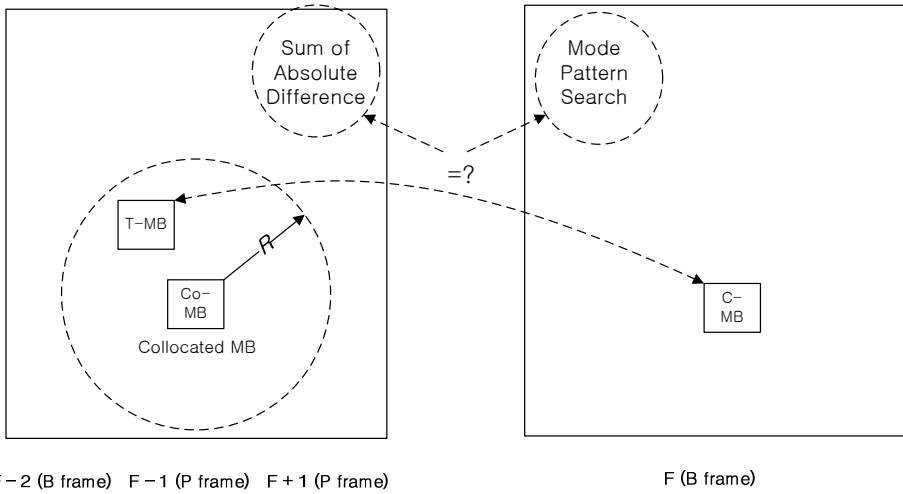Fig. 1. A simplified pseudo code of IBFD

Fig. 2. Image backtracking process based on encoding mode pattern searching

roblock (i.e., C-MB). The encoding mode pattern for each macroblock is searched in the F-1, F-2, and F+1 frames, where F specifies the display order frame number and thus, F, F-1, F+1, and F-2 are the current B frames, the previous P frame, the next P frame, and the previous B frame, respectively. As shown in Fig. 2, the search scope for the encoding mode pattern is limited to *R* from the collocated position of the currently encoded macroblock. This is because, in most cases, objects in a frame cannot move too far in subsequent frames, which thus removes the reason to search an entire frame. If the encoding mode pattern of the currently encoded macroblock does match an encoding mode pattern in previous frames, the matched macroblock, T-MB, is defined to be the BTM of the currently encoded macroblock (i.e., C-MB). The BTM belongs to the same image object of the C-MB. Therefore, if the BTM is encoded in the DIRECT mode, the possibility that the optimal mode of the current macroblock is the DIRECT mode may be high.

However, this is not always true because the BTM is located by not encoding mode patterns according to image data. To increase the accuracy of this DIRECT mode decision algorithm, this scheme uses the following condition to filter out macroblocks having very high rate-distortion costs and thus, it has a low possibility to be encoded in the DIRECT mode.

The filtering condition is satisfied (1) when at least one macroblock in the F-1, F-2, and F+1 frames matches the optimal mode pattern defined above and (2) if the average RD cost of the DIRECT mode macroblocks found in previous frames, is larger than the RD cost of the current macroblock encoded in the DIRECT mode.

Using a more formal notation, the filtering condition is satisfied when at least one macroblock in the F-1, F-2, and F+1 frames matches the optimal mode pattern defined above, if the following condition is satisfied:

$$\text{Average } (RD_{A1}, RD_{B1}, RD_{C1}\ldots) * \alpha \geq RD_{\text{currentMB}}$$

where RD_{A1}, RD_{B1}, RD_{C1}, and $RD_{\text{currentMB}}$ are the RD costs of the DIRECT mode macroblocks found by the pattern matching described above and the current macroblock of α is a constant that

is determined statistically by simulations.

If a macroblock satisfies the condition given above, the IBFD algorithm encodes the macroblock in the DIRECT mode without a complex mode decision process, which saves on encoding time. Otherwise, all of the mode decision calculations are required. In the next section, we evaluate the performance of AFDM and IBFD, and analyze them.

4. PERFORMANCE EVALUATION

To analyze the performance of AFDM and IBFD algorithms, these fast algorithms are implemented in JM 14.2 [10]. For this performance analysis, the following things are assumed: high complexity mode (RDOptimization = 1); GOP = IBPBP; encoding frames including I, B, and P = 199; MV search range = 32, resolution = {qcif, cif}; and QP = 32.

First, the accuracy of the DIRECT mode decision and the average rate-distortion cost (RDcost) error of incorrect mode decisions of the AFDM algorithm and the IBFD algorithm are compared. The results are presented in Tables 1 and 2. In these tables, Total#DM is the total number of macroblocks having the DIRECT mode (DM) as their optimal mode in all the B frames. "DM% in MBs," is the percentage of direct mode macroblocks in all of the frames. The two columns marked "#DM" represent the number of macroblocks that are chosen as direct mode macroblocks in the respective algorithms. Hit% is the ratio of the number of macroblocks that have been correctly determined to the corresponding value of #DM. RDC Error is the RD cost difference between the true optimal mode and a fast direct mode.

As shown in Table 1, the IBFD algorithm improves the accuracy of the direct mode decisions by up to 13%, as compared to the accuracy of the AFDM algorithm (AFDM: 68%, IBFD: 77%, foreman, QCIF). Especially, when the ratio of direct mode macroblocks to the total number of macroblocks is low, the IBFD algorithm outperforms the AFDM algorithm. This is because, in this case, many macroblocks cannot satisfy the condition of the AFDM algorithm, which enforces a complicated mode decision process to determine the DIRECT mode. Thus, this results in a long B frame encoding time.

Additionally, as shown in Table 1, the IBFD algorithm does increase the performance of the AFDM algorithm because, in the cases satisfying the AFDM conditions given in Section 2, the IBFD algorithm does the same thing that the AFDM algorithm does. However, the IBFD algorithm can determine the direct mode macroblocks even when the AFDM conditions are not satisfied. Regarding the RDC error, the IBFD algorithm shows a slightly higher error. However, this value might not suffer significantly because the hit ratio of the IBFD algorithm is higher and thus, fewer macroblocks suffer from it. This can be inferred from Table 2.

Finally, the encoding quality and the METS (Motion Estimation Time Saving) of the AFDM and the IBFD algorithms are compared in Table 2. This comparison is done for QP = 20. The encoding quality is evaluated in an average sense [11], using the fast mode decision algorithms with comparison to the reference JM in terms of BDPSNR and BDBR. A negative BDPSNR result indicates PSNR degradation, whereas a negative BDBR result indicates bit-rate saving. The METS value indicates the ME time savings as a percentage change between the AFDM algorithm and the IBFD algorithm in the high complexity mode.

As shown in Table 2, the IBFD algorithm can reduce ME time by up to 23% with negligible image quality degradation. This proves that the IBFD algorithm better predicts the DIRECT

Table 1. Accuracy comparison of AFDM and IBFD (QP = 32)

Sequences	Size	Total # DM	DM % in MBs	AFDM Algorithm			IBFD Algorithm		
				# DM	Hit %	RDC Error	#DM	Hit %	RDC Error
News	QCIF	8,572	88%	8,782	93%	1,848	8,853	93%	6,538
Foreman	QCIF	6,393	66%	7,036	68%	1,464	7,322	77%	6,107
Hall Monitor	QCIF	8,553	88%	9,351	90%	1,044	9,395	90%	1,574
Container	QCIF	9,262	95%	9,595	96%	1,620	9,621	96%	1,948
Carphone	QCIF	6,309	65%	7,341	71%	1,507	7,538	75%	5,528
Bus	CIF	14,580	50%	17,213	68%	2,719	17,954	69%	10,740
Tempete	CIF	25,272	65%	27,596	73%	2,969	29,549	75%	6,306
Paris	CIF	28,762	74%	33,285	81%	1,590	33,693	82%	4,487
Stefan	CIF	10,792	63%	11,815	75%	2,598	12,496	76%	8,816
Waterfall	CIF	27,556	71%	33,808	73%	1,281	35,337	74%	3,031

Table 2. Performance comparison of AFDM and IBFD algorithms

Sequences	Size	AFDM Algorithm		IBFD Algorithm		METS(%) AFDM-IBFD/AFDM
		BDPSNR	BDBR	BDPSNR	BDBR	
News	QCIF	-0.0135	0.2271	-0.0246	0.4203	9.33
Foreman	QCIF	-0.0626	1.4566	-0.0269	0.6128	4.01
Hall Monitor	QCIF	-0.0172	0.3955	-0.0232	0.5232	18.47
Container	QCIF	0.0086	-0.1911	0.0047	-0.1041	23.21
Carphone	QCIF	-0.0145	0.2954	-0.0218	0.4457	1.31
Bus	CIF	-0.0169	0.3198	-0.0294	0.5594	2.63
Tempete	CIF	-0.0188	0.4324	-0.0312	0.6812	3.47
Paris	CIF	-0.0109	0.1936	-0.0201	0.3601	11.91
Stefan	CIF	-0.0215	0.4254	-0.0311	0.6282	6.01
Waterfall	CIF	-0.0059	0.1149	-0.0225	0.4844	8.27

mode macroblocks and can thus avoid a long and complicated mode decision process time. Additionally, even in the case where the hit ratio of the IBFD algorithm is low in Table 1, the image quality of the IBFD algorithm and the AFDM is not hampered. This indicates that, even in the event of mis-predicting DIRET mode macroblocks, the RD cost errors of the associated macroblocks are relatively small and therefore do not significantly degrade image quality and compression efficiency.

5. CONCLUSION

This work presents an evaluation of an image backtracking-based fast direct mode decision algorithm that extends the AFDM algorithm. Through an extensive evaluation, the IBFD algorithm is shown to outperform the AFDM algorithm by making DIRECT mode decisions not only for macroblocks that satisfy the AFDM conditions but also for macroblocks that do not satisfy these conditions. This leads to the determination of more direct mode macroblocks with a higher hit ratio. Although, in the case of incorrect predictions, the RD cost error of macroblocks

might be increased, this case is rare, and thus, the overall benefit of the IBFD algorithm can conquers its problem, as shown in Table 2.

Additionally, the quality degradation of the IBFD algorithm is negligible due to its prediction accuracy (on average, less than 0.02dB in BDPSNR). Especially, the IBFD algorithm outperforms the AFDM algorithm in video sequences, which move fast or have complicated images. This indicates that, in such sequences, the AFDM algorithm's prediction accuracy is not high enough while the image backtracking of the IBFD algorithm increases prediction accuracy. Overall, the IBFD algorithm effectively reduces B frame encoding time by encoding more macroblocks in the DIRECT mode without executing long mode decision processes.

REFERENCES

[1]   Ankit Chaudhary, Kapil Vatwani, Tushar Agrawal and J.L. Raheja, *"A Vision-based Method to Find Fingertips in a Closed Hand", the Journal of Information Processing Systems*, Vol.8 No.3, 2012, pp.399-408

[2]   Jun Guo, Johee Kim*, "Adaptive Motion Vector Smoothing for Improving Side Information in Distributed Video Coding", the Journal of Information Processing Systems*, Vol. 7 No. 1, 2011, pp.103-110

[3]   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, 1995, pp.560-576

[4]   M. Flierl and B. Girod, *"Generalized B pictures and the draft H.264/AVC video-compression standard", IEEE Trans. Circuits Syst. Video Technol.*, Vol.13, No.7, 2003, pp.587-597

[5]   A. M. Tourapis, F. Wu and S. Li, *"Direct Mode Coding for Bipredictive Slices in the H.264 Standard", IEEE Trans. Circuits Syst. Video Technol.*, Vol.15, No.1, 2005, pp.119-126

[6]   J. Lee, I. Choi, W. Choi, B. Jeon, *"Fast Mode Decision for B slice", Proceedings of JVT-K021*, Antalya, Turkey, 2003

[7]   I. Choi, J. Lee, and B. Jeon, *"Fast Coding Mode Selection With Rate-Distortion Optimization for MPEG-4 Part-10 AVC/H.264", IEEE Trans. Circuits Syst. Video Technol.*, Vol.16, No.12, 2006, pp.1557-1561

[8]   S. Lee, N. Park, Y. Choi*, "Fast Direct Mode Decision Algorithm based on Optimal Mode Pattern Searching", International Journal of Multimedia and Ubiquitous Engineering.*, Vol.7, No.2, April, 2012, pp.415-420

[9]   X. Jin, J. Sun, J. Zhou, Y. Huang, J. Su, T. Ikenaga*, "Adaptive Fast Direct Mode Decision Algorithm using Mode and Lagrangian Cost prediction for B Frame in H.264/AVC", Proceedings of 2011 IEEE International Conference on Multimedia and Expo.* July, 2011, pp.1-6

[10] H.264 Reference Software JM14.2 [Online] http://iphome.hhi.de/suehring/tml/download/old-jm

[11] G. Bjontegaard, Calculation of Average PSNR differences between RD-curves, VCEG-M33, Austin, April, 2001

Yungho Choi

He received his BS and PhD degrees from Yonsei University and University of Southern California, in 1991 and 2001, respectively. He was a senior Engineer in Intel, Shrewsbury, USA. Currently, he is an associate professor in the Department of Electrical Engineering, Konkuk University, Seoul, Korea. His research interests include multiprocessor systems, high performance codec architectures, interconnection networks as well as image recognition.

Neungsoo Park

He received his Ph.D. in Electrical Engineering from the University of Southern California in 2002. He also received his BS and MS in Electrical Engineering from Yonsei University, Seoul, Korea, in 1991 and 1993 respectively. Currently, he is an associate professor in the Department of Computer Science and Engineering, Konkuk University, Seoul, Korea. Prior to joining Konkuk University, he was a senior engineer in Samsung Electronics Co. Ltd., Korea. His research interests include parallel computing, computer architecture, embedded system, high performance computing for signal processing, VLSI computations, and multimedia systems, etc.