Fast Mode Decision Algorithms for Adaptive GOP Structure in the Scalable Extension of H.264/AVC

Chih-Wei Chiou, Chia-Ming Tsai, and Chia-Wen Lin
Department of Computer Science & Information Engineering
National Chung Cheng University
Chiayi 621, Taiwan
cwlin@cs.ccu.edu.tw

***Abstract*—We propose a fast mode decision algorithm to reduce the computational complexity of adaptive GOP structure (AGS) in the scalable extension of H.264/AVC. AGS can improve the coding efficiency of the scalable extension of H.264. It, however, needs to perform motion-compensated temporal filtering (MCTF) of all possible GOP sizes, leading to much higher computation than the fixed GOP structure. In our proposed algorithm, after performing the MCTF with the maximum GOP size, we utilize two features to decide whether to perform the remaining MCTFs of sub-GOPs and mode selection. Experimental results show that the proposed algorithm can significantly reduce unnecessary MCTF computation for AGS, while maintaining good coding efficiency.**

I. INTRODUCTION

With the proliferation of online multimedia contents, the popularity of multimedia streaming technologies, and the establishment of video coding standards, people are able to ubiquitously access and retrieve various multimedia contents via the Internet, promoting networked multimedia services at an extremely fast pace. With video streaming services, users may access a video from heterogeneous networks such as Local Access Network (LAN), Digital Subscriber Line (DSL), cable, wireless networks, and dial-up. The different access networks have different channel characteristics such as bandwidths, bit error rates, and packet loss rates. At the users' end, network appliances including handheld computers, Personal Digital Assistants (PDA), set-top boxes, and smart cellular phones are slated to replace personal computers as the dominant terminals for accessing the Internet. These network terminals vary significantly in resources such as computing power and display capability. To flexibly deliver multimedia data to users with different available resources, access networks, and interests, the multimedia contents may need to be adapted dynamically according to the usage environment. For example, the notion of Universal Multimedia Access (UMA) calls for the provision of different presentations of the same multimedia content, with more or less complexity to suit the different usage environments in which the content is consumed.

There are some traditional methods for video adaptation in a heterogeneous environment [1]. One method is to encode the bitstream at a highest bit-rate/resolution of the Internet then transcodes the bitstream into different bit-rates/formats to meet the requirement of bit-rate/resolution for a client [2]. With a transcoder, the content provider is able to adapt a video bitstream into different bit-rates, resolutions, and formats for different users. Such a transcoder, however, may consume large computing power and time cost to transcode. Another method is to encode a video into multiple bitstreams of different bit-rates, resolution, etc. The server then uses dynamic bitstream switching to send a selected bitstream to a user according to the user's channel requirement or preference. The dynamic bitstream switching scheme requires significantly higher storage cost and may cause drifting problem while performing switching.

Another key technique for achieving content adaptation is scalable video coding (SVC) [3][4][5]. In SVC, the encoder encodes a video sequence into a scalable bitstream. A general scalable bitstream contains one base layer and one or more enhancement layers. According to the requirements of bit-rates/resolutions of users, a server can transmit one base layer for basic video quality/resolution or one base layer plus one/more enhancement layers for higher video quality/resolution. SVC has three dimensions, temporal, spatial and SNR (quality) scalabilities. Besides, these three dimensions can be combined to form the combined scalability. The main advantage of SVC compared with other techniques is that SVC only encodes a video once to form a scalable bitstream and doesn't occupy too much storage capacity of servers. SVC is thus a flexible solution for transmitting video contents over heterogeneous networks. It also allows simple adaptation for various media access devices. The ISO/IEC MPEG and ITU-T formed the Joint Video Team (JVT) to develop an SVC standard as an extension of the H.264/AVC standard [4][5]. The scalable extension of H.264/AVC allows on-the-fly video adaptation in the spatio-temporal and quality dimensions according to the network conditions and receiver capabilities.

For frame-rate adaptation of video streaming, the temporal scalability is an important part of the SVC standard. To achieve the temporal scalability, the motion-compensated temporal filtering (MCTF) technique proposed in [4] is used by the scalable extension of H.264/AVC. The concept of MCTF is to perform the wavelet transform along the motion trajectory. It significantly improves the coding efficiency of SVC compared to the traditional temporal-domain wavelet transform without motion compensation. The implementation of MCTF is based on the lifting scheme [3] that consists of two steps: prediction and update. In addition to the H.264 codec, the MCTF with lifting scheme constitutes the core of SVC.

Adaptive GOP structure (AGS) [7][8] is a new technique that can be used for enhancing the coding performance of the scalable extension of H.264/AVC. The AGS scheme adaptively changes the sizes of GOPs according to the temporal characteristics of a video sequence to improve the coding efficiency of SVC. For better understanding, Fig. 1 shows some possible combinations that consist of some adaptively selected sub-GOPs. In this figure, the

full GOP size is 16 and each number represents the GOP size of the associated sub-GOP. While performing MCTF with AGS, the GOP structure in a full-GOP can be any combinations of sub-GOPs, in which the size of each sub-GOP must be 16, 8, 4, or 2. In the AGS algorithm, the MCTF of every kind of sub-GOPs is performed and the mean squared error (MSE) is calculated for each MCTF. Then, the mode selection step is performed for determining the optimal mode. In the next step, the encoder generates the bitstream according to the optimal mode. Finally, the next full-sized GOP is encoded by repeating this procedure.

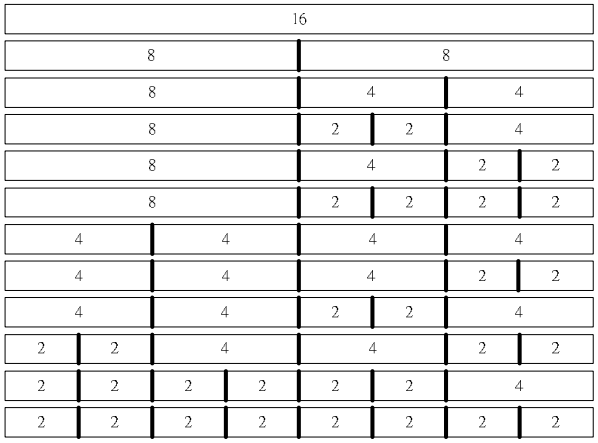


Fig. 1. An example of possible sub-GOP combinations (the full GOP size is 16).

In this paper, we propose to utilize two features, the average motion vector (MV) magnitude and the number of intra mode macroblocks, obtained from the MCTF with full-sized GOP to speed up AGS. For one GOP, if the selected best GOP mode contains sub-GOPs, the video content of this GOP should have larger temporal activities. So the video content of this GOP has relatively larger motion vectors and a larger number of intra mode macroblocks. On the other hand, if the selected best GOP mode only contains the original full-sized GOP; the video content of this GOP should have smaller temporal variation. Thus, this GOP has smaller motion vectors and a smaller number of intra mode macroblocks. In order to reduce the coding complexity of AGS, after executing the first MCTF with full-sized GOP, we can take advantage of these two features to determine whether to perform remaining MCTF procedures of sub-GOPs. As a result, our method can avoid unnecessary MCTF operations of sub-GOPs, thereby reducing the coding complexity of AGS.

II. Prpoposed Feature-Based Fast Mode Decision Algorithms for AGS

According to our experimental results, in terms of run-time, the AGS scheme for SVC [7][8] with full-range mode decision consumes about four times longer than that without AGS, since AGS involves much more MCTF operations of sub-GOPs in mode decision. The aim of this work is to reduce the computational complexity of AGS, while maintaining its good coding efficiency. In order to reduce the complexity of AGS, instead of performing full-range mode decision, we propose to early terminate the mode decision according to some features extracted from the first MCTF procedure. However, how to select useful features to achieve fast and accurate mode decision is an important problem to be addressed.

A. Feature Selection for fast mode decision of AGS

Our goal is to extract reliable features from the first MCTF operation with the full GOP size to reduce the complexity of mode modes in AGS. We can utilize these features to determine whether to perform the following MCTF procedures of sub-GOPs such that the number of unnecessary MCTF operations can be minimized.

Typically, with AGS, full-sized GOPs with high temporal activities will be divided into sub-GOPs. The average MV magnitude, as defined in (1), is a good metric to characterize the temporal motion activity of a GOP.

$$MV_{mag} = \frac{1}{N_{\text{block}} \cdot N_{\text{GOP}}} \sum_{j=1}^{N_{\text{GOP}}} \sum_{i=1}^{N_{\text{block}}} \left(\left| MV_x(i,j) \right| + \left| MV_y(i,j) \right| \right) \quad (1)$$

where $MV_x(i,j)$ and $MV_x(i,j)$ are the horizontal and vertical components of the i-th 4x4 block of the j-th frame in a GOP. N_{GOP} denotes the GOP size, and N_{block} represents the numbers of 4x4 blocks in a frame.

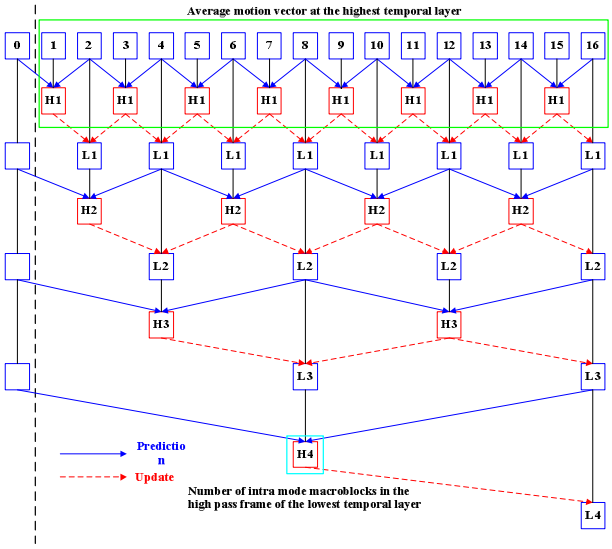


Fig. 2. Features used in the MCTF procedure of full-sized GOP.

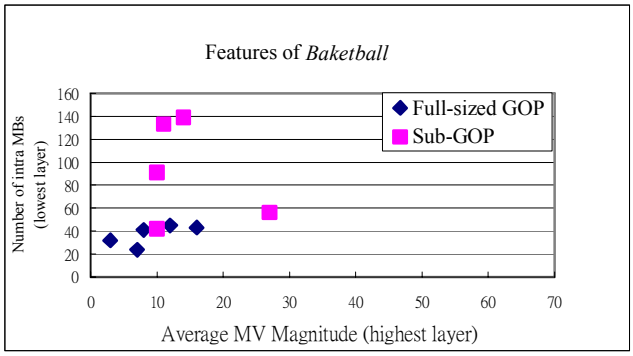
(a) (b)

Fig. 3. Intra-coded blocks in the high-pass frame of the lowest temporal layer in (a) the third GOP (higher temporal activity) and (b) the eighth GOP (lower temporal activity) for *Football*.

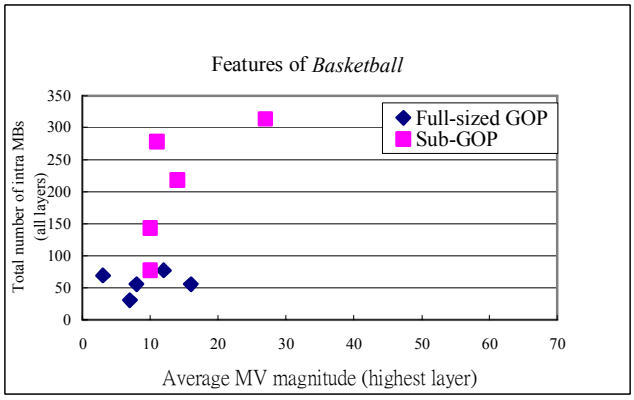
Therefore, as illustrated in Fig. 2, we propose to use the average MV magnitude obtained from the highest temporal layer in the MCTF of full-sized GOP as one feature for mode decision in AGS. In addition to the motion activity feature, the number of intra-coded macroblocks is also a good feature to measure the temporal activities of a GOP. Therefore, we also use the number of intra-coded macroblocks that is computed from the high-pass frame of the lowest temporal layer (or all layers) in the MCTF of full-sized GOP as another feature. Fig. 3 depicts the intra-coded macroblocks

in a high-pass frame of the third and eighth GOPs in the *Football* sequence, which is obtained from the lowest temporal layer in the MCTF operation of full-sized GOP. We can observe the high-pass frame of the third GOP contains a relatively large number of intra-coded macroblocks compared to that of the eighth GOP because the third GOP has higher temporal activity.

Fig. 4 shows the joint distributions of the average MV magnitude and the number of intra-coded blocks in the high-pass frames of MCTF with full-sized GOPs for the *Basketball* test sequence. We classify the GOP modes determined by AGS into the "Full-sized GOP" and "Sub-GOP" modes. The GOPs encoded with the "Full-sized GOP" mode consist of only a full-sized GOP, whereas those coded with the "Sub-GOP" mode are composed of the combinations of sub-GOPs except the full-sized GOP. We can observe from Fig. 4 that the GOPs coded with the "Full-sized GOP" mode usually have relatively smaller average MV magnitudes and fewer numbers of intra-coded macroblocks compared to those coded with the "Sub-GOP" mode. On the other hand, a GOP that has a large MV magnitudes and a large number of intra-coded macroblocks will very likely be partitioned into sub-GOPs in AGS. Other sequences also have similar characteristics.



(a)



(b)

Fig. 4. The distributions of the average MV magnitude (in the highest layer) and the number of intra-coded macroblocks in high-pass frame(s) of (a) the lowest layer, and (b) all the layers of the MCTF operation of full-sized GOP for *Basketball*.

Therefore, after performing the first MCTF operation with full-sized GOP, we can utilize these features to determine whether or not to perform the subsequent MCTF procedures of sub-GOPs. As shown in Fig. 4, it is easier to distinguish the two coding modes if the number of intra-coded blocks is computed from the high-pass frames of all layers (Fig. 4(b)) than from the lowest layer (Fig. 4(a)) of MCTF of full-sized GOP. The reason is that using the intra-block statistics from all layers of MCTF reduces the possibility that a GOP that should be encoded with the "Sub-GOP" mode be erroneously encoded with the "Full-sized GOP" mode, thereby improving coding efficiency. This, however, may reduce the time saving of our algorithm due to the increasing of "Sub-GOP" mode selected by our algorithm.

B. Proposed Fast Mode Decision Algortihm

Fig. 5 shows the flowchart of the proposed method. At first, the first MCTF operation with the full GOP size is performed when the encoder encodes the frames of a GOP. Subsequently, the average MV magnitude and the number of intra-coded macroblocks are calculated as the features for fast mode selection. As mentioned above, the number of intra mode macroblocks can be calculated from the high-pass frame of the lowest temporal layer (Scheme 1) or all high-pass frames of the whole MCTF procedure (Scheme 2).

In the next step, if either of the features is less than its predefined threshold, the following MCTF operations of sub-GOP sizes will not be performed and the AGS procedure is thus early terminated. However, if both of the features are larger than their thresholds, the following MCTF operations and mode decision procedures will be processed. In this way, for GOPs with small temporal activities, unnecessary MCTFs of sub-GOPs can be omitted to reduce the coding complexity of AGS. Besides, for GOPs with high temporal activities, the essential MCTFs of sub-GOPs and the mode selection procedure are still performed to maintain the good coding efficiency of AGS.

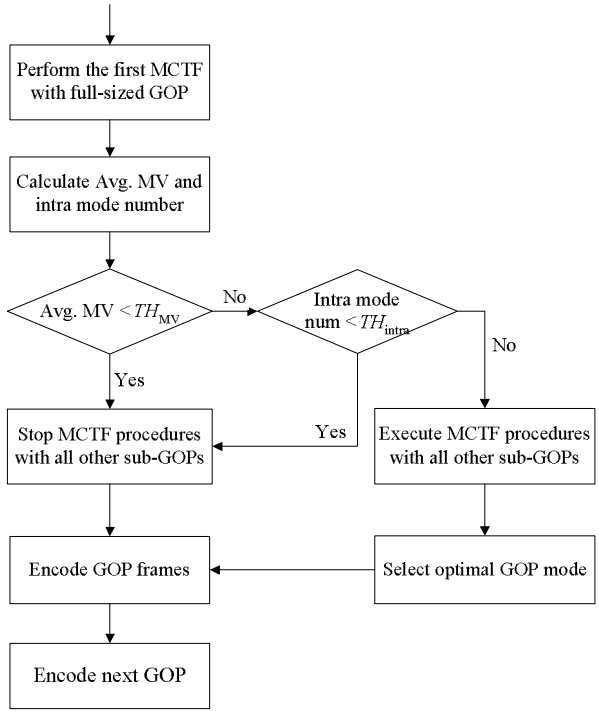


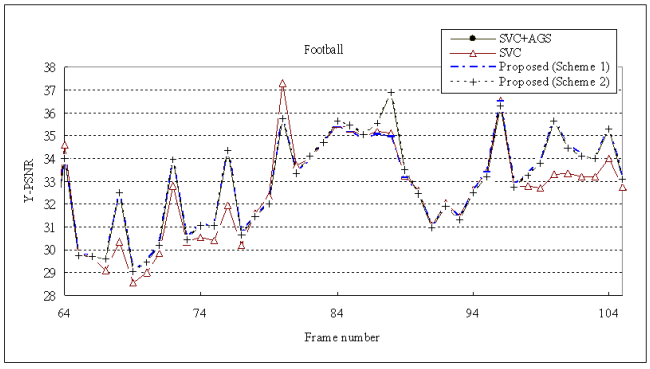
Fig. 5. Flowchart of the proposed fast mode selection algorithm for AGS.

III. Experimental Results

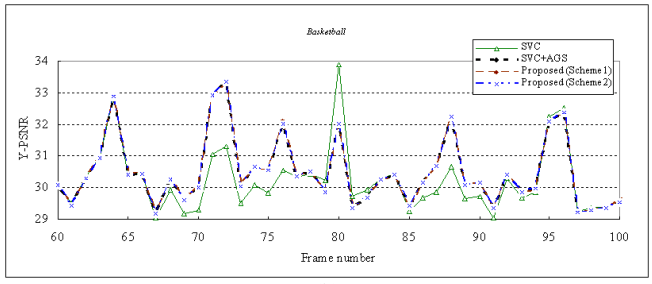
In our experiments, we use the SVC reference software JSVM 2.0 [9] to implement the proposed fast mode decision algorithm. Four CIF (352x288) test sequences, *Football*, *Foreman*, *Stefan*, and *Basketball*, are used in our experiments. For each test sequence, the GOP size of 16 frames is used and the update step is turned on in MCTF. The test conditions of Palma-CE1-Conditions in the SVC reference software are used in our experiments. Scheme 1 uses the average MV magnitude of the highest layer and the number of intra-coded macroblocks of the high-pass frame at the lowest layer of MCTF with the full GOP size as the features, whereas, in Scheme 2, the total number of intra-coded macroblocks is computed from the high-pass frames of all layers of MCTF. The threshold of average MV magnitude is 10 and the thresholds of number of intra-coded macroblocks are 50 and 60 for Scheme 1 and Scheme 2, respectively. The same thresholds are applied to all the sequences.

Table 1 Coding time (sec) comparison and time savings of the proposed methods

Sequences	MCTF	MCTF+AGS	Scheme 1	Scheme 2
Football	1268	4406	2725 (38%)	3449 (22%)
Foreman	927	3435	1140 (67%)	1144 (67%)
Stefan	974	3629	1193 (67%)	1197 (67%)
Basketball	685	2732	1590 (42%)	1973 (28%)



(a)



(b)

Fig. 6. Frame-by-frame PSNR performance comparisons using the original MCTF, MCTF with AGS, and two fast algorithms for AGS: (a) *Football*, (b) *Basketball* (CIF30).

Table 1 compares the coding times and the time savings using Scheme 1 and Scheme 2. The numbers in the first and second columns show the coding times of MCTF without and with AGS, respectively. In the third and fourth columns, the coding times and the percentages of time saving of Scheme 1 and Scheme 2 are listed, respectively. The time saving of Scheme 1 ranges from 38% to 67%, while Scheme 2 achieves the time saving of 22~67%. Scheme 2 achieves less time savings for the *Football* and *Basketball* sequences compared with Scheme 1, because Scheme 2 increases the number of GOPs that are coded with the "Sub-GOP" mode that require more MCTF operations of sub-GOPs.

Fig. 6 shows the frame-by-frame PSNR performance comparisons of MCTF (denoted as "SVC"), MCTF with AGS (denoted as "SVC+AGS"), Scheme 1, and Scheme 2 at the highest bit-rate of CIF30. We can observe that AGS achieves significant coding efficiency improvement in many frames compared to the original SVC. With Scheme 1, some frames that should be encoded with the "Sub-GOP" mode may be erroneously encoded with "Full-sized GOP" mode, leading to PSNR drops of some GOPs compared to the original AGS. Scheme 2 can avoid those PSNR drops due to the increased precision of the feature, the number of intra mode macroblocks. Overall, our proposed schemes retain the coding efficiency of AGS, while reducing the computational complexity significantly.

IV. Conclusion

In this paper, we proposed a fast GOP mode selection algorithm to reduce the coding complexity for AGS. We proposed to select the average MV magnitude and the number of intra-coded macroblocks as features to capture the temporal characteristics of a GOP. These features are utilized to decide whether the encoder should perform the following MCTF procedures with sub-GOPs after performing the first MCTF procedure with full-sized GOP. As a result, the proposed method avoids unnecessary MCTF procedures of sub-GOPs so as to reduce coding complexity. Our experimental results show that the proposed method achieves significant saving of AGS coding time with slight PSNR performance degradation.

References

[1] S.-F. Chang and A. Vetro, "Video adaptation: Concepts, technologies, and open issues," *Proc. IEEE*, vol. 93, no. 1, pp. 148-158, Jan. 2005.

[2] J. Xin, C.-W. Lin, and M.-T. Sun, "Digital video transcoding," *Proc. IEEE*, vol. 93, no. 1, pp. 84-97, Jan. 2005.

[3] J. R. Ohm, "Advances in scalable video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 42-56, Jan. 2005.

[4] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable H.264/MPEG4-AVC extension," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2006.

[5] ITU-T and ISO/IEC JTC1, "Joint Draft 8: Scalable Video Coding," JVT-U201, Oct. 2006.

[6] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Processing,* vol.3, no.5, pp.559-571, Sept. 1994.

[7] G. H. Park, M.W. Park, S Jeong, K. Kim, and J. Hong, "Improve SVC Coding Efficiency by Adaptive GOP Structure (SVC CE2)," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT-O018, 2005.

[8] G. H. Park, M.W. Park, S Jeong, J. Cha, K. Kim, and J. Hong, "Adaptive GOP Structure for SVC", ISO/IEC JTC1/SC29/WG11, M11563, 2005.

[9] ITU-T and ISO/IEC JTC1, "Joint Scalable Video Model JSVM 2," JVT-O202, Apr. 2005.