AN ENHANCED RATE CONTROL SCHEME WITH MOTION ASSISTED SLICE GROUPING FOR LOW BIT RATE CODING IN H.264

Avin Kumar Kannur and Baoxin Li

Department of Compute Science and Engineering
Arizona State University, Tempe, Arizona, 85281
Email: {akannur, baoxin.li} @asu.edu

ABSTRACT

This paper presents an enhanced rate control scheme for H.264 using motion detection and motion analysis. PSNR based measure is used for determining motion complexity and detecting scene changes. Motion Vectors and Rate Distortion measures are further used to classify regions with motion inside a frame and Macro-blocks are then grouped into slices using the Flexible Macro-block Ordering feature of H.264. We present a filtered slice grouping technique to maintain a stable region of motion. The proposed method provides smoother quality videos with better average PSNR. The bit rates are within bound and fewer frames are skipped in sequences with scene changes at low bit rates.

Index Terms – Flexible Macro-block Ordering (FMO), Filtered Slice Grouping, Scene change, Frame Skipping.

1. INTRODUCTION

H.264/AVC is the latest video coding standard proposed by ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group (MPEG) and supports a wide variety of video applications with varying bit rate constraints. While there are many advanced rate control schemes for H.264/AVC, there is still a need for efficient rate control schemes which can handle exceptions such as scene changes (regions where conventional flow of rate control is broken) in low bit rate applications in order to maintain the smoothness of quality in delivering the video. The rate control scheme of [1], which is based on PSNR and MAD measures for rate control, is along such a direction.

In addition to handling the irregularity of contents along the temporal axis as in the case of scene change, there is also a possibility of allocating different number of bits within a frame to regions of different properties (such as treating static regions differently from regions of moving objects). These types of techniques typically rely on a segmentation step as preprocessing, which may be done efficiently in the compressed domain (e.g., [2]). A motion assisted rate control based on pre-filtered and global motion compensated Motion Vectors (MV) is presented in [3] which is found to enhance the subjective quality of regions with high visual sensitivity. FMO [4] is a unique tool set available in H.264 and is primarily used for error resilience and error concealment video coding [5]. Not limiting to this, it is also shown to be a handy feature for coding arbitrary-shaped Region Of Interest (ROI) [6] [7]. In such methods, the picture is divided into a foreground region with multiple or single slice groups and a background region (often static or with no or less motion contents).

In this paper we present a rate control scheme for low bit rate applications based on an existing frame layer bit allocation scheme [1] and a novel ROI classification technique for grouping macro-blocks (MB) into slice groups. The grouping is based on the motion analysis using MV-based classification, which is further coupled with a spatial object mask derived from change detection. This results in two rate control schemes for H.264, which are evaluated and compared with a set of standard test videos.

The proposed rate control schemes are presented in Section 2. Experimental results are documented in Section 3, and we draw conclusions from our work in Section 4.

2. MV ASSISTED SLICE GROUPING FOR FMO BASED RATE CONTROL

Conventional rate control algorithms are not effective in addressing scene changes, which result in abrupt variation of signal statistics along the temporal axis. Schemes without considering such abrupt variation often result in buffer overflow especially at low bit-rates, due to the fact that most MB in a scene change are typically intra-coded. In these cases the computed QP value is no longer suitable and it must be adjusted so that the coming frames are not penalized during bit allocation. An enhanced rate control scheme using PSNR based frame complexity measure was presented in [1] which we incorporate in our rate control. In this approach, scene changes are detected based on the derived PSNR ratios and are used to serve as signaling function for bit allocation at scene changes. Bits are allocated in the frame layer based on the frame complexity, taking into account the current buffer status and bits remaining for encoding the remaining frames of the Group of Picture (GOP).

Furthermore, since MV represents the motion dynamics within a frame, we group the MBs within a frame into motion regions based on the MVs: those corresponding to moving regions and those corresponding to static regions. The moving regions may be further classified into multiple regions depending on the motion magnitude. Upon the categorization of the MBs, we utilize the "explicit slice group ordering" option in FMO to define slice groups (SG), with different slices corresponding to groups of MBs with different motion properties. With such slice grouping, we can code the slice groups with different quality. For example, in applications for which we believe that moving regions correspond to objects of higher significance, we can code the slice groups with high motion MB finely while coding others with low motion MB relatively coarsely. In the following subsections, we detail our algorithmic components for implementing the idea.

2.1. Slice Group Ordering

We assume that neighboring frames differ not significantly (smooth video) and hence the MV's of the previous encoded frame are used to determine the regions of motion in the next frame. This enables our algorithm to be implemented within a single pass. We use the magnitudes of the x and y components of the MV's in our analysis. The MV can be preprocessed to account for global motion [2,3]. In an MB that is coded with multiple coding sub-blocks, we take the maximum value of the MV components to determine the component values of MB, as shown below.

$$\begin{aligned} MV_{\max}(i,x) &= \max(MV(i,j,x)) \\ MV_{\max}(i,y) &= \max(MV(i,j,y)) \end{aligned} \tag{1}$$

where $MV_{\max}(i,x)$ and $MV_{\max}(i,y)$ are the maximum MV for the i^{th} MB in x and y directions respectively and j is a MB sub partition index. We calculate the mean MV in the neighborhood of the i^{th} MB using

$$E\left[MV_{i,x}\right] = \frac{\sum_k MV_{\max}(k,x)}{N_{\text{int}\,er}} \quad k \in \textit{Neighborhood of } i \tag{2}$$

$$E\left[MV_{i,y}\right] = \frac{\sum_k MV_{\max}(k,y)}{N_{\text{int}\,er}} \quad k \in \textit{Neighborhood of } i \tag{3}$$

where N_{inter} is the number of MBs coded in inter prediction mode in the neighborhood region of i^{th} MB. The slice grouping label for the i^{th} MB is determined as (for a two-category example)

$$\mathrm{L}_{\mathrm{i}} = \begin{cases} \mathrm{SG1, if}\left(MV_{i,x} > \mathrm{E}[MV_{i,x}]\right) or \left(MV_{i,y} > \mathrm{E}[MV_{i,y}]\right) \\ \mathrm{SG0, otherwise} \end{cases} \tag{4}$$

This basic technique, while being intuitive, is not effective in practice, and thus two schemes are proposed below to enhance this basic idea.

2.1.1. Scheme 1: Filtered Slice Group Ordering

We found through experiments that, updating the slice group mapping of the MB for each frame would result in rapid changes in Region of Motion (ROM). To ensure a relatively stable ROM, we update the slice groups only periodically at a predefined rate (e.g., every 4th frame). Considering that an MB might have moved before the next update of its slice grouping label, we also label the neighboring MBs of a high-motion MV to be SG1. This helps to preserve motion in a small region in between the two update points. Furthermore, in order to ensure that regions with large distortion but small motion (such as textured regions) are allocated sufficient bits, we also incorporate the distortion measure in the slice grouping scheme as follows.

We first calculate the mean distortion E[D] using ,

$$E[D] = \frac{\sum_{k=1}^{N} RD\cos t_{\min}(k)}{N} \tag{5}$$

where $RD\text{cost}_{\min}$ is the rate distortion cost equation used by the encoder to choose the optimal mode for encoding an MB with the least cost, N is the total number of MB coded in inter mode. Then we compute slice grouping label using the following filter

$$X_i = \begin{cases} 1, if \begin{pmatrix} (MV_{i,x} > E[MV_{i,x}]) or (MV_{i,y} > E[MV_{i,y}]) \\ or (RD\cos t_{\min}(i) > E[D]) \end{pmatrix} \\ \qquad 0, Otherwise \end{cases} \tag{6}$$

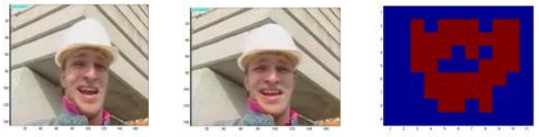
$$Y_i = \begin{cases} \alpha_0 * X_i + (1-\alpha_0) * Y_{i-1}; \text{if}\,(X_i = 1) \\ \beta_0 * Y_{i-1}; \qquad \text{otherwise} \end{cases} \tag{7}$$

where X_i is the filter input, Y_i is the filter output for the i^{th} MB, α_0 and β_0 are filter coefficients with values 0.15 and 0.95 respectively. α_0 is a measure of how fast the system adapts to a change in ROM and β_0 is a measure of how long it retains an updated ROM.

We then use the updated filter output value to compute the slice group mapping.

$$L_i = \begin{cases} SG1, if\,(Y_i > Y_{threshold}) \\ SG0, \quad \text{otherwise} \end{cases} \tag{8}$$

$Y_{threshold}$ is set to 0.5 in our experiments. One sample result of this slice grouping algorithm is shown in Fig 2.1. We see that this scheme may result in fragmented object segmentation. An alternative approach is discussed next to alleviate this issue.



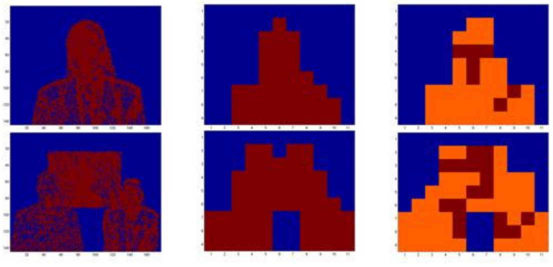
(a) (b) (c)

Fig 2.1. (a) Frame 19, (b) Frame 20 of Foreman Sequence, (c) Slice Grouping – **SG 0**, **SG 1**.

2.1.2 Scheme 2: Filtered Slice Group Ordering with Object Mask

MV based slice grouping can lead to fragmented MB slicing and may not be able to capture the entire object of interest, but only regions of motion within it. We need an object mask which can track the entire object (especially in sequences like video conferencing where motion can be within an object which occupies significant portion of the frame spatially). We incorporate a spatial domain object mask generation using the Change Detection Mask presented in [8]. We then apply our Scheme 1 to further enhance the motion details within these objects. We now group the MB into 3 slice groups, SG0 for static background, SG1 for static parts within the object and

SG 2 for motion parts within the object. Some sample slice grouping results are shown in Fig 2.2.



(a) (b) (c)

Fig 2.2. (a) Change Detection Masks, (b) Object Masks generated for Akiyo and News sequences, (c) Slice Grouping, **SG 0**, **SG 1**, **SG 2**.

2.2. Selection of QP values for the Slice Groups

We choose different QP values for the slice groups so as to assign more bits to regions with high motion and distortion contents (SG2 and SG1) and fewer bits to low motion region (SG0). A simple way of achieving this is

$$QP_{i,SG(j)} = \begin{cases} QP_i - 2, SG(2) \\ QP_i - 1, SG(1) \\ \min(QP_i + floor(QP_i / \alpha) + \beta, 51), SG(0) \end{cases} \quad (9)$$

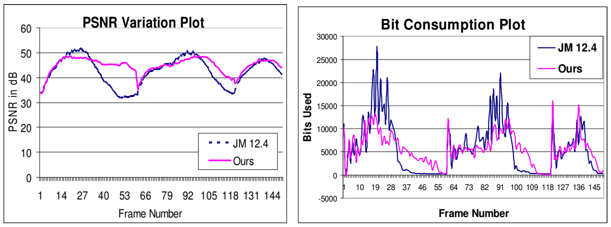
where QP_i is the quantization parameter for the i^{th} frame, and $QP_{i,SG(j)}$ is the refined QP for the i^{th} frame and j^{th} slice group. α and β are constants with values 8 and 2 respectively.

It was found that the selection of QP for the first I frame of the sequence is very critical and an improper QP will break the rate control scheme. With our experiments we found that an initial QP of 28 works well for most sequences. We can further refine this by taking into account the GOP size and number of bits available per picture.

2.3. Scene Change Handling and Frame Skipping

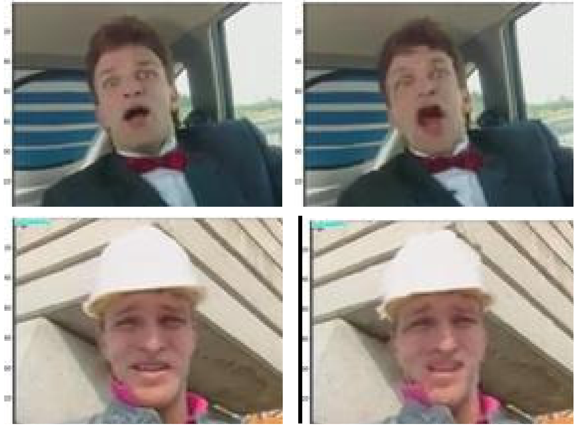
When a scene change is detected, we cannot use the above filtered slice grouping scheme as all the MB are new. In this case, we update the slice group mapping on scene changes since we cannot use the slice group map of the previous iteration. We use a default rectangular window at the center for slice grouping, which is a reasonable approach since for most deliberately captured video; the region of interest is typically located at the center of a frame. In subsequent iterations the slice grouping will be updated as described above.

A common problem noticed in low bit rate applications with scene changes is frame skipping as the encoder will skip frames if the buffer has reached the safety margin. Frames will be skipped irrespective of their importance and thus it is very likely that frames with high motion (which may be significant in terms of their semantics) are skipped, resulting in a perceptual loss of video quality. Instead of skipping frames after the safety margin has reached, we proactively skip very low complexity frames (details in next section) which are very effectively similar to the previously encoded frames. This proactive skipping of frames provides noticeable benefits as will be seen in results listed in the next section.



(a) (b)

Fig 3.1 (a) PSNR Variation and (b) Bit Consumption, for Akiyo sequence coded at 15fps, 84kbps. (**JM** , **Proposed**)



(a) (b)

Fig 3.2. Subjective video quality (a) Proposed; (b) JM 12.4.

3. EXPERIMENTAL RESULTS

We have implemented the proposed rate control algorithms based on the JM 12.4 H.264 video codec. For our tests we used the encoder in Base Profile, RDO enabled, search range for motion estimation is set to 16, number of reference frame as 1, entropy coding technique to be Context Adaptive Variable Length Coding (CAVLC). Standard test sequences in QCIF 4:2:0 formats were taken for testing our rate control schemes. GOP size was set to 60 and frame rate to be 15fps (except for the Claire and the Carphone sequences for Scheme 2, where the frame rate is 20 and 10 fps respectively). We compare our proposed rate control schemes with JM 12.4 quadratic rate control scheme.

Fig. 3.1 gives one sample result for the PSNR variations and bit allocations for the Akiyo sequence. It was found that the PSNR variation is much smoother in the proposed method and better visual quality is achieved. The bit allocation is done on the frame complexity basis and we see in 3.1(b) that we have better bit allocation for frames at the end of GOP (the sequence contains 3 GOPS). More results are tabulated in Table 3.1, where for comparative convenience of both the schemes, we have included a few identical test cases (e.g., Akiyo at about 84kbps). The table also contains a couple of sequences that consist of multiple concatenated sequences (e.g., Akiyo-Carphone) which were intended to simulate scene changes. In these concatenated sequences, we repeat the component sequences to introduce four scene changes.

While Table 3.1 presents the comparison in terms of PSNR, Fig 3.2 illustrates the subjective video quality of the proposed approach in comparison with that of the JM 12.4 version. It is obvious that we gain subjective video quality by allocating more bits for regions with motion. Fig 3.3 shows the RD performance curves for two test sequences (Akiyo and News), which show that for a range of low bitrates (above 32kbps), the proposed schemes outperform the JM coder. (For even lower rates, the overhead of our scheme due to slice mapping etc will offset its benefit and thus the performance gain diminishes.)

We also found that the proposed method of skipping frames on a proactive basis achieves better result than JM. For our experiments we set 80% as the buffer fullness limit for JM. In our method we skip frames when the buffer fullness crosses 65% limit and the PSNR drop is less than 0.45 dB. As a result we saw our method skips only 5 frames when compared to 12 frames by JM in the salesman sequence. In case of Akiyo-carphone sequence the frames dropped are 3 and 19 in the proposed and the JM versions respectively.

4. CONCLUSION

We have presented a rate control method based on slice grouping using motion-analysis-based MB categorization. When coupled with the frame layer bit allocation scheme, this leads to an efficient rate control algorithm for low bit-rate applications of H.264. We evaluated the proposed approach and compared with the quadratic rate control scheme of JM 12.4, using a set of standard test videos. Results have shown the effectiveness and advantages of the proposed method.

5. REFERENCES

[1] Minqiang Jiang and Nam Ling, "On Enhancing H.264/AVC Video Rate Control by PSNR-Based Frame Complexity Estimation", IEEE Transactions on Consumer Electronics, 284 Vol. 51, No. 1, February 2005.

[2] Zhi Liu, Yu Lu, and Zhaoyang Zhang, "Real-time spatiotemporal segmentation of video objects in the H.264 compressed domain", *Journal of Visual Communication and Image Representation*, v 18, n 3, 275-90, June 2007.

[3] Gounyoung Kim, Eleftheriadis, A.; "Motion Vector Field Analysis in Motion-Assisted Rate Control for H.264"**, *ICIP*,** Page(s):61-64, 8-11 Oct. 2006.

[4] Y. Dhondt and P. Lambert, "Flexible Macroblock Ordering: an error resilience tool in H.264/AVC," in *Fifth FTW PhD Symposium,* no. 106, Dec. 2004.

[5] Sood, Amit; Chilamkurti, Naveen K; and Soh, Ben, "Study and analysis of an error resilient technique in H.264 video using flexible macroblock grouping," *WiCOM 2006*, 2007, p 4149485.

[6] Lambert, Peter; De Schrijver, Davy; Van Deursen, Davy; De Neve, Wesley; Dhondt, Yves; Van De Walle, , "A real-time content adaptation framework for exploiting ROI scalability in H.264/AVC" *ACIVS Proceedings,* p 442-453, *2006.*

[7] P. Sivanantharasa', W.A.C. Fernando, SMIEEE, H. Kodikara Arachchi, "Region of Interest Video Coding with Flexible Macroblock Ordering", First International Conference on Industrial and Information Systems, ICIIS 2006, 8 – 11 August 2006.

[8] J. Y. Zhou, Ee Ping Ong, Chi Chung Ko, "Video Object Segmentation and Tracking for Content-Based Video Coding", *ICME (III)*, 1555-1558, August 2000.

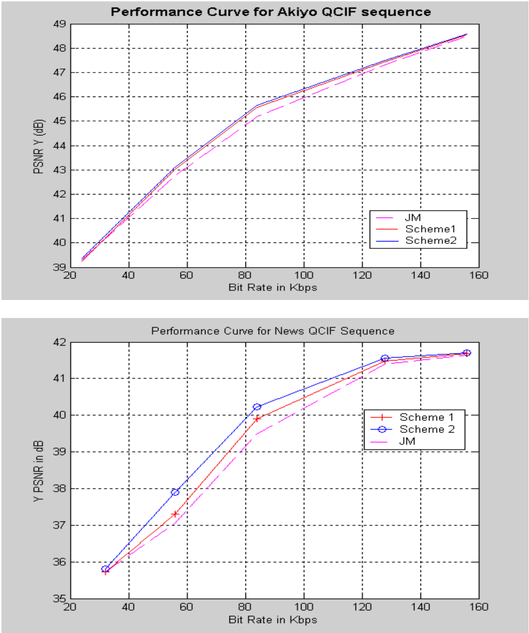


Fig 3.3. Performance Curves (PSNR v/s Bit Rate)

Table 3.1. PSNR and Bit Rate comparative measures.

Test Sequence	*Average PSNR (dB)*			*Bit Rates (kbps)*	
(QCIF, 4:2:0)	JM	**Scheme 1**	Gain	JM	**Scheme 1**
Akiyo	45.19	45.55	0.36	84.24	84.1
Salesman	37.20	37.38	0.18	56.54	56.07
Foreman	35.45	35.57	0.12	64.26	64.06
Akiyo-Carphone	37.29	37.94	0.65	64.08	63.91
Foreman - News	38.92	39.41	0.49	96.7	96.37
Container	38.86	39.1	0.24	48.01	48.23
	JM	**Scheme 2**	Gain	JM	**Scheme 2**
Akiyo	45.19	45.65	0.46	84.24	84.25
Salesman	37.20	37.53	0.33	56.54	56.62
Foreman	35.45	35.84	0.39	64.26	64.16
News	39.07	39.77	0.7	77.35	76.33
Claire	45.4	45.73	0.33	76.39	76.53
Carphone	40.79	40.96	0.17	84.18	84.05