BUFFER CONSTRAINED RATE CONTROL FOR LOW BITRATE DUAL-FRAME VIDEO CODING

Mayank Tiwari[1], *Theodore Groves*[2], *and Pamela Cosman*[1]

[1]Department of Electrical and Computer Engineering, [2] Department of Economics,
University of California, San Diego, La Jolla, CA 92093-0407

ABSTRACT

In dual-frame video coding, one long-term reference (LTR) and one short-term reference (STR) frames are used for motion estimation and compensation. In previous work, it was shown that the performance of video coding can be improved by pulsing the quality of LTR frames in dual-frame video coding, but this increases the encoder delay buffer size. Also, buffer constrained real-time video transmission requires an efficient rate control algorithm to meet the delay requirement. In this paper, we propose a rate control algorithm for dual-frame video coding under a delay buffer constraint. With the proposed rate control algorithm and motion activity detection for determining the LTR quality, simulation results using H.264/AVC show a significant PSNR improvement over H.264 rate control and other rate control algorithms for dual-frame video coding.

***Index Terms*—** H.264/AVC, Dual-frame buffer, video compression, multiple frame prediction.

1. INTRODUCTION

In multiple frame prediction, more than one past frame is used in the search for the best match block. This has been shown to provide a clear advantage in compression performance [1, 2]. This concept was adopted in the H.264/AVC [3] video coding standard that allows up to 16 reference frames for motion estimation. A further improvement in video coding is achieved if the reference frames are temporally separated [4]. In dual-frame video coding, two frames are used for inter prediction, a short-term reference (STR) and a long-term reference (LTR), as shown in Figure 1. Both encoder and decoder store LTR and STR frames. For encoding frame n, the STR is frame $n-1$ and the LTR is frame $n-k$, for $k > 1$. The LTR frame can be chosen by jump updating, in which the LTR frame remains the same for encoding N frames, then jumps forward by N frames and again remains the same for encoding the next N frames. In jump updating, every frame serves as an STR, but only every N^{th} frame serves as an LTR; this allows the use of high quality LTRs, where the LTR frames are allocated more bits than regular frames. This has been shown to enhance the quality of the entire sequence [5]. While a performance improvement is achieved using dual-frame video coding, a price is paid in larger delay buffer to accommodate the high quality LTR frames. In real-time video transmission, the amount of delay is limited. For example, the maximum delay that can be tolerated in video telephony is less than 300ms. Therefore, high quality LTR frames may pose a challenge for using dual-frame coding in real-time video transmission.

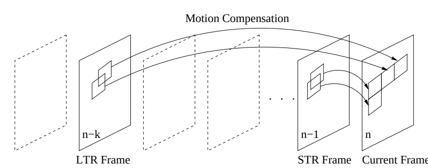


Fig. 1. Dual-frame video coding.

Rate control for video coding has been extensively studied [6, 7]. A buffer constrained rate control algorithm for H.264 in [7] uses a pre-analysis unit to accurately achieve the target bitrate. However, rate control for dual-frame video coding is largely untouched. In particular, assignment of high quality to LTR frames presents difficulties for rate control. One can reserve a portion of the buffer to accommodate LTR frames but this reduces the buffer usage for other frames. In this paper, we examine a rate control method for dual-frame video coding with a delay buffer constraint. This work is inspired by the rate control used in [8] where separate rate control was implemented for both regular and high quality LTR frames.

In this paper, we propose a buffer threshold strategy to accommodate large size LTR frames. For reducing loss due to buffer overflow, we use a buffer threshold for quantization parameter (QP) adjustment that limits the buffer usage. We use a motion activity detection algorithm to determine the number of bits for a high quality LTR frame. The proposed method outperforms the standard H.264 rate control [9] and the rate control for dual-frame video coding proposed in [8] even when a modification for reducing the loss due to buffer overflow is incorporated in both these methods. Note that the typical H.264 rate control algorithms do not perform well be-

This work was supported by the National Science Foundation, the Center for Wireless Communications at UCSD, and the UC Discovery Grant program of the State of California.

cause those algorithms are not designed to handle the extra bits for the LTR frames in dual-frame video coding.

The rest of the paper is organized as follows: Section 2 describes the delay associated with video encoding. Section 3 discusses the rate control method for dual-frame video coding using the motion activity detection algorithm. Simulation results and conclusions are given in Section 4.

2. VIDEO ENCODING DELAY

Delay at the encoder comes from input buffer delay, encoder processing delay, and output buffer delay as shown in Figure 2. We use IPPP coding format, so all frames are processed sequentially, and there is a constant input buffer delay of one frame. The processing delay is platform dependent and, for the purpose of rate control, we ignore this delay. The encoder generates a variable size of encoded bitstream for each frame while we assume transmission at constant bit rate. Therefore, we need to store bits in an encoder output buffer.

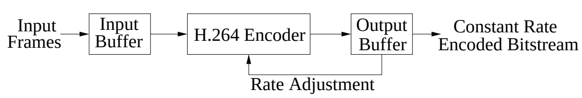


Fig. 2. Delay components at the video encoder.

Let R be the transmission rate and let the video be encoded at f frames per second. If each frame is encoded with $\frac{R}{f}$ bits then we do not need any encoder output buffer. However, this leads to a very poor video quality since not all frames require the same number of bits. In practical scenarios, frames are assigned bits based on their relative complexities. Frame complexity is often estimated using mean absolute difference (MAD) which is the difference between the original frame and the predicted frame. Since the current frame is not yet encoded, the H.264 rate control [9] algorithm predicts the current frame MAD from the previously encoded frame MAD. A more accurate MAD prediction was proposed in [10]. For a given target rate for the current frame and MAD, the QP is calculated using a quadratic rate-distortion (R-D) model. However, it is difficult to predict the exact QP that will produce the encoded bits for a frame close to its target bits. This leads to the requirement of having an encoder output buffer that can convert variable encoder output rate to constant rate for transmission. With a buffer constraint, the frames (or part of a frame) that exceed the buffer limit are dropped. This leads to error propagation and video quality deterioration.

3. RATE CONTROL FOR DUAL-FRAME VIDEO CODING

With dual-frame coding where many bits are assigned to a LTR frames, the chances of a portion of LTR frames getting dropped is high. This also happens in the rate control implementation given in [8] where two separate rate control paths, one each for the regular and LTR frames, were used by the encoder. The rate for LTR frames was assumed to be three times that of regular frames. The rate control in H.264/AVC was used for both paths and the quality improvement as a function of encoder output buffer size was shown over video coding with two STR frames. Since the LTR frames are encoded with separate rate control, there may be cases where the quality of an LTR and its adjacent regular frames is similar, thus losing the importance of an LTR frame. This method uses the SKIP mode to drop macroblocks (MBs) in case a frame would cause a buffer overflow. While this rate control method works well for high rate, the LTR frames in low rate coding suffer many MB drops due to their large size.

In our approach for rate control using a buffer, we try to keep the buffer fullness at a predetermined fraction (bf_low) of total buffer size. We slowly increase the QP if the buffer level crosses bf_low to avoid MB losses. Otherwise, we let the H.264 R-D optimization determine the QP. If an LTR frame comes then we increase the buffer fullness threshold to a higher level (bf_high) because LTR frames are assigned more bits. After the LTR frame, we slowly reduce the threshold from bf_high to bf_low within bf_slope frames. This process is shown in Figure 3. We use the H.264 rate control if the buffer fullness is within the specified buffer fullness level otherwise we increase the QP. Note that bf_high, bf_low, and bf_slope are determined experimentally using a set of training sequences. We have not yet tried to optimize these values for any particular type of sequence.

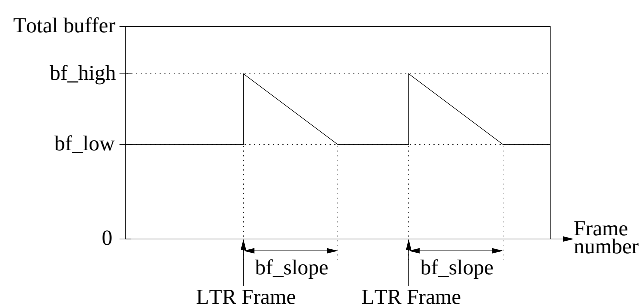


Fig. 3. Target buffer level for the proposed rate control algorithm for dual-frame video coding.

With this single rate control path to accommodate both regular and LTR frames, the LTR frames are of higher quality than adjacent frames yet we seldom need to skip MBs to avoid buffer overflow. The number of bits and the QP for a high quality LTR frame are determined by motion activity as discussed below.

Motion activity detection and rate allocation: In dual-frame coding, a key issue is to allocate an appropriate number of bits to ensure a high quality LTR frame. For a low motion video, we allocate many bits to the LTR frame since the quality of subsequent frames will also be high as they are similar to the LTR frame. For a high motion video, it is not desirable to spend many bits on an LTR frame because its higher quality will soon be lost as the subsequent frames rapidly become different from the LTR. Motion activity detection and rate allocation for LTR frames was proposed in [11] and is briefly described here.

The motion of a video sequence is detected by comparing the current frame with the previous frame. We calculate the sum of absolute differences (SAD) between each MB and the co-located MB in the previous frame. The MB is considered active if the SAD is above a predetermined threshold (500), otherwise it is considered inactive. The bit allocation for the LTR frame is given by

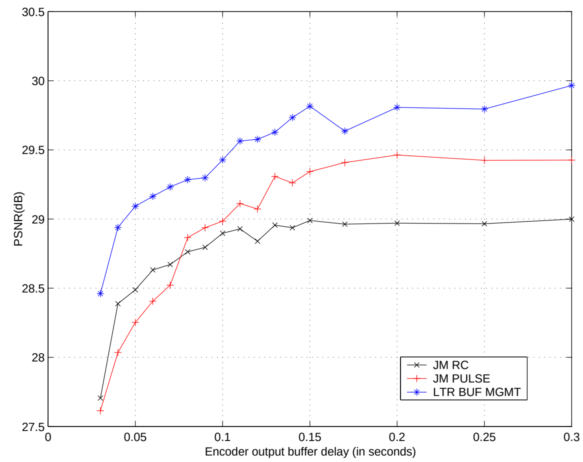
$$LTR\_bits = \begin{cases} 2 \times reg\_bits, & \text{if } m > 0.5 \\ 10 \times reg\_bits, & \text{if } m < 0.1 \\ (12 - 20 \times m) \times reg\_bits, & otherwise \end{cases} \quad (1)$$

where m is the average fraction of active MBs in the 10 frames prior to an LTR frame and reg_bits is the average number of bits assigned to a regular frame. LTR_bits are also upper-bounded by the remaining space in the encoder buffer. The encoded bitstream is standard compatible.
To make an unbiased comparison, we modified the rate control algorithm in H.264/AVC by maintaining a buffer threshold in the encoder output buffer. If the buffer fullness exceed this threshold, we adjusts the QP to reduce MB losses. Similar modification is also applied in the rate control proposed in [8]. These improved rate control algorithms are then compared with the rate control proposed in this paper.

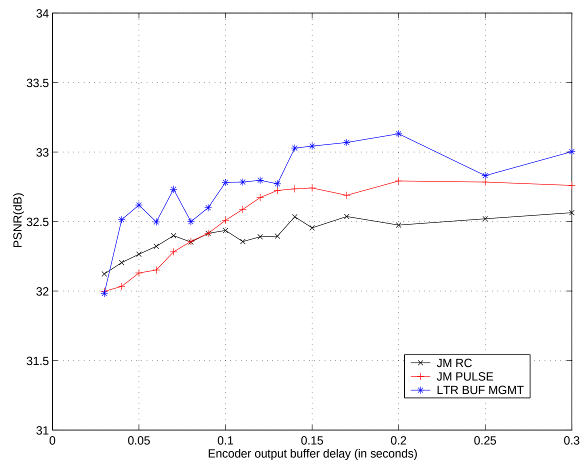
4. RESULTS

The simulation was performed using JM 10.1 [12] reference software for H.264/AVC baseline profile. All the video sequences used in the simulation were 300 frames QCIF (176×144 pixels) at 30 fps. The distance between two LTR frames was 25 frames. We calculate the average MSE for each frame and across all frames of a video sequence, and then convert to PSNR for reporting our results.

For H.264 rate control with two STR frames (**JM RC**) and the rate control used in [8] (**JM PULSE**), we keep the encoder output buffer threshold at 50%. This means we start adjusting the QP at this threshold for avoiding MB losses in the frame. If the frame size exceeds the buffer size, then we drop MBs using the skip mode. The skipped MBs are reconstructed using motion compensated prediction from the STR frame where neighboring motion vectors are used to find the motion vector of the lost MB. In our work (**LTR BUF MGMT**), we keep bf_low at 40% and bf_high at 65% of the total buffer size. The bf_slope is 15 frames for the LTR distance of 25 frames.



(a)



(b)

Fig. 4. Variation of PSNR with the encoder output buffer delay (in seconds) for (a) News and (b) Container video sequence at 18kbps.

Figure 4 shows the variation of PSNR with the encoder output buffer delay for News and Container videos at 18kbps. In both videos, JM RC performs better than JM PULSE for smaller encoder buffer size because the chances of loss in the LTR frames due to buffer overflow are very high. Even with the QP adjustments, sometimes it is not possible to avoid MB loss. As expected, JM PULSE performs better than JM RC at larger encoder buffer size due to the advantage of high quality LTR frames over the two STR frames. By appropriately managing the buffer usage for LTR frames, MB losses are further reduced in LTR BUF MGMT thus improving the performance over JM PULSE. The performance is further boosted by choosing the appropriate number of bits for high quality LTR frames based on motion activity level instead of using some average number of bits. Therefore, LTR BUF MGMT outperforms both other methods at all encoder buffer sizes. The effect of high quality LTR frames can be seen from Fig-

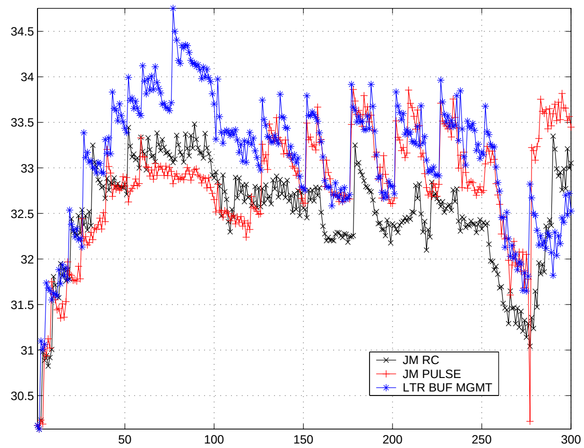


Fig. 5. PSNR fluctuation with frame number for Container video at 18kbps and 0.15 seconds encoder output buffer delay

ure 5. The LTR BUF MGMT curve is almost always above the other two curves. The pulsing of LTR frames is not perceptually visible. Similar results were found for Akiyo and Container videos at 36kbps (the latter is shown in Figure 6).

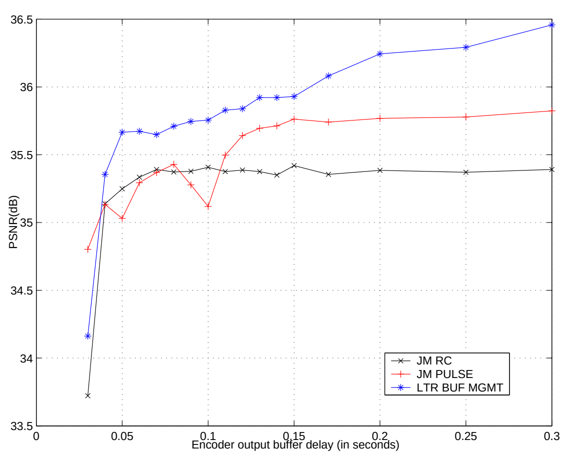


Fig. 6. Variation of PSNR with the encoder output buffer delay (in seconds) for Container video sequence at 36kbps.

Conclusion: we proposed a rate control method for handling high quality LTR frames in dual-frame video coding for buffer constrained real-time video communication. The method uses a motion activity detection algorithm to appropriately allocate bits to an LTR frame. The proposed method outperforms H.264 rate control and a previously proposed rate control method for dual-frame video coding, even when these methods are modified to reduce the MB losses in a frame. The rate control algorithms were studied at low rates as the importance of high quality LTR frames fades away at high rates. We studied low motion video sequences in this paper since dual-frame video coding does not provide significant gain for high motion video sequences. The buffer level and bit allocation for LTR frames can be optimized for a particular video sequence to further improve the performance.

5. REFERENCES

[1] M. Gothe and J. Vaisey, "Improving motion compensation using multiple temporal frames," *Proc. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, vol. 1, pp. 157–160, May 1993.

[2] M. Budagavi and J. D. Gibson, "Multiframe block motion compensated video coding for wireless channel," *Proc. IEEE Asilomar conference on Signals, Systems and Computers*, pp. 953–957, Nov. 1996.

[3] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[4] T. Fukuhara, K. Asai, and T. Murakami, "Very low bit-rate video coding with block partitioning and adaptive selection of two time-differential frame memories," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 212–220, Feb. 1997.

[5] V. Chellappa, P. Cosman, and G. Voelker, "Dual-frame motion compensation with uneven quality assignment," *Proc. IEEE DCC*, pp. 262–271, Mar. 2004.

[6] F. Tonci, C. Adsumilli, M. Carli, A. Neri, and S. Mitra, "Buffer constraints for rate-distortion optimization in mobile video communications," *Proc. IEEE Int. Symposium on Signals, Cir., and Sys.*, pp. 71–74, July 2005.

[7] P. Li, X. Yang, and W. Lin, "Buffer-constrained R-D model-based rate control for H.264/AVC," *Proc. IEEE ICASSP*, pp. 321–324, Mar. 2005.

[8] A. Leontaris and P. C. Cosman, "Compression efficiency and delay tradeoffs for hierarchical B-pictures and pulsed-quality frames," *IEEE Trans. on Image Processing*, vol. 16, no. 7, pp. 1726–1740, July 2007.

[9] K.-P. Lim, G. Sullivan, and T. Wiegand, "Text description of joint model reference encoding methods and decoding concealment methods," *JVT of ISO/IEC MPEG and ITU-T VCEG, JVT-K049*, Mar. 2004.

[10] M. Jiang and N. Ling, "Low-delay rate control for realtime H.264/AVC video coding," *IEEE Transactions on Multimedia*, vol. 9, no. 3, pp. 467–477, June 2006.

[11] M. Tiwari, T. Groves, and P. Cosman, "Multiplexing video streams using dual-frame video coding," *Proc. IEEE ICASSP*, pp. 693 – 696, Mar. 2008.

[12] "H.264/AVC ref. software," http://iphome.hhi.de/suehring/tml.