MULTI-RATE ENCODING OF A VIDEO SEQUENCE IN THE DCT DOMAIN*

André Zaccarin[1] and Boon-Lock Yeo[2]

(1) Dept. of Electrical and Computer Engr., Université Laval,Québec, Qc., G1K 7P4 Canada
(2)Swarm Networks Inc., 10181 Bubb Road,Cupertino, CA, 95014 USA
zaccarin@gel.ulaval.ca , yeo@swarm.com

ABSTRACT

In today streaming servers, video sequences are offered to users at different fixed bit rates. This paper presents an efficient approach for simultaneous encoding of a video sequence at multiple bit rates. In this encoder, motion estimation is performed only once for the reference stream. We also take the DCT out of the encoding loop so that it is only computed once per frame. As a result, no iDCT has to be computed at the encoder. However, motion compensation is performed in the DCT domain, and drift error can be introduced adaptively to reduce the computational cost of the DCT-domain motion compensation. Results show that significant computational reduction can be achieved with less than 0.3 dB loss in PSNR when compared to independently encoding the video sequence at multiple rates.

1. INTRODUCTION

This paper addresses the problem of encoding a video sequence at multiple bit rates in order to make the video content available to clients with different access bandwidth. Scalable codecs have been proposed for such scenarios. These codecs can generate a binary stream that can be served to clients with different bandwidth by proper truncation. Their coding efficiency however is lower than that of single stream video encoding. Transcoding can also be used to generate bit streams at lower rates from a compressed stream. This approach has the advantage that any lower bit rate can be generated when needed, but also at the cost of lower coding efficiency due to successive requantization. In today's streaming servers, the video sequence is typically encoded at pre-defined rates. This is the scenario for which this work was done. We have developed an encoder to efficiently and simultaneously encode a video sequence at multiple rates. In this encoder, motion estimation is performed only once for the reference stream. We also take the DCT out of the encoding loop so that it is only computed once per frame. As a result, no iDCT has to be computed at the encoder. However, motion compensation is performed in the DCT domain. By adaptively introducing drift error, we can reduce the computational cost of the DCT-domain motion compensation.

We describe the encoder structure in section 2, and the adaptive DCT-domain motion compensation in section 3. Results are presented in section 4.

2. SIMULTANEOUS ENCODER

Figures 1 and 2 show the block diagrams of the encoder we present in this paper. The encoder is used to generate N streams at different rates from the same video sequence. For each frame f_n, the encoder computes the compressed data for all streams. The first stream is the reference stream, and the others are dependent streams. We use f_n to denote the n^{th} frame of the video sequence, and F_n to denote the DCT transformed image. We use $\hat{F}_n^i$ and $\tilde{F}_n^i$ to respectively denote, for the i^{th} encoded stream, the n^{th} motion compensated frame and the n^{th} reconstructed frame, both in the DCT domain. E_n^i is the prediction error, in the DCT domain, for the n^{th} frame of the i^{th} encoded stream, and $\tilde{E}_n^i$, the prediction error after quantization.

Figure 1 shows the block diagram of the encoder for the reference stream. The motion estimation is computed from the original frames of the sequence. It is also possible to use the reconstructed frames of the reference stream for the motion estimation. This however would require to compute the iDCT of the frames $\tilde{F}_n^i$. In the simulation results presented in section 4, the motion estimation for the multi-rate encoder was done using only original frames.

* This work was done while A. Zaccarin was at Intel Labs.

0-7803-7448-7/02/$17.00 ©2002 IEEE

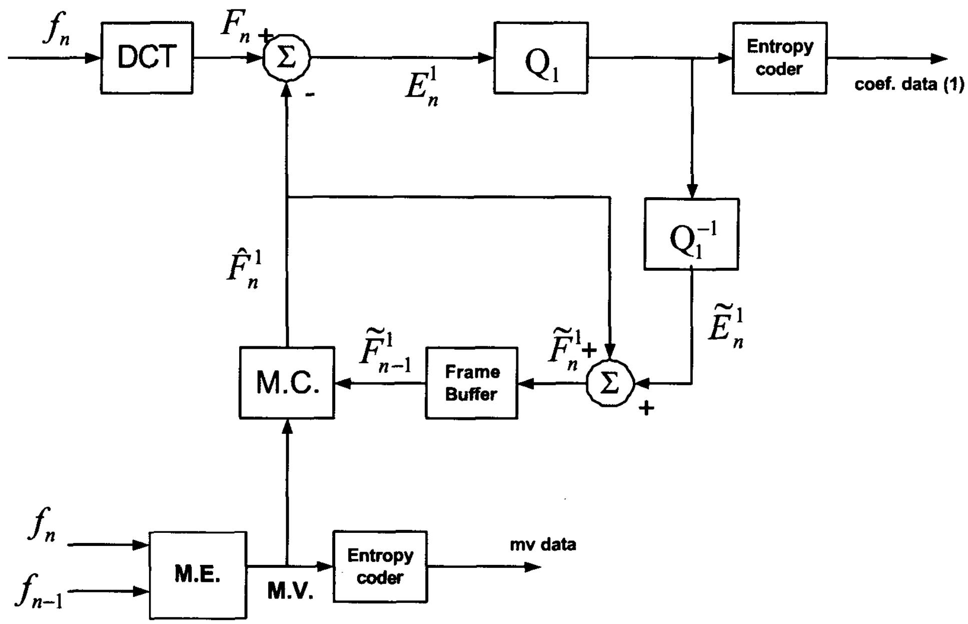

Figure 1 Block diagram of encoder for the reference stream

Figure 2 shows the encoder structure for the dependent streams. Note that no DCT, iDCT or motion estimation is needed for the encoding of these dependent streams. Motion compensation, however, has to be performed in the DCT-domain. The following section addresses this issue and presents the adaptive motion compensation we have implemented in order to reduce the complexity of the DCT-domain motion compensation.

If the number of streams that are generated covers a wide range of bit rates, it is possible to reset the system and introduce a new reference stream for which motion vectors can be re-estimated. More importantly, the reference stream is used to reset the base information used in the adaptive motion compensation. Finally, the reference stream can be defined on a macroblock basis as the macroblocks are independently coded.

3. ADAPTIVE MOTION COMPENSATION

Motion compensation in the DCT domain, and fast algorithms for its computation have been previously described in the literature [1-3]. Our usage of motion

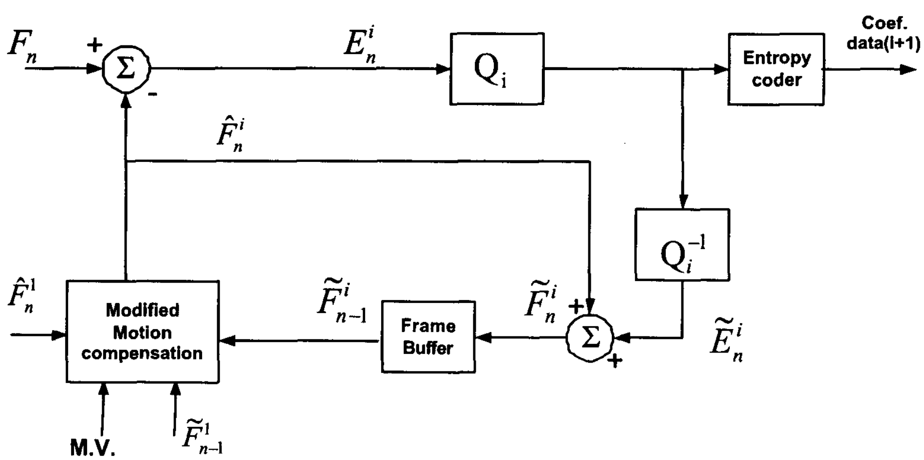

Figure 2 Block diagram of encoder for dependent streams

compensation in the DCT domain (MC-DCT) does note preclude the usage of any of these implementations. However, because we are sequentially encoding a video sequence at different rates, we have the possibility to re-use data that was computed for the reference stream to reduce computation, and also determine how much loss we incur if no motion compensation or only part of it is performed on the dependent stream. The adaptation of the motion compensation is done on a macroblock level.

Figure 3 illustrates the three options we have for the MC-DCT. In the first mode, we have the standard approach,

$$\hat{F}^i_n = \text{MC-DCT}(\tilde{F}^i_{n-1}).$$

where the previously reconstructed frame of the same stream is motion compensated to generate the current predicted frame.

In the second mode, we have

$$\hat{F}^i_n = \hat{F}^1_n + \text{MC-DCT}(\tilde{F}^i_{n-1} - \tilde{F}^1_{n-1}).$$

Here, the motion compensation is computed on the difference between the $n-1^{th}$ reconstructed frames of the current dependent stream and the reference stream. Typically, this difference will be small. In the DCT domain, this translates by a higher number of small or zero coefficients. An appropriate implementation of the MC-DCT can then take advantage of this by not computing the motion compensated DCT coefficients where we expect them to be zero or small. We also use the values of $\tilde{E}^1_n$ to estimate which DCT coefficients will be zero in the motion compensated frame difference in order to further reduce computation. Of course, if a DCT coefficient of the motion compensated frame is not correctly computed at the encoder, it will contribute to a drift (mismatch) error at the decoding time. We have tested different approaches to keep the drift error to a minimum, and results comparing these approaches based on their rate-distorition curves are presented below.

Finally, in the third mode, we simply use the data from the motion compensated frame of the reference stream, i.e.,

$$\hat{F}^i_n = \hat{F}^1_n.$$

This is more appropriate for B frames since the mismatch errors that are introduced do not propagate to other frames.

Note that for all three scenarios, the decision on which approach to take is made on a macroblock basis. Since the first approach does not introduce any errors in the motion compensation, it also resets the reference stream for any given macroblock. If streams are encoded with a decreasing bit rate, resetting the reference stream will increase the number of zero coefficients for the following streams.

4. RESULTS

We have tested the performance of the proposed multi-rate

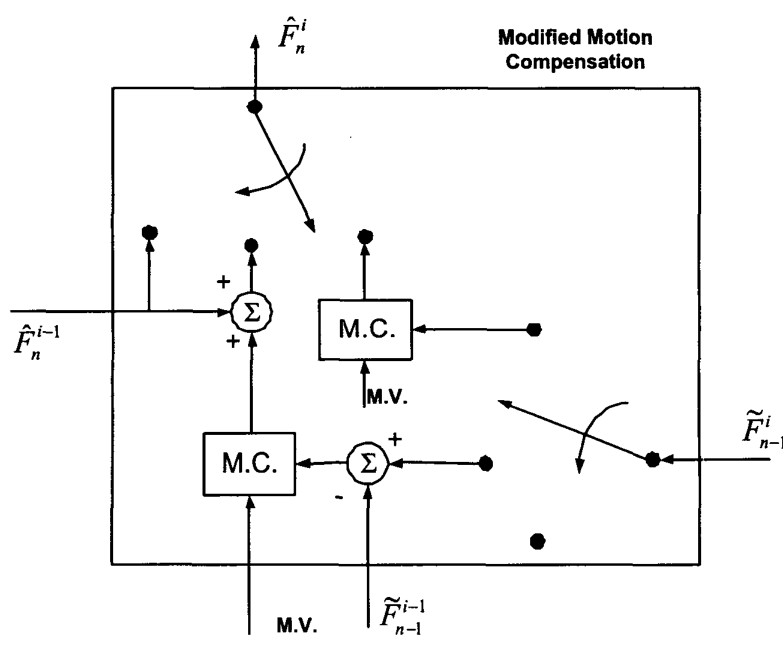

Figure 3 Block diagram of adaptive motion compensation module

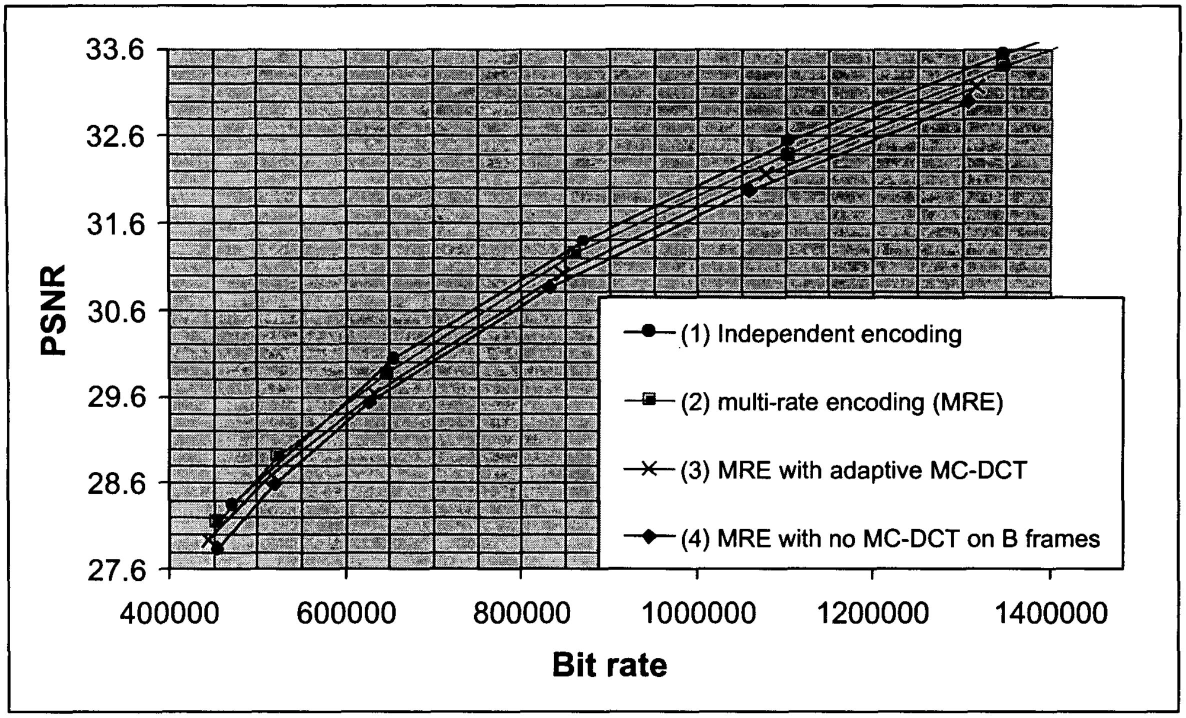

Figure 4 PSNR as function of the bit rate for four different multi-rate encoders

encoder. Simulations were done on a number of CIF sequences with the rate control turned off. The quantization parameter was fixed at each given bit rate, and was the smallest (higher rate) for the first stream.

Figure 4 shows the rate-distortion curves for 150 frames of the sequence BUS. Four curves are shown on this figure. The 1st curve shows the performance of a single stream encoder with the same quantization parameter values used by the multi-rate encoder. Motion estimation is done using the reconstructed frames. The 2nd curve shows the performance of the multi-rate encoder when the motion vectors are estimated from the original frames, and when the MC-DCT is computed for all DCT coefficients. The 3rd curve gives the performance of the multi-rate encoder when the 2nd mode is used to compute the MC-DCT along with a prediction of the location of zero or small DCT coefficients so that their computation can be skipped, thus reducing the overall complexity of the MC-DCT. Finally, for the 4th curve, the MC-DCT was performed accurately for the I & P frames, but skipped for the B frames (3rd mode of MC-DCT). As we can see from these curves, the performance degradation with appropriate approximations and the 2nd mode is under 0.3 dB.

5. CONCLUSION

This paper has presented a multi-rate encoder that can be used to efficiently encode a video sequence at multiple rates. Motion-estimation and DCT are computed once. Motion compensation is done in the DCT domain, and its complexity for dependent streams can be significantly reduced by using information computed for the reference stream. Simulation results show this adaptive motion compensation can be used with this encoder with only a minor degradation in PSNR.

REFERENCES

[1] S.-F. Chang and D.G. Messerschmitt, "A new approach to decoding and compositing motion-compensated DCT images", IEEE ICASSP-93, pp. V-421-424, 1993.

[2] P. Assuncao, M. Ghanbari, "A Frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams", *IEEE TR-CSVT*, Dec. 1998, pp. 953-967.

[3] J. Song, B.-L. Yeo, "A fast algorithm for DCT-domain inverse motion compensation based on shared information in a macro-block", *IEEE TR-CSVT*, August 2000, pp. 767-775.