Rate Control for Motion JPEG2000 Using Correlation Prediction

Jun Hou, Xiangzhong Fang, Rong Hou

Shanhai Jiaotong University,Shanghai,China
{fjshj@hotmail.com, xzfang@cdtv.org.cn, ruthhowe1975@hotmail.com}

***Abstract*-motion JPEG2000 has been widely used for its fine scalability with bit stream and efficiency. A single scalable bitstream can provide precise rate control for variable bitrate(VBR) traffic. The paper present a algorithm that make use of correlation among the video frames to get consistent quality. By using one frame's distortion-rate to predict several next frames' distortion-rate values according to correlation among frames, we use only a few buffers to achieve highly consistent quality which used to use many buffers to achieve. Experimental results using Motion JPEG2000 demonstrate substantial benefits.**

***IndexTerm-* Motion JPEG2000, rate control, correlation, video sequence**

I. INTRODUCTION

The JPEG2000 compression standard [1]-[5] is promising a lot as for as compression efficiency and features to be supported. The JPEG2000 standard allows to manage images with functions that were not present in the other formats and this suggest a wide usage of this format in mobile imaging applications. JPEG2000 has a better efficiency than JPEG for high compression rates and currently perform on better results than any other standard. Moreover using a progressive coding/decoding, the image visualization will be refined at each step adapting itself to the particular resolution display used. JPEG2000 is accurate rate control due to the embedded nature of the encoding scheme EBCOT [7] , which allows compressed bit-streams to be simply truncated to fit a rate constraint. [3] defines a file format called MJ2 (or MJP2) for motion sequences of JPEG 2000 images. MJ2 does not involve inter-frame coding: each frame is coded independently using JPEG 2000. MJ2 is thus scaleable and better suited to networked and point-to-point environments. MJ2 has applications include: storing video clips taken using digital still cameras, high-quality frame-based video recording and editing, digital cinema, medical and satellite imagery.

A rate control technique recommended by the JPEG2000 standard to maximize image quality at a given rate is called post-coding rate distortion optimization (PCRDopt). The advantage of using this method is that at any given rate, the optimum video quality can be achieved for that rate since the algorithm uses the actual rate and distortion values of each code-block in each frame to arrive to the desired rate. In this paper, we propose terminal correlation rate control(TCRC) methods that provide constant quality video under buffer constraints. The paper organizes as following: section II introduce the proposed algorithm, section III give the simulation results and section IV is conclusion.

II. THE PROPOSED ALGORITHM

Embedded wavelet encoders have the capability to set up precisely the bit-rate of each frame. Very often one needs to find out how many bits should be allocated to each frame in order to obtain the highest average signal-to-noise ratio(alternatively, subjective image quality) of the entire sequence for a given rate budget. This optimal bit allocation problem can be stated as [8]:

Given a set of N frames $\{f_i,\ i=1,....N\}$ and an average target rate of Rt bits/frame, we have to encode frame f_i with rate R_i, yielding distortion D_i such that:

$$D = \frac{1}{N}\sum_{i=1}^{N} D_i \tag{1}$$

to minimum, while satisfy $R = \frac{1}{N}\sum_{i=1}^{N} R_i \le Rt$ (2)

where R and D are the average bit-rate and average distortion for the N frames. This problem can be solved via Lagrangian optimization. It can be state as:

minimize $J = D + \lambda R$ (3)

as [9] model the MSE of the ith frame by

$$E_i = \varepsilon^2 \sigma_i^2 w^{-2R_i} : \tag{4}$$

where ε^2 is a constant, then the optimal rate allocation is then given by [10]:

$$R_i = R + \frac{1}{2}\log_2 \frac{\sigma_i^2}{G} \tag{5}$$

where G is the geometric mean of the variances of the frames,

$$G = \left[\prod_{j=1}^{N} \sigma_j^2\right]^{1/N} \tag{6}$$

with the optimum rate allocation, the frame distortions become as [9] gives:

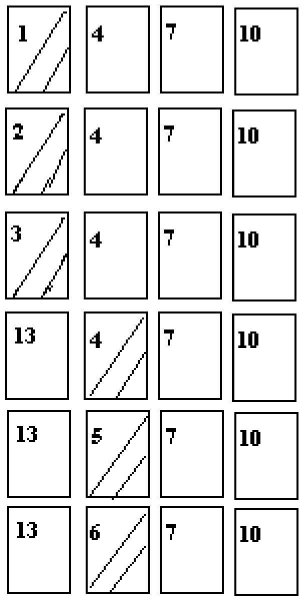

Fig.1 illustrate the buffer content when not end of sequence

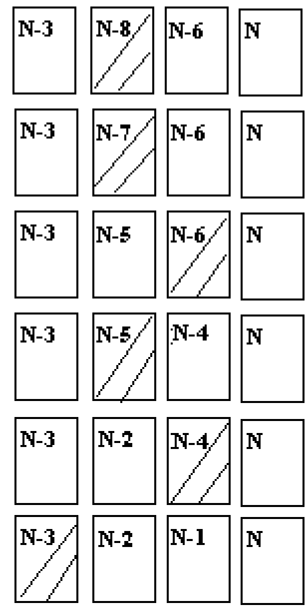

Fig.2 illustrate buffer bank when to end of sequence

$$D_i = G\varepsilon^2 2^{-2R} \quad (7)$$

It shows that D is constant for each frame, minimizing the total distortion means each frame has same distortion. In other words, minimizing average distortion should result in constant quality[9].

Our work is motivated by [9], which use two buffers to smooth the pictures. The algorithm [9] used is called Double Buffer Rate Controller (DBRC). Supposing the delay is N frames, N frames are streamed into first buffer with same distortion-rate(DR), and the total buffer length is N*Rt. At transmission time, the compressed frame are added to delay buffer and truncated to fit buffer limitation at regular intervals corresponding to the frame rate. Simultaneously, data are pulled out of the buffer at some desired constant transmission rate. When the buffer is full, the bitstreams already in the buffer along with the new bitstream to be inserted, are truncated via the embedding property to maintain constant quality across all frames in the buffer. This strategy relies on the highly scalable natrue of JPEG2000. The key idea of [9] is to introduce delay, the more buffer means more delay and better consistent quality as well as more storage memory. [10] share the same idea with [9] to get consistent quality.

Remember where the MJ2 is used. In most case, there are some correlation among frames. One frame has many same scene with the next one. When to JPEG2000 encode, those frames have similar codes, in other words, have similar DR. Usually all the record frames have already stored in media and coded later. In these case, we can use one frame's DR to predict the next ones. By this way, we can use less memory to get effect, which [9] reaches with more delay.

The detail of the proposed algorithm is described as follows:

Suppose there are total M frames in the sequence and there are N buffers. Each buffer has length

L=C*Rt, (8)

where Rt is the budget bitrate, C is constant larger than 1. n(i), i=1....N, is the number of frame, thus $1 \le$ n(i) $\le$ M. sort n(i) i=1.....N in an increasing order, n(t_1), n(t_2), n(t_3) ....n(t_N)

Δ_i is the interval of neighbor sorted n(i) sequence. $\Delta_i = n(t_{i+1}) - n(t_i)$, i=1....N-1, and $\Delta_N = 1$

1) initial: n(1)=1, n(N) $\le$ M, for each buffer
2) code the frames which has frame number n(i), i=1....N. Truncate the coded frames into the ith buffer. Those coding passes that make the total produced bits larger than L will be discarded.
3) Determine a distortion-rate(DR) threshold T_{DR}. each frame in the buffer bank produce bit R_i (i=1....N) while the coding passes with DR slopes $\ge T_{DR}$. So the total produced bits with threshold T_{DR} is the most close to $n(t_N)$*Rt-B and satisfy:

$$\sum_{i=1}^{N} \Delta_i * R_{t_i} \le Rt * n(t_N) - B \quad (9)$$

where B is the coded and released to bitstream number, in other words, $B = \sum^{z \prec \min(n(j))} Rz$, where j=1...N. This step use the n(t_i)th frame code DR character to predict the following Δ_i frame's DR according to the correlation among frames.

4) Release the R_{t_1} bits from the t_1 th buffer and Clear t_1 th buffer., update B=B+ R_{t_1}

5) code a new frame, $k = n(t_1) + 1$,there are two cases:

i) kth frame is not in buffer bank, then code kth frame and truncated it to L bits and put it into t_1 th buffer .

ii) kth frame is already in buffer bank, in other words, kth frame is in t_2 th buffer, so the t_1 th buffer is empty. There are also two cases :

a) n(t_N) $\neq M$, that means the last frame is not in buffer bank, put a frame which number is larger than n(t_N) into t_1 th buffer, i.e:

$n(t_1) = n(t_N) + \Delta$ and satisfy $n(t_1) \leq M$

b) n(t_N)=M, which means the last frame is in buffer bank, increase k until kth frame is not in buffer bank, code kth frame and truncate L bits into t_1 th buffer..

6) comes to the end of coding, Release R_i , i=1......N, bits in each buffer which satisfy T_{DR} threshold.

The idea of the algorithm is that the coding passes having largest DR slopes are retained, consistent with the buffer space available. These coding passes provide the maximum possible decrease in MSE per bit spent, thus minimizing the average distortion of the frames currently in the buffer. There are some extra explain to the TCRC algorithm. 1) since it introduce buffer delay, the DR slopes(determined at compression time) should be stored as auxiliary information on the video server. A conservative estimate of the cost is 5%-10% of the compressed file. The spend cost is the same as [9]. 2)since TCRC use correlation prediction and its interval between two buffer $\Delta \geq 1$, it introduces more delay than [9] does although both of them have same large buffer bank. 3) since the frames are not stored in buffer bank according to their number order, it needs N extra memory to store each frame's number in the buffer bank, only by this way it can point out which buffer should be squeeze out to the bitstream. It's a very small cost which can be eligible. 4) unlike [9], each frame has different delay, so TCRC is suitable for those whose origin date have already stored somewhere. Most of application using MJ2 meet this demand. Fig.1 and Fig.2 give the illustration of the TCRC algorithm. Rectangle means the buffer, here we set $\Delta_i = 3$ i=1...N-1. The digital in the rectangle n indicate that the nth frame in this buffer., in other words: n(i). Rectangle with shade shows this frame will be squeezed to bitstream.

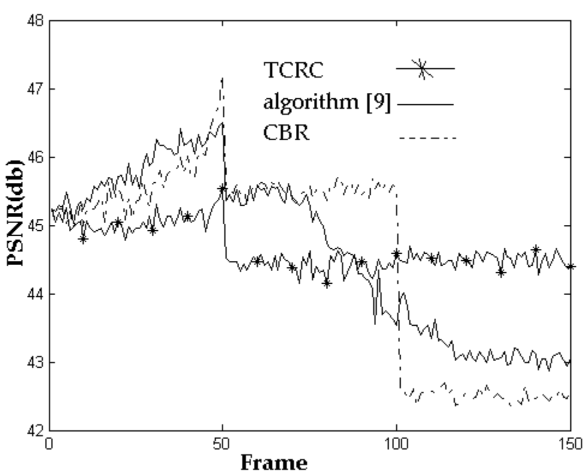

Fig.3 PSNR of first sequence with 30 buffers delay, interval=5

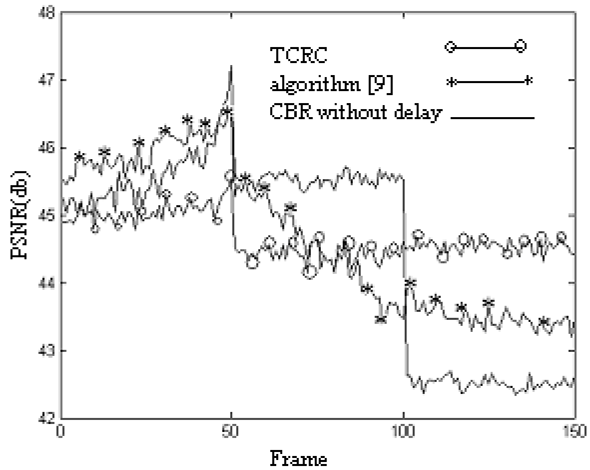

Fig.4 PSNR of first sequence with 45 buffers, rate=0.125

III. SIMULATION

First sequence is a concatenated of CIF video sequence: suzie, miss am, and container. Each scene has 50 frames. Each frame is coded using JASPER software to JPEG2000. The rate is 0.125 and we use loss compression. The size of each codeblock was 64*64. We use Peak Signal-to-Noise-Ratio (PSNR) as the distortion measure. Note DBRC uses two buffers, one buffer is N*Rt bits length and the other is s*N*Rt. TCRC uses N buffers. For convenience of comparing their performance, we suppose both occupy the same length in totally. So we choose C in (8) equals to (1+s). In the simulation, we choose s in DBRC 0.5. We choose C in (8) 1.5. Fig.3 gives the PSNR with 30 buffers in DBRC. It also gives the PSNR of CBR without delay, which means the each frame is coded with Rt bits. In TCRC, the origin interval Δ_i is 5, i=1.....29. Both algorithms with buffers decrease the variance of PSNR. TCRC gets the smoothest PSNR.

We also use 45 buffers for the first sequence. Three algorithms are used to compared the performance. The interval Δ is not constant, some is 3 and some 4. Fig.4 give the PSNR of first sequence with 45 buffers in TCRC. In TCRC, the variance of PSNR is only 7.5% of DBRC's.

TCRC has the smoothest performance comparing with the other two algorithms.

Table.I give the statistical value for three algorithms. For TCRC, (n) in the first column means n buffers in buffer bank. For DBRC it means n frames delay. Table.I shows that for a certain algorithm the more buffer used, the smoother the performance is, as [9][10] have pointed out. TCRC get more consistent performance than DBRC with same buffer length. TCRC using 30 buffers get better performance than DBRC using 45 delays. Also with the smooth of PSNR, the mean of PSNR get higher.

Table.I statistical for first sequence

	mean	var
TCRC(30)	44.6514	0.0900
TCRC(45)	44.6651	0.0889
algorithm [9] (30)	44.6373	1.3879
algorithm [9] (45)	44.6502	1.1963
CBR without delay	44.5412	2.1573

The second sequence is a concatenated of CIF video sequence: car phone, tennis, and coastguard. Each scene has 50 frames. 5 levels of wavelet transform were applied using the irreversible (9,7) wavelet. The size of each codeblock was 64*64. Also TCRC in (8) set C=1.5. In DBRC, the second buffer has half length of the first buffer. TCRC has 20 buffers in buffer bank and DBRC has 20 delays. Thus both algorithms occupy the same length of the memory. In TCRC, the initial interval Δ_i=5. Fig.5 gives the PSNR of second sequence using three algorithms separately. Both algorithms with buffers get smoother performance than CBR without delay. TCRC has finer performance than DBRC.

Table.II statistical of second sequence

	mean	var
TCRC (20)	42.3628	0.1953
algorithm [9] (20)	42.3029	1.1015
CBR without delay	42.1712	1.9589

Table.II give the statistical value for second sequence using three algorithms separately. The variance of PSNR using TCRC is about 17.8% of DBRC's. DBRC's variance of PSNR is about 56.2% of CBR without delay. Also the TCRC has higher means of PSNR.

IV. CONCLUSION

Using the embedded nature of JPEG2000, the bitstream can be truncated anywhere for a JPEG2000 picture. The proposed algorithm TCRC is a VBR rate control for MJ2. It introduces buffer delay and use the correlation among frames to predict the DR character of unknown frames. By this way, it gets smoother quality for video sequence. Comparing with DBRC, it gets more consistent quality while using same length buffers. It's very suitable for those sequences that all origin frames have already stored somewhere. TCRC is very simple and suitable for real-time application.

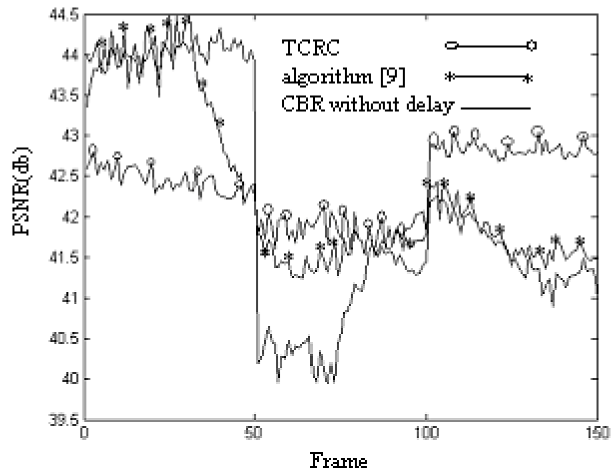

Fig.5 PSNR of second sequence with 20 buffers, rate=0.125

REFERENCE:

[1] ITU-T Recommendation T.800| ISO/IEC 15444-1:2002, Information technology -JPEG 2000 Image encoding system -part1: Core coding system

[2] ITU-T Recommendation T.801| ISO/IEC 15444-2:2002, Information technology -JPEG 2000 Image encoding system -part2: Extensions

[3] ISO/IEC 15444-3:2002, Information technology -JPEG 2000 Image encoding system -part3: Motion JPEG2000

[4] ITU-T Recommendation T.803| ISO/IEC 15444-4:2002, Information technology -JPEG 2000 Image encoding system -part 4: Conformance testing

[5] ITU-T Recommendation T.804| ISO/IEC 15444-5:2002, Information technology -JPEG 2000 Image encoding system -part 5: Reference software

[6] http://www.jpeg.org/jpeg2000

[7] D.Taubman, "High performance scalable image compression with EBCOT", IEEE Trans. Image Processing, vol.9, pp.1158-1170, July 2000

[8] Ramchandran, K., Ortega, A, . "Rate-distortion methods for image and video compression", IEEE signal processing, vol 15, pp23-50

[9] Joseph C. Dagher, Ali Bilgin, and Michael W.Marcellin, "Resource-Constrained Rate Control for motion JPEG2000", vol 12, pp1522-1529, Dec 2003

[10] N.S. Jayant and P. Noll, " Digital Coding of Waveforms" Englewood Cliffs, NJ:, Prentice-Hall, 1984

[11] Rogerio Caetano, Eduardo A.B.da silva, "A bit allocation scheme for a class of embedded wavelet video encoders", journal Visual communication & image representation, vol 14 ,2003, pp136-149

[12] M. W. Marcellin, M. J. Gormish, A. Bilgin, and M. P. Boliek, "An overview of JPEG-2000," in *Data Compression Conference*, Snowbird, UT, Mar. 2000, pp.523–541.

[13] J. Xu, Z. Xiong, S. Li, and Y. Q. Zhang, "Three-dimensional embedded subband coding with optimized truncation (3-D ESCOT)," *Appl.Comput. Harmon.Anal.*, vol. 10, pp. 290–315, 2001.