Rate-distortion estimation for fast JPEG2000 compression at low bit rates

K.N. Vikram, V. Vasudevan and S. Srinivasan

In JPEG2000 block coding, all coding passes are generated before rate allocation is performed among code-blocks. Unwanted passes are then discarded. For low bit-rate coding, this results in discarding of a large number of coding passes. In this letter, we propose a rate-distortion estimation method that enables pre-compression rate-distortion optimization to be carried out, wherein only the required passes need to be coded. Experiments using the proposed technique demonstrate speedup factors ranging from 1.17 to 1.78 at 0.0625 bpp, for JPEG2000 compression.

Introduction: JPEG2000 is a current standard for image compression. An important feature of JPEG2000 is state-of-the-art low bit-rate performance in terms of decoded image quality [1]. Requirement of efficient low bit rate implementation for JPEG2000 compression is therefore of paramount importance.

Block coding in JPEG2000 is based on Embedded Block Coding with Optimized Truncation (EBCOT) [2]. EBCOT operates on small blocks of quantized subband data called code-blocks, in terms of fractional bit-planes called coding passes, to generate coded sub-bit-streams. EBCOT achieves compression by discarding less important coding passes from each sub-bit-stream such that the distortion is minimized while the target rate is met. At low bit rates, a large number of coding passes are discarded. It is possible to speed up the encoder, if coding of unnecessary passes can be avoided. Considerable effort has been expended recently [3–7] to speedup the block coding engine and speedup results for Tier-1 of block coding have been reported, based on this idea. However, information on the speedup obtained for the overall JPEG2000 encoding process is not available in the literature. In this letter, we propose a simple and efficient technique to estimate rate-distortion characteristic of each code-block. This enables us to perform pre-compression rate-distortion optimization and thus avoid coding of the unnecessary passes. We compare the speedup obtained for Tier-1 coding using the proposed method, with the previous attempts. We also report the speedup values obtained for the whole compression process.

Motivation: EBCOT retains the required number of coding passes from the sub-bit-stream of each code-block, based on a post-compression rate-distortion optimization (PCRD-opt) algorithm [2]. Optimization requires knowledge of the contribution of each coding pass or fractional bit-plane towards increase in rate and reduction in distortion. We have observed that based on the significance information derived from each code-block, it is possible to quickly estimate the rate-distortion values for each bit-plane. This observation is the basis of our procedure, which is now described.

Proposed rate-distortion estimation: A code-block coefficient is said to become significant in bit-plane i if that bit plane holds the most significant (non-zero) bit of the coefficient. We number the bit-planes starting from 1 for the least significant bit-plane to N for the most significant bit-plane. The coefficient which becomes significant in bit-plane *i* is said to be in a state of magnitude-refinement for all bit-planes < *i* and insignificant for bit-planes > *i*. We define two metrics ΔD_M^i and ΔL_M^i, which are measures of the decrease in distortion (ΔD^i) and increase in sub-bit-stream length (ΔL^i) respectively, for bit-plane *i* of a code-block, as

$$\Delta D_M^i = (N_s^i + 0.25 N_r^i)(2^{(i-1)})^2 \quad (1)$$

$$\Delta L_M^i = 2N_s^i + N_r^i + N_{insig}^i \quad (2)$$

where N_s^i is the number of code-block coefficients that become significant in bit-plane *i* and N_r^i is the number of coefficients that are in a state of magnitude refinement. N_{insig}^i is the number of insignificant coefficients that are deemed to contribute significantly towards an increase in the bit-rate, based on run-mode coding in JPEG2000. For computation of N_{insig}^i, each column of the bit-plane is tested for insignificance in groups of four, starting from the first row. If all four coefficients in a group turn out to be insignificant, they are excluded from counting towards N_{insig}^i. The factor 0.25 in equation (1) is based on the expression for expected distortion reduction for magnitude refinement coefficients, derived in [8]. The factor 2 in equation (2) accounts for the fact that whenever a coefficient becomes significant, its sign bit is also coded. Computation of N_s^i, N_r^i and N_{insig}^i requires only the significance information of the quantized coefficients in the code-block. Computation of ΔD_M^i and ΔL_M^i requires only addition and shifting operations, since the multiplications in equations (1) and (2) are with powers of 2.

We have compared the computed values of ΔD_M^i and ΔL_M^i for each bit-plane, against the actual values ΔD^i and ΔL^i, for a set of reference images. We have done the comparison separately for each subband. For all subbands other than the lowest frequency subband LL, we have observed a remarkable correlation between ΔD_M^i, ΔL_M^i and ΔD^i, ΔL^i (correlation coefficient between 0.95 and 1.00). For LL subband, there is a reduced correlation between ΔL_M^i and ΔL^i, for reasons explained later. Within a subband, it is possible for code-blocks to have a different number of bit-planes. For comparison between ΔD_M^i and ΔD^i, we align the least significant bit-planes of all code-blocks in a subband, while for comparison of ΔL_M^i with ΔL^i, we align the most significant bit-planes of all code-blocks in a subband. We have obtained lines of best fit for each bit-plane, for plots of ΔD^i vs ΔD_M^i and ΔL^i vs ΔL_M^i. This is done for each subband separately. Since the correlation coefficient values are very high, accurate estimation of the actual distortion reduction and rate increase for every bit-plane in a code-block is possible (for any image), using

$$\Delta D_E^i = K_{1b}^i \times \Delta D_M^i + K_{2b}^i \quad (3)$$

$$\Delta L_E^i = K_{3b}^i \times \Delta L_M^i + K_{4b}^i \quad (4)$$

where the subscript E indicates that it is the estimated value. K_{1b}^i, K_{2b}^i, K_{3b}^i and K_{4b}^i are parameters of lines of best fit for bit plane *i* in subband *b*. Knowledge of the values of ΔD_E^i and ΔL_E^i for all code-blocks is sufficient to perform a Lagrangian optimization for choosing the required number of coding passes, in exactly the same way as in EBCOT. The algorithm for rate-distortion estimation may now be formulated as

for each subband *b* other than LL **do**

for each code-block **do**

Obtain significance information for all coefficients

for each bit-plane *i* **do**

Obtain N_s^i, N_r^i and N_{insig}^i

Compute ΔD_M^i and ΔL_M^i using equations (1) and (2) respectively

Estimate ΔD_E^i and ΔL_E^i using equations (3) and (4) respectively

end for

end for

end for

While correlation between ΔD^i and ΔD_M^i is good for the LL subband, reduced correlation between ΔL_M^i and ΔL^i for code-blocks in LL may be attributed to large clustering of the significant coefficients inherent in LL, leading to higher compression efficiency [9]. This results in equation (2) overestimating ΔL^i to a large extent. To obtain accurate rate-distortion information, we have decided to code all the passes for code-blocks in LL subband.

Results: We have used a set of 12 reference grayscale images for one-time offline computation of the line parameters K_{1b}^i, K_{2b}^i, K_{3b}^i and K_{4b}^i. We have used Jasper software [10] for obtaining the rate-distortion statistics. The distortion measure used is the mean square error. We have tested our proposed technique on a different set of grayscale images, after modifying Jasper for our method. For all subbands other than LL, we estimate the rate-distortion values for each code-block, using our algorithm. For the LL subband, we code all the passes and obtain the rate-distortion values from the actual measured parameters. We then perform a Lagrangian optimization to compute the required number of passes. After this, we code only the required passes for code-blocks in the non-LL subbands. In the LL subband, we discard the unnecessary passes. The results are given in Table 1. We have employed 3 levels of (9, 7) wavelet decomposition and code-block size of 64x64. Rate is in bits per pixel (bpp). T-1 % is the percentage of total execution time occupied by Tier-1 of block coding, in the original Jasper implementation. T-1 % gives an indication of the overall speedup that can be achieved. PSave, TSave and CSave give the savings respectively in the number of passes, execution time and number of coded contexts, for Tier-1 coding. ΔPSNR is the drop in PSNR value due to our method, compared to PCRD-opt, while ΔR is the deviation in achieved rate from the target rate. Speedup reported is with respect to the original Jasper implementation (that employs PCRD-opt) for the overall compression process. Speedup results include the overhead due to rate-distortion estimation. The results indicate that large speedup ratios are attainable, with small deviation from target rate and small PSNR drop.

Table 2 compares our technique with the existing methods present in the literature. In the literature, three different measures for Tier-1 coding speedup have been used. We compare our method using all the three measures. The results in [3], [5] and [6] are an average over a set of images. This is used here as it is. Refs [4] and [7] have separate results for different images. The best and worst case results are respectively presented here. Similarly, the best and worst case results of our proposed technique are also given. Our best-case results are better than the other results in the table. Our worst-case results fall slightly below a few of the other results. However, our ΔPSNR values are better in these cases. The efficiency of our technique is demonstrated by the overall speedup achieved (Table 1).

Conclusions: We have presented a simple and efficient method to estimate rate-distortion statistics of each code-block. We have demonstrated that a rate-distortion estimation strategy is useful for speeding up JPEG2000 at low bit rates. The proposed technique achieves speedup with very low degradation in decoded picture quality and very small deviations from the target bit rate.

References

1. A. SKODRAS, C. CHRISTOPOULOS, and T. EBRAHIMI: ‘The JPEG 2000 Still Image Compression Standard’, IEEE Signal Processing Magazine, vol. 18, no. 5, pp. 36–58, September 2001.
2. D. TAUBMAN: ‘High Performance Scalable Image Compression with EBCOT’, IEEE Transactions on Image Processing, vol. 9, no. 7, pp. 1158–1170, July 2000.
3. T. MASUZAKI, H. TSUTSUI, T. IZUMI, T. ONOYE, and Y. NAKAMURA: ‘JPEG2000 adaptive rate control for embedded systems’, in IEEE International Symposium on Circuits and Systems (ISCAS), May 2002, pp. IV–333 – IV–336.
4. Y. M. YEUNG, O. AU, and A. CHANG: ‘Successive bit-plane rate allocation technique for JPEG2000 image coding’, in 2003 International

Conference on Acoustics, Speech, and Signal Processing (ICASSP), April 2003, pp. III–261–III–264.

5. T.-H. CHANG, C.-J. LIAN, H.-H. CHEN, J.-Y. CHANG, and L.-G. CHEN: ‘Effective hardware-oriented technique for the rate control of JPEG2000 encoding’, in 2003 International Symposium on Circuits and Systems (ISCAS), May 2003, pp. II–684 – II–687.
6. Y. M. YEUNG, O. AU, and A. CHANG: ‘An efficient optimal rate control scheme for JPEG2000 image coding’, in 2003 International Conference on Image Processing (ICIP), September 2003, pp. 761–764.
7. W. YU: ‘Integrated rate control and entropy coding for JPEG 2000’, in Data Compression Conference (DCC), March 2004, pp. 152 – 161.
8. J. LI and S. LEI: ‘An embedded still image coder with rate-distortion estimation’, IEEE Transactions on Image Processing, vol. 8, no. 7, pp. 913–924, July 1999.
9. T.T. LU and P.C. CHANG: ‘Significant bit-plane clustering technique for JPEG2000 image coding’, Electronics Letters, vol.40, no. 17, pp.1056-1057, 19th August 2004.
10. M. D. ADAMS and F. KOSSENTINI: ‘Jasper: a software-based JPEG-2000 codec implementation’, in International Conference on Image Processing (ICIP), September 2000, pp. vol.2, 53–56.

Authors’ affiliations:

K.N. Vikram, V. Vasudevan and S. Srinivasan (Department of Electrical Engineering, Indian Institute of Technology Madras, Chennai 600036, India).

E-mail: vikramkn@ieee.org

Table captions:

Table 1 Results for the proposed technique

Table 2 Comparison with existing techniques. Entries with two values are (best-case, worst-case) pairs.

Table 1

Image	Rate, bpp	T-1, %	Speedup	ΔPSNR, dB	ΔR, %	Psave, %	Csave, %	Tsave, %
Mountain (640×480)	0.5	61.7	1.63	–0.14	3.02	74.21	82.62	83.77
	0.25		1.72	0.08	3.26	85.23	89.83	90.55
	0.125		1.75	0.06	–3.71	90.04	93.39	93.26
	0.0625		1.78	0.13	–1.50	95.02	96.39	95.79
Baboon (512×512)	0.5	29.9	1.26	0.06	1.60	80.89	84.30	84.31
	0.25		1.27	–0.07	–0.81	86.38	89.83	90.09
	0.125		1.27	–0.01	3.78	93.57	94.20	93.93
	0.0625		1.27	0.03	–4.25	95.55	95.97	95.30
Lena (512×512)	0.5	22.0	1.14	0.33	2.16	66.71	73.70	69.63
	0.25		1.15	0.26	2.42	80.76	84.40	83.10
	0.125		1.17	0.12	2.73	90.08	91.14	90.14
	0.0625		1.17	–0.13	4.20	94.33	94.09	93.75
Boy (768×512)	0.5	39.8	1.31	0.00	3.65	71.67	80.57	80.67
	0.25		1.38	–0.12	4.40	80.30	87.81	87.80
	0.125		1.31	0.00	1.43	88.72	92.68	91.88
	0.0625		1.30	0.02	3.52	93.68	95.43	94.50
Church (640×480)	0.5	60.1	1.54	0.05	2.94	76.53	81.78	82.54
	0.25		1.63	–0.12	4.70	84.79	88.65	89.28
	0.125		1.63	0.06	1.00	90.80	93.01	93.30
	0.0625		1.65	–0.04	0.79	94.56	95.87	95.07

Table 2

Rate, bpp	Parameter	[3]	[4]	[5]	[6]	[7]	Proposed
0.5	Psave, %	70	—	—	40	66.7, 50.0	80.9, 66.7
	Csave, %	—	—	65	—	—	84.3, 73.7
	Tsave, %	—	77, 71	—	—	—	84.3, 69.6
0.25	Psave, %	83	—	—	51	80.0, 66.7	86.4, 80.3
	Csave, %	—	—	79	—	—	89.8, 84.4
	Tsave, %	—	89, 85	—	—	—	90.6, 83.1
0.125	Psave, %	92	—	—	67	88.9, 80.0	93.6, 88.7
	Csave, %	—	—	87.5	—	—	94.2, 91.1
	Tsave, %	—	94, 93	—	—	—	93.9, 90.1
0.0625	Psave, %	—	—	—	76	93.8, 90.9	95.6, 93.7
	Csave, %	—	—	—	—	—	96.4, 94.1
	Tsave, %	—	—	—	—	—	95.8, 93.8