NEAR-OPTIMAL LOW-COST DISTORTION ESTIMATION TECHNIQUE FOR JPEG2000 ENCODER

Amit Kumar Gupta, David Taubman, Saeid Nooshabadi

University of New South Wales,
Sydney, Australia

ABSTRACT

Optimal rate-control is a very important feature of JPEG2000 which allows simple truncation of compressed bit-stream to achieve best image quality at a given target bit-rate. Accurate distortion estimation with respect to allowed bit-stream truncation points, is essential for rate-control performance. In this paper, we address the issues involved in accurate distortion estimation for hardware oriented implementation of JPEG2000 encoding systems with generic block coding capabilities. We propose a novel hardware friendly distortion estimation technique. Rate control based on the proposed technique results in only an average 0.02 dB PSNR degradation with respect to the optimal distortion estimation approach used in the software implementations of JPEG2000. This is the best performance reported in comparison to existing techniques. The proposed technique requires only an additional 4096 bits per block coder which is 80% less than the memory requirements of optimal approach.

1. INTRODUCTION

Optimal rate-control is one of the important features of the new JPEG2000 image compression standard. The rate-control algorithm used by the JPEG2000 standard is a part of EBCOT algorithm proposed in [1]. EBCOT is a key module of JPEG2000 compression system consisting of embedded block coder (Tier-I EBCOT) and Post Compression Rate Distortion (PCRD) analysis (Tier-II EBCOT) as shown in Figure 1. The block coder consists of Bit Plane Coder (BPC) and Arithmetic Coder (AC) sub-modules. It encodes independently so-called *code-blocks* of sub-band wavelet coefficients. For every code-block, the block coder generates an embedded bit-stream with several possible truncation points. The PCRD algorithm selects optimal truncation points from all code-blocks to achieve a target bit-rate while minimizing associated distortion.

The performance of rate-control algorithms [1] rely heavily on accurate estimation of distortion associated with a truncation point. Our simulation results show that inaccurate distortion estimation results in an average of 2.5 dB PSNR degradation and thus, considerable image quality loss especially at low bit-rates.

Distortion estimation is embedded within block coding algorithm. For software implementations of JPEG2000 such as Kakadu [2], distortion estimation is trivial and done with a simple look-up table embedded within the block coder module. But a simple look-up table based approach requires up to 20 kbits of memory for a hardware oriented generic block coder module such as [3] and [4], as shown in Section 2.3.

In this paper, we propose a novel distortion estimation approach which is suitable for hardware implementation. The proposed approach is simple yet effective and achieves very close to optimal performance with only an average 0.02 dB performance degradation. Additionally it only requires an extra 4096 bits of memory. Further, we present a VLSI architecture of our proposed technique integrated with the block coder accelerator.

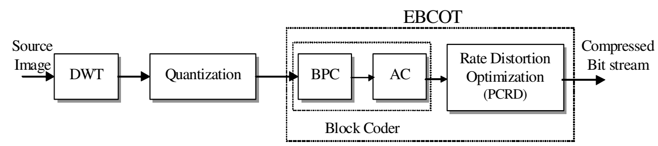

Fig. 1: A high level block diagram of JPEG2000 Encoder system

2. BACKGROUND

2.1. JPEG2000 Rate-control Algorithm

Though a JPEG2000 encoder is free to choose its rate-control algorithm, the JPEG2000 standard recommends use of the PCRD algorithm [5] for rate-control. The PCRD algorithm is a part of EBCOT algorithm along with the block coder as shown in Figure 1. A code-block i is encoded in a bit-plane by bit-plane fashion starting from most significant magnitude bit-plane, $p_{\max_i}$. Each magnitude bit-plane, p, is further decomposed into three coding passes denoted as $C^{(p,k)}, k \in [0,1,2]$. The first pass is Significant Propagation (SP) pass; followed by the Magnitude Refinement (MR) pass, and the CleanuP (CP) pass, represented as $C^{(p,0)}$, $C^{(p,1)}$ and $C^{(p,2)}$ respectively.

A set of feasible truncation points, $\{n_i\}$, for the embedded bit-stream of a code-block is formed by the coding pass end-points, mainly as coding pass end-points are expected to lie on convex-hull representation of rate-distortion function [6]. Let $(L_i^{n_i}, D_i^{n_i})$ represents the resulting length and distortion if i^{th} code-block is truncated to the truncation point n_i. Note that $n_i = 1$ refers to the truncation point at the end point of first coding pass and so on. The rate-control optimization process selects the truncation points of all code-blocks under the given rate constraint

$$L \geq \sum_i L_i^{n_i} \tag{1}$$

such that overall additive image distortion [5] $D = \sum_i D_i^{n_i}$ is minimized. Here L denotes the overall bit-stream length at a target bit-rate. This is equivalent to saying that any set of truncation points, $\{n_i(\lambda)\}$, which minimizes the cost function

$$D(\lambda) + \lambda L(\lambda) = \sum_i (D_i^{n_i(\lambda)} + \lambda L_i^{n_i(\lambda)}) \tag{2}$$

for some $\lambda > 0$. Thus, the rate-control optimization problem reduces to find a value of λ such that the resulting set of truncation points for all code-blocks, $\{n_i^{(\lambda)}\}$, minimizes the cost function of equation (2) subject to the rate constraint of equation (1). It is shown in [1] that minimization of equation (2) reduces to an independent minimization for each code-block i.e. to find $n_i^{(\lambda)}$ which minimizes $D_i^{n_i} + \lambda L_i^{n_i}$ for a code-block i.

A "distortion-length slope" is defined for a truncation point as

$$S_i^{n_i} = \frac{\triangle D_i^{n_i}}{\triangle L_i^{n_i}} = \frac{D_i^{n_i-1} - D_i^{n_i}}{L_i^{n_i} - L_i^{n_i-1}} \tag{3}$$

An iterative approach [1] is used to find the convex-hull of truncation points among all truncation points which have strictly decreasing distortion-length slope. Finally a suitable global value of λ is found using distortion-length slopes of reduced set of truncation points of all code-blocks.

2.2. Distortion Estimation-Mathematical Formulation

Rate-control algorithm selects truncation points based on their distortion-length slopes. Distortion-length slope in turn, depends on change in distortion by which a coding pass $C^{(p,k)}$ reduces distortion with respect to the previous coding pass. Henceforth we refer to the change in the distortion associated with a coding pass $C^{(p,k)}$ of i^{th} code-block as delta-distortion $\triangle D_i^{(p,k)}$. For mean square error as the measure of distortion, it is shown [6] that delta-distortion may be accurately estimated using 2 small lookup tables indexed by fractional bits as shown in equation (4) below

$$\triangle D_i^{(p,k)} = W_i^{(p)} \begin{cases} \sum\limits_{j \in C^{(p,k)}, \nu^{(p)}[j]=1} \mathrm{LUT}_S[\overline{\upsilon}[\mathbf{j}]];\ \mathrm{k} \in [0,2] \\ \sum\limits_{j \in C^{(p,k)}} \mathrm{LUT}_M[\overline{\upsilon}[j]];\ \mathrm{k}=1 \end{cases} \tag{4}$$

Here, $W_i^{(p)} = G_i \triangle_i^2 2^{2p}$, G_i, $\triangle_i$are energy gain factor of the subband and deadzone quantization parameter respectively. $\mathbf{j}$ represents a sample-location within i^{th} code-block. The quantities $\nu^{(p)}[\mathbf{j}]$ and $\overline{\nu}^{(p)}[\mathbf{j}]$ denote the integer and fraction parts of $\nu[\mathbf{j}]$ truncated for least p magnitude bit-planes, where $\nu[\mathbf{j}]$ is the quantized subband coefficient magnitude. Specifically,

$$\overline{\nu}^{(p)}[\mathbf{j}] = \frac{\nu[\mathbf{j}]}{2^p} - \nu^{(p)}[\mathbf{j}] = \frac{\nu[\mathbf{j}]}{2^p} - \left\lfloor \frac{\nu[\mathbf{j}]}{2^p} \right\rfloor \tag{5}$$

where, $\lfloor\rfloor$ represents the 'floor' operation. LUT_S is the lookup table used for SP and CP coding passes. LUT_M is the lookup table used for MR coding pass. The lookup tables are indexed by $\overline{\upsilon}[\mathbf{j}]$ which is a integer generated from $\overline{\nu}^{(p+1)}[\mathbf{j}]$ as $\overline{\upsilon} = 2^{p+1}\overline{\nu}^{(p+1)}[\mathbf{j}]$. The condition $\nu^{(p)}[\mathbf{j}] = 1$ signifies that for SP and CP passes, only those samples for which first non-zero magnitude bit is coded, contribute to delta-distortion.

The length of lookup tables depend on the number of fractional bits used to estimate delta-distortion. 5 fractional bits are commonly used by commercial JPEG2000 software such as Kakadu [2] which is known for its best rate-control performance. Let us define a fractional bit representation for $\overline{\nu}^{(p+1)}[\mathbf{j}]$ consisting of 5+1 fractional bits as

$$\overline{\nu}^{(p+1)}[\mathbf{j}] = 0.x_5x_4x_3x_2x_1x_0 \tag{6}$$

where x_5 denotes the first fractional magnitude bit and so on. It is to be noted that x_5 denote the magnitude bit in p^{th} bit-plane. Since $\nu^{(p)}[\mathbf{j}] = 1$ for a sample turning significant during p^{th} bit-plane, $\overline{\nu}^{(p+1)}[\mathbf{j}]$ is represented as $\overline{\nu}^{(p+1)}[\mathbf{j}] = 0.1x_4x_3x_2x_1x_0$. Thus, for 5 fractional bits, LUT_S has 32 entries indexed by 5 bit decimal value represented as $\{x_4x_3x_2x_1x_0\} \in [0-31]$. LUT_M has 64 entries indexed by a 6 bit quantity, $\{x_5x_4x_3x_2x_1x_0\} \in [0-63]$.

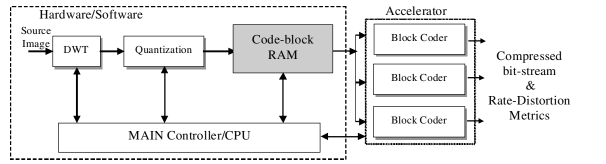

Fig. 2: A block diagram for Hardware oriented JPEG2000 employing accelerator for block coding

2.3. Hardware Oriented JPEG2000 Encoder

Figure 2 shows a block diagram of a generic hardware oriented JPEG2000 encoder system such as [7]. It uses a pool of block coder accelerators while other modules like DWT and quantization are implemented on a general purpose processor.

To reduce internal memory and bandwidth, a generic block coder has just enough memory to hold 1 magnitude bit-plane [4]. The encoding is done while transferring the data in a bit-plane by bit-plane fashion from code-block RAM (Figure 2). Thus, fractional bits are not accessible within the block coder module. Delta-distortion can not be accurately estimated directly from subband coefficients as distribution of significant samples between SP and CP passes is dynamically decided during block coding. This makes delta-distortion estimation a non-trivial issue for JPEG2000 encoder systems employing dedicated hardware accelerator for the block coding.

3. EXISTING APPROACHES

3.1. Software-Oriented Approach

Software-Oriented (SO) approach is a simple look-up table based approach as discussed in Section 2.2. In software implementations like Kakadu, access to fractional bits is available while block coding and hence, are used for accurate delta-distortion estimation. SO approach has optimal performance but its hardware implementation requires at least $5 \times 4096 = 20$ kbits extra memory per block coder.

3.2. Model-Oriented Approach

A Model-Oriented (MO) approach is suggested in [8]. It requires number of samples in a coding pass from the block coder and model fractional bits using image statistics. MO approach has the advantage that distortion estimation is done with out access to code-block memory. A main disadvantage of this approach is its poor performance due to strong modelling dependency on images used to extract model parameters. Our simulation results show that even for ISO images, the MO approach suffers an average 2 dB PSNR degradation.

3.3. Pre Compression Approach

The Pre-Compression (P_{re}C) approach [9] only allows end points of MR coding passes as possible truncation points for all but last two bit-planes. For last two bit-planes, both SP and MR passes end points are considered. This approach has the advantage of generating the distortion estimation directly from subband coefficients but suffers from PSNR performance degradation by an average 3 dB (based on our simulation results). It is mainly because all three coding passes have high percentage (SP pass $\sim 40-70\%$, MR pass $\sim 70-80\%$ and CP pass $\sim 80-85\%$ [6]) of being a termination point on the rate-distortion convex hull. Thus, restricting the truncation points to only MR pass may result in selection of non-optimal truncation points.

4. PROPOSED APPROACH AND VLSI ARCHITECTURE

4.1. Proposed Approach

We propose the use of 1 most significant fractional bit and modelling of remaining 4 fractional bits for distortion estimation. Referring again to equation (4), delta-distortion for SP and CP passes may be written as:

$$\triangle D_i^{(p,k)} \cong W_i^{(p)} \left(\begin{array}{c} N_{x_4=0} \left(\sum_{\overline{v}=0}^{15} \left(f_{pmf-S}^{(p,0)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) + \\ N_{x_4=1} \left(\sum_{\overline{v}=16}^{31} \left(f_{pmf-S}^{(p,1)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) \end{array} \right) \quad ; k \in [0,2] \quad (7)$$

where, $f_{pmf-S}^{(p,x_4)}(\overline{v})$ denote the conditional probability distribution of $\overline{v}$ during SP and CP passes for a given value of x_4 at p^{th} bit-plane; $N_{x_4=0}$ and $N_{x_4=1}$ denote the number of samples belonging to the respective coding pass which have their x_4 bit equal to 0 and 1 respectively. Similarly for MR pass delta-distortion is estimated by

$$\triangle D_i^{(p,1)} \cong W^{(p)} \left(\begin{array}{c} N_{x_5x_4=00} \left(\sum_{\overline{v}=0}^{15} \left(f_{pmf-M}^{(p,00)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) + \\ N_{x_5x_4=01} \left(\sum_{\overline{v}=16}^{31} \left(f_{pmf-M}^{(p,01)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) + \\ N_{x_5x_4=10} \left(\sum_{\overline{v}=32}^{47} \left(f_{pmf-M}^{(p,10)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) + \\ N_{x_5x_4=11} \left(\sum_{\overline{v}=48}^{63} \left(f_{pmf-M}^{(p,11)}(\overline{v}) \mathrm{LUT}_S[\overline{v}] \right) \right) \end{array} \right) \quad (8)$$

where, $f_{pmf-M}^{(p,x_5x_4)}(\overline{v})$ denote conditional probability distribution of $\overline{v}$ during MR pass for a given value of x_5 and x_4 bits at p^{th} bit-plane; $N_{x_5x_4=00}$ (and others) denote the number of samples belonging to MR pass at p^{th} bit-plane which have the respective values for x_5 and x_4 bits. The count values N_{x_4} and $N_{x_5x_4}$ are generated from actual subband coefficients while block coding as discussed in Section 4.2.

For estimating $f_{pmf-S}^{(p,x_4)}$ and $f_{pmf-M}^{(p,x_5x_4)}$, we propose 2 different models, Laplacian and Uniform. The Laplacian model is selected as high frequency subbands of natural still images are known to have Laplacian distribution [10] represented as:

$$f(x) = \frac{\gamma}{2} e^{-\gamma |x|} \quad (9)$$

where γ denote the Laplacian distribution. A simple way to estimate γ is via the standard deviation (σ) of image subband coefficients on a code-block by code-block basis as $\gamma = \frac{\sqrt{2}}{\sigma}$. A trade-off exists between computational cost and model accuracy. We find that even an average value of γ achieves excellent performance (~0.02 dB performance degradation). The Uniform model is selected to avoid modelling error and computational cost involved in γ calculation. Simulation results show that the Uniform model also results in near-optimal performance.

4.2. VLSI Architecture

The VLSI architecture for Distortion Estimation (DE) module based on proposed approach is tightly embedded with the Bit Plane Coder (BPC) module of the block coder as it uses actual distribution of samples among the coding passes. We retrofitted our existing BPC module [11] with the DE module. The DE module includes counters to generate the count values N_{x_4} and $N_{x_4x_5}$. Additionally, a dual port RAM of 4096 bits is used. Thus the BPC module has two data bit-plane memories which are used as ping-pong buffer to store current and next magnitude bit-plane.

As evident from equations (7) and (8), our proposed approach only needs N_{x_4} values during SP and CP pass; and $N_{x_5x_4}$ values during MR pass, for distortion estimation. The DE module uses Coding State (CS), Significance State (SS) bits and magnitude bits [11] to identify samples distribution among coding passes. Specifically, CS bit for a sample-location identify whether the sample-location has been coded or not in the current bit-plane. The SS bit maintains the significance status of the location; it becomes 1 after the first non-zero bit of the location has been coded.

Encoding starts as soon as sufficient data for most significant magnitude bit-plane has been transferred to the block coder. N_{x_4} values for the first coding pass, $C^{(p_{\max},2)}$, are generated during the second coding pass, $C^{(p_{\max}-1,0)}$. The sample-locations which become significant in $C^{(p_{\max},2)}$ are identified by their SS state bits. Since the current magnitude bit-plane data bits represent the x_4 bit, N_{x_4} are easily updated. The count values, N_{x_4} and $N_{x_4x_5}$, for all subsequent coding passes are generated during the corresponding MR pass using current and next magnitude bit-plane data bits. Count values are output as distortion metrics which, along with 4 modelled fractional bits, are used for delta-distortion estimation.

5. IMPLEMENTATION RESULTS

In this section, we present the performance of the proposed approach and its comparison with existing approaches. PSNR difference with Kakadu's performance is used as an evaluation basis. Rate-control is performed using the PCRD algorithm. We use ISO test images Lena (512×512), Bike (512×640), Cafe (512×640), Woman (512×640), Compound2 (2000×2587) and Chart (1688×2347). The images, Compound2 and Chart, represent artificial images with varying statistics across the image. These images are particularly selected to evaluate the performance across a rich gamut of still images. All are compressed for lossless case using 5/3 wavelet kernel using 5 levels of DWT and 64×64 code-block size. For the proposed Laplacian model approach, we use an average value for $\gamma = 0.1765$ obtained using statistics of Lena, Bike, Cafe and Woman ISO images.

Table 1 presents the PSNR difference of Kakadu's performance with existing and proposed techniques at different bit-rates. The results for Cafe and Woman are not included due to space restrictions but they exhibit similar patterns. The results show that the proposed Laplacian and Uniform approaches perform near-optimal with an average performance degradation of 0.02dB only, while both existing approaches have large degradation (average 2.5 dB). It is to be noted that reported results for the existing approaches are 0.3 dB (MO approach [8]) and 0.3-0.7dB (P_{re}C approach [8]). A probable reason for the difference in simulation and reported results may be the use of different rate-control algorithm. Nevertheless, even a comparison with reported results show that the proposed approaches have 93-97% less PSNR degradation than the existing approaches. A comparison of Laplacian and Uniform approaches is shown in Figure 3 for two sets of images. Set 1 consists of Compound2 and Chart. Bike, Lena, Cafe and Woman constitute set 2. It shows that both Laplacian and Uniform model have similar performance with 0.014 and 0.016 dB degradation respectively. Even for set 2 which contains artificial still images, our proposed approaches have only an average 0.02 dB PSNR degradation.

Requirement of additional 4096 memory bits/block-coder is the main increase in hardware cost of the proposed technique with re-

Table 1: PSNR (dB) Difference of distortion estimation appraoches with Kakadu. 'MO' -Model oriented approach; '$P_{re}C$' - Pre-compression distortion approach; 'Proposed (γ)'- Proposed Laplacian model (γ=0.1765), and 'Proposed (U) '- Proposed Uniform model.

Image	Approach	Bit-Rate (bpp)											
		0.1	0.2	0.5	0.75	1	1.25	1.5	2	2.5	3	3.5	4
Lena	MO	2.34	2.48	2.65	2.05	2.53	2.52	2.76	3.0	3.0	3.17	3.95	3.12
	$P_{re}C$	0.46	0.4	0.37	0.94	0.85	1.36	2.21	2.51	2.79	4.47	6.07	4.15
	Proposed (γ)	0.03	0.08	0.02	0.02	0.03	-0.01	0.03	0.004	0.019	0	0	0
	Proposed (U)	0.002	0.10	0.02	0.04	0.04	0.04	0.03	0.04	0.045	0	0	0
Bike	MO	1.75	1.5	1.5	1.12	1.12	0.92	1.14	1.65	0.99	1.24	1.03	1.74
	$P_{re}C$	0.64	1.59	3.67	5.45	5.84	6.1	6.4	7.68	2.54	1.3	1.96	2.67
	Proposed (γ)	0.19	-0.01	0.03	0.004	0.05	0.01	-0.008	0.02	0.001	0.03	0.017	0.005
	Proposed (U)	0.03	0	0.03	0.03	0.05	0.007	-0.02	0.05	-0.009	0.04	0.004	0.012
Compound2	MO	1.92	2.05	1.73	1.74	1.68	1.82	1.77	1.65	1.88	1.96	2.2	2.33
	$P_{re}C$	1.62	1.97	4.43	4.34	5.27	5.19	4.82	4.55	5.06	4.03	2.92	2.06
	Proposed (γ)	0.04	0.07	0.11	0.06	0.03	0.06	0.02	-0.006	0.009	-0.01	0	0
	Proposed (U)	0.04	0.06	0.11	0.06	0.03	0.11	0.02	0.025	0.008	0.05	-0.001	0
Chart	MO	2.12	2.03	2.74	3.07	2.71	3.0	2.97	3.06	4.51	3.77	8.21	0
	$P_{re}C$	3.29	4.08	5.74	4.99	5.39	2.77	2.87	1.95	2.32	1.56	1.79	0
	Proposed (γ)	0.07	0.03	0.02	0.068	-0.01	0.02	-0.06	0.006	-0.004	0.002	0	0
	Proposed (U)	0.05	0.02	0.14	-0.006	0.02	-0.03	0.008	-0.009	-0.033	0.002	0	0

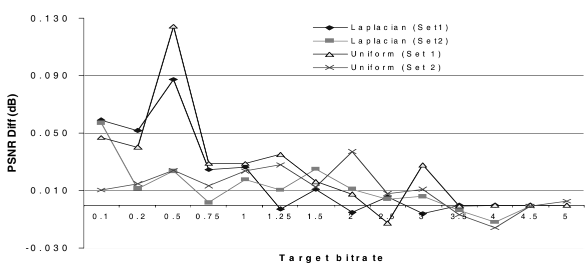

Fig. 3: A comparison of average PSNR difference of Laplacian and Uniform approaches for two sets of images: Set1- Compound2 and Chart ; Set2- Bike, Lena, Cafe and Woman.

spect to existing approaches [8] [9]. But this memory requirements is 80% less than the memory requirements of the optimal SO approach.

6. CONCLUSION

In this paper we have proposed a simple yet effective distortion estimation approach for hardware oriented JPEG2000 encoder system. The proposed approach is simple to implement and requires only extra 4096 bits of memory per block-coder which is 80% less than the memory requirements for the optimal approach. The PSNR performance of proposed approach is near-optimal with only 0.02 dB average degradation with respect to optimal performance. We presented two models for fractional bits - Laplacian and Uniform models. Even an average value of the Laplacian distribution parameter results in near-optimal performance for artificial images. Further, JPEG2000 implementation systems may choose to use real image statistics to model the Laplacian distribution. The Uniform model does not require calculation of model parameters and also has near-optimal performance, though slightly less than the Laplacian model.

7. REFERENCES

[1] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Transactions on Image Processing*, vol. 9, pp. 1158–1170, 2000.

[2] D. Taubman, "Kakadu software- a comprehensive framework for JPEG2000." http://www.kakadusoftware.com/.

[3] C.-J. Lian, K.-F. Chen, H.-H. Chen, and L.-G. Chen, "Analysis and architecture design of block-coding engine for EBCOT in JPEG 2000," in *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 219 – 230, 2003.

[4] A. K. Gupta, M. Dyer, A. Hirsch, S. Nooshabadi, and D. Taubman, "Design of a single chip block coder for the EBCOT engine in JPEG2000," *IEEE Int. Midwest Consference on Circuits and Systems*, 2005.

[5] "JPEG2000 part I final committee draft version 1.0 ISO/IEC JTC1/SC29/WG1N1646R," March 2000.

[6] D. S. Taubman and M. W. Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*. Norwell, Massachusetts 02061 USA: Kluwer Academic Publishers, 2002.

[7] D. Freeman and G. Knowles, "Novel architecture for the JPEG2000 block coder," *Journal of Electronic Imaging*, vol. 13(4), pp. 897–906, 2004.

[8] Q. Xing, Y. Xiaolang, G. Haitong, and Y. Ye, "A simplified algorithm of JPEG2000 rate control for VLSI implementation," *IEEE Int. Conference on Circuits and Systems (ISCAS)*, pp. 6316–6319, 2005.

[9] Y.-W. Chang, H.-C. Fang, C.-J. Lian, and L.-G. Chen, "Novel pre-compression rate-distortion optimization algorithm for JPEG 2000," *Proc. of SPIE-Journal of Electronic Imaging*, vol. 5308, pp. 1353–1361, 2004.

[10] S. R. Smoot and L. A. Rowe, "Laplacian model for AC DCT terms in image and video coding." http://bmrc.berkeley.edu/research/publications/1996/110/110.html.

[11] A. K. Gupta, S. Nooshabadi, and D. Taubman, "Concurrent symbol processing capable VLSI architecture for bit plane coder of JPEG2000," *IEICE Transactions on Information and Systems, Special Section on Recent Advances in Circuits and Systems*, vol. E88-D, pp. 1878 – 1884, 2005.