An Embedded Still Image Coder with Rate-Distortion Optimization

Jin Li, *Member, IEEE,* and Shawmin Lei, *Senior Member, IEEE*

***Abstract*—It is well known that the fixed rate coder achieves optimality when all coefficients are coded with the same rate-distortion (R-D) slope. In this paper, we show that the performance of the embedded coder can be optimized in a rate-distortion sense by coding the coefficients with decreasing R-D slope. We denote such coding strategy as *rate-distortion optimized embedding* (RDE). RDE allocates the available coding bits first to the coefficient with the steepest R-D slope, i.e., the largest distortion decrease per coding bit. The resultant coding bitstream can be truncated at any point and still maintain an optimal R-D performance. To avoid the overhead of coding order transmission, we use the expected R-D slope, which can be calculated from the coded bits and is available in both the encoder and the decoder. With the probability estimation table of the QM-coder, the calculation of the R-D slope can be just a lookup table operation. Experimental results show that the rate-distortion optimization significantly improves the coding efficiency in a wide bit rate range.**

***Index Terms*—Embedded coding, image coding, rate-distortion optimization, rate-distortion slope, scalability, wavelet.**

I. Introduction

EMBEDDED image coding receives great attention recently. In addition to providing a very good coding performance, the embedded coder has the property that the bitstream can be truncated at any point and still decoded a reasonable good image. Some representative works of embedding include the embedded zerotree wavelet coding (EZW) proposed by Shapiro [1], the set partitioning in hierarchical trees (SPIHT) proposed by Said and Pearlman [2], and the layered zero coding (LZC) proposed by Taubman and Zakhor [3]. The ability to adjust the compression ratio by simply truncating the coding bitstream makes embedding very attractive for a number of applications such as progressive image transmission, internet browsing, scalable image and video database, digital camera, low delay image communication, etc. Taking the internet image browsing as an example, with the functionality of embedding, we may store only one copy of high quality image at the server side, and deliver to the browser a part of the bitstream depending on the user demand, channel condition, and browser monitor quality. At the earlier stage of browsing, images may be retrieved with coarse quality so that a user can quickly go through a large number of images and choose the one of his or her interest. Then the chosen image can be downloaded with a much better quality level. During the download process, the quality of the image can be gradually refined, and the user may terminate the download process as soon as the image quality is satisfactory.

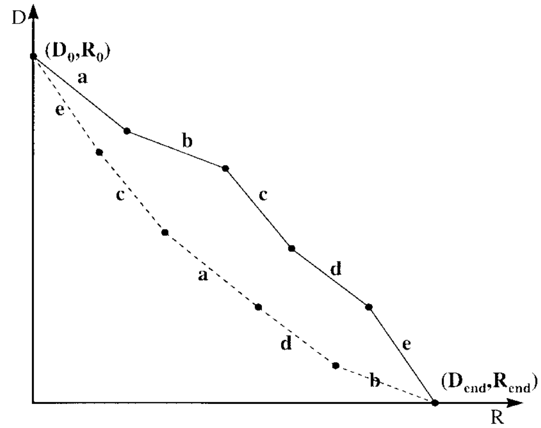


Fig. 1. Initiative of rate-distortion optimization.

The essence of embedding is that the bitstream can be arbitrarily truncated. An immediate question is: Is there an optimal coding strategy to generate an embedded bitstream so that the coder is not only optimized at the final rate, but also optimized at every truncation point? It turns out that the optimal strategy is to first encode those symbols with the steepest rate-distortion slope. The initiative can be illustrated in Fig. 1. Suppose there are five symbols a, b, c, d and e that can be coded independently. The coding of each symbol requires a certain amount of bits and results in a certain amount of distortion decrease. Sequential coding in the order of symbol a to e gives the R-D curve shown as the solid line in Fig. 1. If the coding is reordered so that the symbol with the steepest R-D slope is encoded first, we can get the R-D curve shown as the dashed line in Fig. 1. Though both performance curves reach the same final R-D point, the algorithm that follows the dashed line performs much better when the output bitstream is truncated at an intermediate bit rate. We therefore propose a rate-distortion optimized coder (RDE), which allocates the available coding bits first to the coefficient with the steepest R-D slope, i.e., the one with the largest distortion decrease per coding bit.

There are quite a few works on rate-distortion optimization for fixed rate coders. It is well known that a fixed rate coder achieves optimality if the rate-distortion (R-D) slopes of all coded coefficients are the same [4]. The criterion was used

Manuscript received December 2, 1997; revised September 10, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Robert Forchheimer.

The authors are with Sharp Laboratories of America, Camas, WA 98607 USA (e-mail: lijin@sharplabs.com).

Publisher Item Identifier S 1057-7149(99)05107-6.

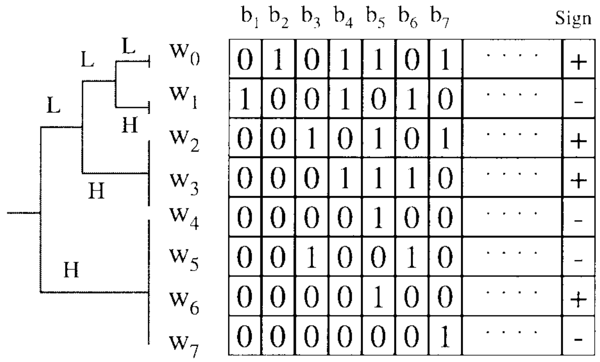


Fig. 2. Bit array after transform.

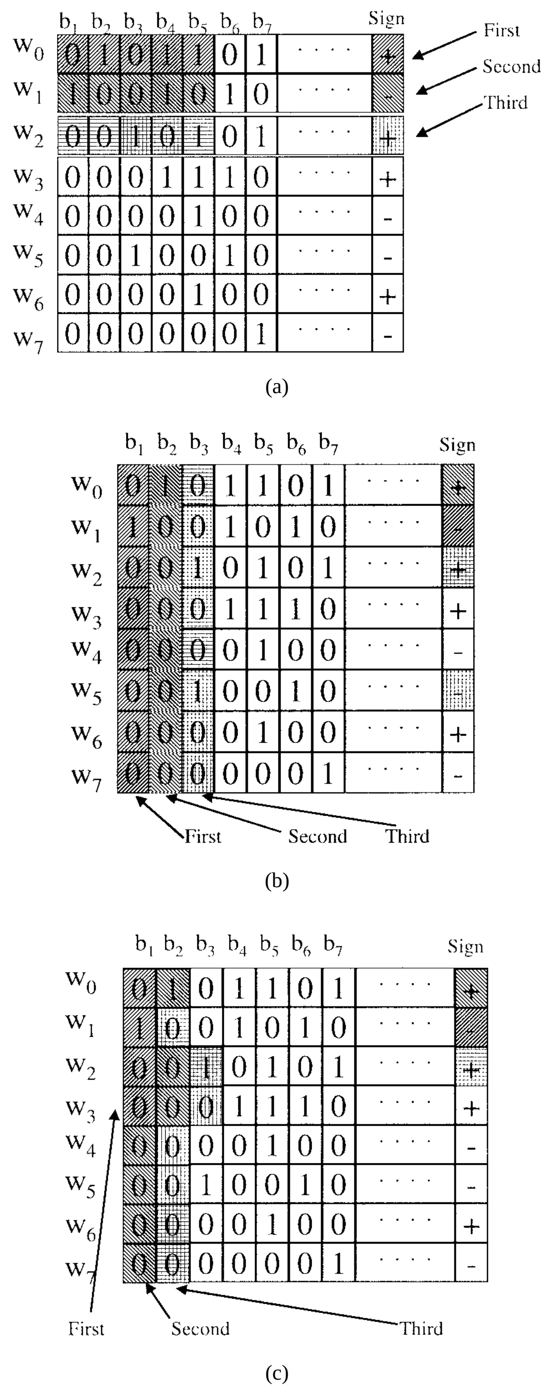


Fig. 3. Coding order of (a) conventional coder, (b) embedded coder, (c) rate-distortion optimized embedded coder (RDE).

in rate control [5], [6] to adjust the quantization step size Q of each macroblocks, in which case the coding of video was optimal when the R-D slopes of all macroblocks were constant. Xiong and Ramchandran [8] also used the constant rate-distortion slope criterion to derive the optimal quantization for wavelet packet coding. However, to our knowledge, there were no existing works on rate-distortion optimization of the embedded coder. Li *et al.* [9] showed that the R-D slopes of significance identification and refinement coding were different, and by placing the significance identification before the refinement coding, the coding efficiency could be improved. However, the improvement of [9] was fairly limited as it only affects the coding order of a few coefficients.

This paper is organized as follows. The framework and the implementation detail of RDE are investigated in Section II. We focus primarily on the two key steps of RDE, i.e., the R-D slope calculation and the coefficient selection. To avoid sending the overhead of coding order, RDE is based on the expected R-D slope that can be calculated by both the encoder and the decoder. We simplify the calculation of R-D slope to one lookup table operation with the help of the probability estimation table of the QM-coder [10], [11]. In Section III, the performance of RDE is compared with various other algorithms with extensive experiments. It is shown that RDE significantly improves the coding efficiency. Concluding remarks are presented in Section IV.

II. Implementation of Rate-Distortion Optimized Embedding (RDE)

A. Notations

Let us assume that the image has already been converted to the transform domain. The transform used in the embedded coding is usually the wavelet decomposition, but it can be DCT as well, as in [15]. Let the index of a transform coefficient be denoted by $i = (s, d, x, y)$, where s is the scale of the wavelet decomposition, d is the subband of decomposition which includes LL, LH, HL, and HH, and x, y are the spatial positions within the subband. The first and second letter in d represent the filter applied in the vertical and horizontal direction, respectively. We use L for lowpass filter and H for highpass filter. Let the total number of transform coefficients be denoted by N. Let the coefficient at index position i be denoted by w'_i. Suppose the coefficients have already been normalized through the division of the maximum absolute value of the transform coefficients T_0:

$$w_i = \frac{w'_i}{T_0} \quad \text{with} \quad T_0 = \max_i |w'_i|. \tag{1}$$

The normalized transform coefficients w_i are used throughout the following discussion. Because w_i is between -1 and $+1$, it can be represented by a stream of binary bits as

$$\pm 0.b_1 b_2 b_3 \cdots b_j \cdots \tag{2}$$

TABLE I
ELABORATED CODING ORDER OF RDE FOR FIG. 3(c)

Order	Symbol	Value	Order	Symbol	Value	Order	Symbol	Value
1	b_1 of w_0	0	10	b_1 of w_4	0	19	b_2 of w_5	0
2	b_1 of w_1	1	11	b_1 of w_5	0	20	b_2 of w_6	0
3	sign of w_1	-	12	b_1 of w_6	0	21	b_2 of w_7	0
4	b_1 of w_2	0	13	b_1 of w_7	0	.		
5	b_1 of w_3	0	14	b_2 of w_1	0	.		
6	b_2 of w_0	1	15	b_3 of w_2	1	.		
7	sign of w_0	+	16	sign of w_2	+			
8	b_2 of w_2	0	17	b_3 of w_3	0			
9	b_2 of w_3	0	18	b_2 of w_4	0			

where b_j is the jth most significant bit or the jth coding layer of coefficient w_i. In the proposed rate-distortion optimized embedding (RDE), the coding symbol, which is defined as the smallest unit for R-D optimization, is either one single bit b_j of the coefficient w_i or the sign of w_i. Nevertheless, the concept of RDE can be extended to other embedded coders, where the coding symbol may consist a group of bits, as in the case of the embedded zerotree wavelet coding (EZW) [1] or the set partitioning in hierarchical trees (SPIHT) [2]. A sample bit array produced by a one-dimensional (1-D) wavelet transform is shown in Fig. 2, in which the ith row of the bit array represents the transform coefficient w_i, and the jth column of the bit array represents the bit plane b_j. We place the most significant bit at the left most column, and place the least significant bit at the right most column.

The order to encode the bitarray is different among the conventional, embedded, and rate-distortion optimized embedded coder. The conventional coder such as JPEG [12] or MPEG [13] first determines the quantization precision, or equivalently, the number of bits to encode each coefficient, then sequentially encodes one coefficient after another with certain entropy coding. Using the bit array of Fig. 2 as an example, the conventional coding is ordered row by row as shown in Fig. 3(a). The embedded coding is distinctive from the conventional coding in the sense that the image is coded bit-plane by bit-plane, or column by column as shown in Fig. 3(b). The embedding bitstream can be truncated and still maintain reasonable image quality, since the most significant part of each coefficient is coded first. It is also suited for progressive image transmission because the quality of the decoded image gradually improves as more and more bits are received. On the other hand, the coding order of rate-distortion optimized embedding (RDE) is optimized for progressive image transmission. RDE calculates the R-D slope λ_i for each bit b_j and encodes first the one with the largest R-D slope. The actual coding order of RDE depends on the calculated R-D slope and is image dependent. An example of the coding order of RDE is shown in Fig. 3(c). A more elaborated coding order of RDE is shown in Table I, where the order of coding, the symbol to encode and its value are listed in column 1, 2 and 3, respectively.

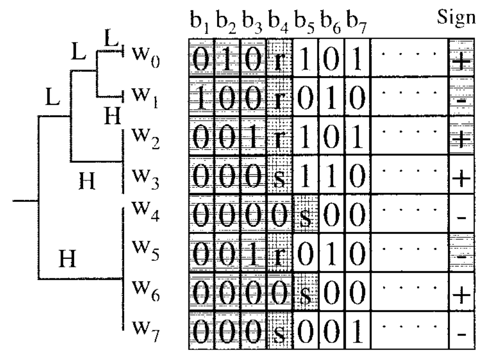


Fig. 4. Estimation of the rate-distortion slope based on the transmitted bits. (The bits marked by horizontal bars have already been coded; the bits marked with check board patterns are the next bits to be encoded.)

B. The Expected Rate-Distortion Slope

If the optimization is based on the actual R-D slope, the decoder has to be informed of the order of coding. The overhead to transmit the location of the symbol with the largest actual R-D slope is so large that it easily nullifies any advantages that can be brought up by rate-distortion optimization. To avoid transmitting the coding order, we use the expected R-D slope that can be calculated by both the encoder and the decoder. The concept can be shown in Fig. 4. Suppose at a certain coding instance, the most significant $(n_i - 1)$ bits of coefficient w_i, $i = 1, \cdots, N$ have been encoded, and the next bit under consideration is the n_ith bit. RDE calculates the expected R-D slope λ_i for each candidate bit b_{n_i}, and encodes the one with the largest λ_i value. The expected R-D slope λ_i is based on the coding layer n_i, the significance status of coefficient w_i (whether all of the previous $(n_i - 1)$ bits of w_i are none zero), and the significance statuses of its surrounding coefficients. It gives an estimate of the distortion decrease per coding bit if bit b_{n_i} is coded. Since the information used to calculate the expected R-D slope is available at the decoder, the decoder can follow the coding order of the encoder without any overhead transmission. The coding strategy ensures that at each step, RDE encodes the symbol that gives the maximum expected distortion decrease per bit spent, thus achieves the best R-D performance for the embedded coding just as shown by the dashed line in Fig. 1.

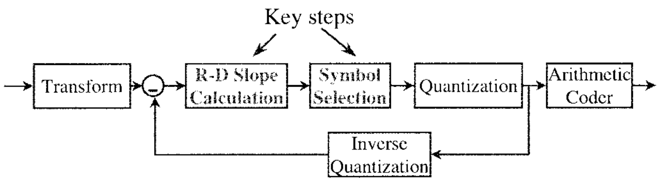


Fig. 5. Operation flow chart of rate-distortion optimized embedding.

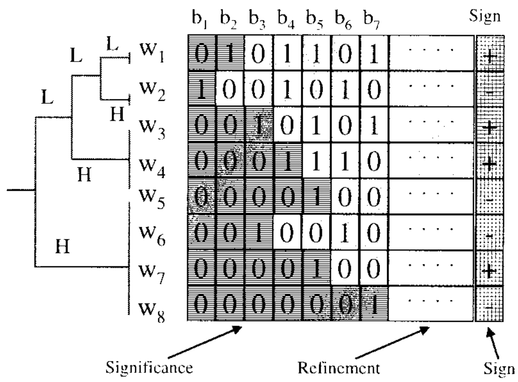


Fig. 6. Significance identification (marked with horizontal bars), refinement coding (marked with dots), and sign bit (marked with check board patterns).

The operation flow chart of RDE can be shown in Fig. 5. Compared with traditional embedding, there are two key distinguished steps in RDE, i.e., R-D slope calculation and coefficient selection. Both steps have to be efficient so that the computational complexity of RDE remains low. We will discuss the two steps in details in the following sections.

C. Calculation of the Rate-Distortion Slope

In this section, we develop a very efficient algorithm that calculates the expected R-D slope with just a lookup table operation. We first describe the coding strategy of RDE, which is closely related to the R-D slope calculation. In RDE, the coding of candidate bits b_{n_i} falls into two categories—significance identification and refinement. If all previously coded bits b_j in coefficient w_i are zeros, $j = 1 \cdots n_i - 1$, the significance identification mode is used to encode bit b_{n_i}, otherwise, the refinement mode is used. For convenience, coefficient w_i is called insignificant if all its previously coded bits are zeros. The insignificant coefficient is reconstructed with value zero at the decoder side. When the first nonzero bit b_{n_i} is encountered, coefficient w_i becomes significant. Its sign needs to be encoded to distinguish the coefficient between positive and negative, and it becomes nonzero at the decoder. From that point on, the refinement mode is used to encode the rest bits of coefficient w_i. We show an example in Fig. 6. All the bits that have undergone significance identification are marked by horizontal bars, and all the bits that have undergone refinement are marked by dots. The bits marked by checkerboard patterns are sign bits, which are encoded when a coefficient just becomes significant. The expected R-D slopes and coding processes for significance identification and refinement are completely different.

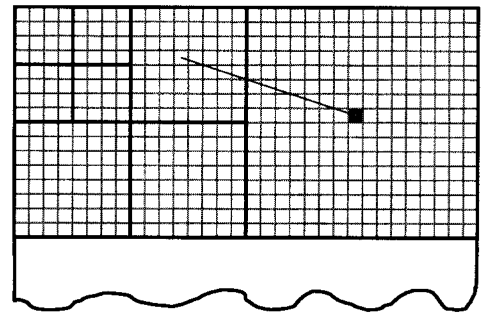


Fig. 7. Context for QM-coder. (■ is the current coding coefficient w_i and □ are its context coefficients.)

In significance identification, the coded bit is highly biased toward zero, i.e., nonsignificance. We encode the result of significance identification with a QM-coder, which estimates the probability of significance of coefficient w_i (denoted as p_i) with a state machine, and then arithmetic encodes it. As shown in Fig. 7, the QM-coder uses a context which is a 7-b string with 6 b representing the significant statuses of six spatial neighbor coefficients and 1 b representing the significant status of the parent coefficient which corresponds to the same spatial position but one scale up the current coefficient w_i. The coder uses a total of 128 context registers, each of which contains two bytes recording the QM-coder state and the most probable symbol, respectively. The context is shared between different wavelet scale and orientations (LH, HL, HH). Whenever a neighbor or parent coefficient is unavailable, e.g., for the coefficients in the LL subband of the coarsest scale or at the boundary of a subband, the corresponding context bit is set to zero. By monitoring the pattern of past zeros ("insignificance") and 1s ("significance") under the same context (i.e., the same neighborhood configuration), the QM-coder estimates the probability of significance p_i of the current coding symbol. The concept is that if there were n_0 0 symbols and n_1 1 symbols in the past coding with the same context, the probability p that the current symbol appears one can be calculated by Bayesian estimation as

$$p = \frac{n_1 + \delta}{n_0 + \delta + n_1 + \delta} \tag{3}$$

where δ is a parameter between $[0, 1]$ which relates to the *a priori* probability of the coded symbol. We may associate the probability p with a state. Depending on whether the coded symbol is one or zero, it increases or decreases the probability p and thus transfers the coder to another state. By merging of the state of similar probabilities and balancing between the accuracy of probability estimation and quick response to the change in source characteristics, a QM-coder state table can be designed. For details of the QM-coder and its probability estimation, we refer to [10], [11], and [12]. In general, the probability estimation is very simple and is just a table transition operation. In RDE, the estimated probability of significance p_i is used not only for arithmetic coding, but also for the calculation of the R-D slope λ_i. On the other hand, the refinement and sign bits are equilibrium between zero and one. They are encoded by an arithmetic coder with fixed probability 0.5.

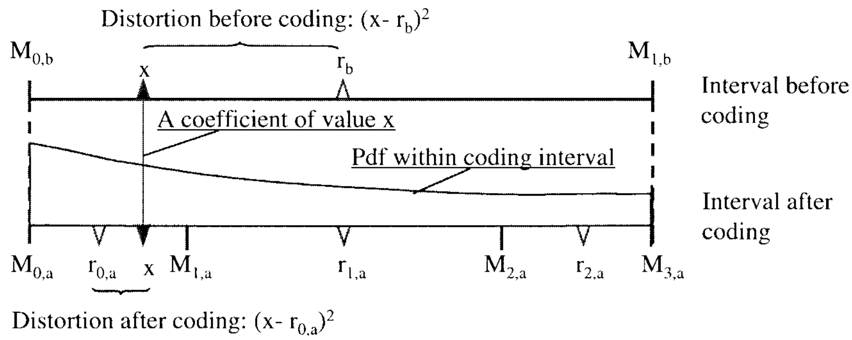


Fig. 8. Illustration of coding interval subdivision.

RDE needs to calculate the expected R-D slope λ_i for all the candidate bits b_{n_i}, which is the average distortion decrease divided by the average coding rate

$$\lambda_i = \frac{E[\Delta D_i]}{E[\Delta R_i]}. \tag{4}$$

The expected R-D slope can not be calculate by averaging the distortion decrease per coding rate:

$$\lambda_i \neq E\left[\frac{\Delta D_i}{\Delta R_i}\right]. \tag{5}$$

The reason behind (4) is just like the calculation of the average speed. When a vehicle travels through different segments with varying speed, its average speed is equal to the total travel distance divided by the total travel time, it is not equal to the average of speed of different segments.

Suppose before coding bit b_{n_i}, coefficient w_i is within interval $(M_{0,b}, M_{1,b})$ with decoding reconstruction r_b. Coding of bit b_{n_i} supplies additional information of coefficient w_i and restricts it into one of K subintervals $(M_{k,a}, M_{k+1,a})$ with decoding reconstruction $r_{k,a}$, $k = 0, \cdots, K-1$, as illustrated in Fig. 8. The interval boundaries satisfies the relationship:

$$M_{0,b} = M_{0,a} < M_{1,a} < \cdots < M_{K,a} = M_{1,b}. \tag{6}$$

Whereas the decoding reconstruction is usually at the center of the interval:

$$r_b = (M_{0,b} + M_{1,b})/2 \tag{7}$$

and

$$r_{k,a} = (M_{k,a} + M_{k+1,a})/2, \quad k = 0, \cdots, K-1. \tag{8}$$

For the coefficient with an actual value x and coded into subinterval $(M_{k,a}, M_{k+1,a})$, the distortion before and after coding is $(x - r_b)^2$ and $(x - r_{k,a})^2$, respectively, as illustrated through Fig. 8. Let $p(x)$ be the *a priori* probability distribution of the coding symbol within the interval $(M_{0,b}, M_{1,b})$, which is normalized so that the probability of the entire interval is equal to 1:

$$\int_{M_{0,b}}^{M_{1,b}} p(x)\, dx = 1. \tag{9}$$

The average distortion decrease can be calculated as a weighted average of the distortion decrease over the coding interval

$$E[\Delta D_i] = \sum_{k=0}^{K-1} \int_{M_{k,a}}^{M_{k+1,a}} [(x - r_b)^2 - (x - r_{k,a})^2] p(x)\, dx \tag{10}$$

while the average coding rate is the entropy of the coding subintervals

$$E[\Delta R_i] = \sum_{k=0}^{K-1} -n_k \log_2 n_k \quad \text{with} \quad n_k = \int_{M_{k,a}}^{M_{k+1,a}} p(x)\, dx. \tag{11}$$

In the case the candidate bit b_{n_i} undergoes significance identification, coefficient w_i is insignificant and is within interval $(-2T_{n_i}, 2T_{n_i})$ before the coding of b_{n_i}, where $T_{n_i} = 2^{-n_i}$ is the quantization step size determined by the coding layer n_i. After the coding of b_{n_i}, coefficient w_i may be negatively significant with interval $(-2T_{n_i}, -T_{n_i}]$, positively significant with interval $[T_{n_i}, 2T_{n_i})$, or still insignificant with interval $(-T_{n_i}, T_{n_i})$. We thus have the following three possible segments after significance identification with segment boundaries:

$$\begin{aligned} &M_{0,b} = M_{0,a} = -2T_{n_i}, \quad M_{1,a} = -T_{n_i}, \\ &M_{2,a} = T_{n_i}, \quad \text{and} \quad M_{3,a} = M_{1,b} = 2T_{n_i}. \end{aligned} \tag{12}$$

The decoding reconstruction value before significance identification is

$$r_b = 0. \tag{13}$$

The decoding reconstruction values of each segment after significance identification are:

$$r_{0,a} = -1.5T_{n_i}, \quad r_{1,a} = 0, \quad r_{2,a} = 1.5T_{n_i}, \text{ respectively.} \tag{14}$$

Because the probability of significance p_i can be formulated as

$$p_i = \int_{-2T_{n_i}}^{-T_{n_i}} p(x)\, dx + \int_{T_{n_i}}^{2T_{n_i}} p(x)\, dx. \tag{15}$$

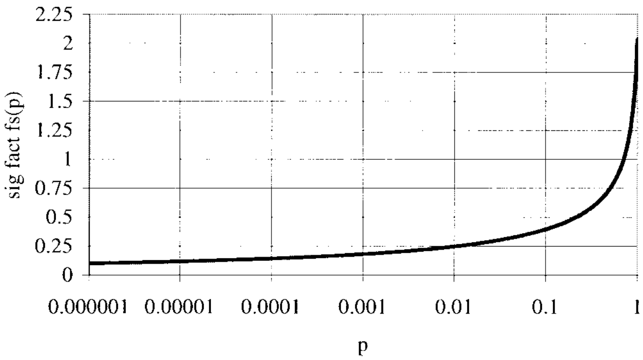


Fig. 9. Rate-distortion slope modification factor for significance identification.

Assuming that the *a priori* probability distribution within the significance interval is uniform

$$p(x) = \frac{p_i}{2T_{n_i}}, \quad \text{for } T_{n_i} < |x| < 2T_{n_i}. \tag{16}$$

By substituting (12)–(14) and (16) into (10) and (11), we may calculate the average distortion decrease and average coding rate for significance identification as

$$E[\Delta D_i] = p_i 2.25 T_{n_i}^2 \tag{17}$$

$$\begin{aligned} E[\Delta R_i] &= (1-p_i)[-\log_2(1-p_i)] + 2\frac{p_i}{2}\left(-\log_2\frac{p_i}{2}\right) \\ &= p_i + H(p_i) \end{aligned} \tag{18}$$

where $H(p_i)$ is the entropy of a binary symbol with probability of one equals to p_i:

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p). \tag{19}$$

Note that the average distortion (17) is not related to the probability density function within insignificance interval $(-T_{n_i}, T_{n_i})$, because within that interval the decoding values before and after coding are both zero, and thus the distortion change is zero. It is straightforward to derive the expected R-D slope for significant identification from (17) and (18) as

$$\lambda_{i,\text{sig}} = \frac{E[\Delta D_i]}{E[\Delta R_i]} = \frac{2.25T_{n_i}^2}{1 + H(p_i)/p_i} = f_s(p_i)T_{n_i}^2. \tag{20}$$

Function $f_s(p)$ is the significance R-D slope modification factor defined as

$$f_s(p) = \frac{2.25}{1 + \frac{H(p)}{p}}. \tag{21}$$

It is plotted in Fig. 9. Apparently, the symbol with higher probability of significance has a larger R-D slope and is thus favored to be coded first. The calculation of the R-D slope is only based on the coding layer n_i and the probability of significance p_i, which is in turn estimated through the QM-coder state.

We may similarly derive the expected R-D slope for refinement coding, where coefficient w_i is refined from interval $[S_i, S_i + 2T_{n_i})$ to one of the two segments $[S_i, S_i + T_{n_i})$ or $[S_i + T_{n_i}, S_i + 2T_{n_i})$. $T_{n_i} = 2^{-n_i}$ is again the quantization step size determined by the coding layer n_i, and S_i is the start of

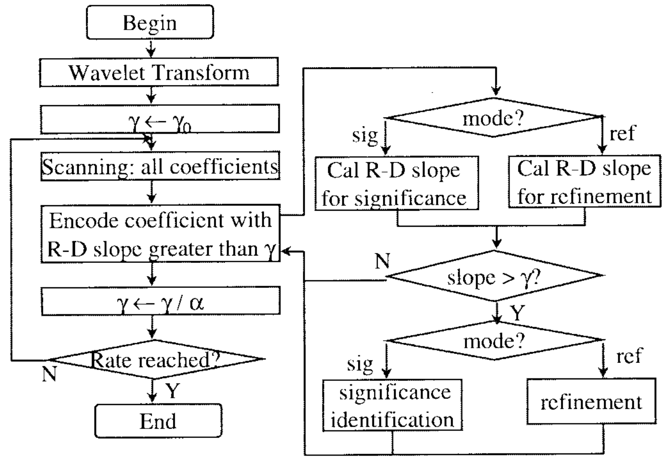


Fig. 10. Flowchart of rate-distortion optimized embedding.

the refinement interval which is determined by the previously coded bits of coefficient w_i. The segment boundaries are

$$\begin{aligned} &M_{0,b} = M_{0,a} = S_i, \quad M_{1,a} = S_i + T_{n_i}, \quad \text{and} \\ &M_{2,a} = M_{1,b} = S_i + 2T_{n_i} \end{aligned} \tag{22}$$

and the corresponding decoding reconstruction values are

$$\begin{aligned} &r_b = S_i, \quad r_{0,a} = S_i + 0.5T_{n_i}, \quad \text{and} \\ &r_{1,a} = S_i + 1.5T_{n_i}. \end{aligned} \tag{23}$$

Assuming that the *a priori* probability distribution within interval $[S_i, S_i + 2T_{n_i})$ is uniform

$$p(x) = \frac{1}{2T_{n_i}}, \quad \text{for } S_i < x < S_i + 2T_{n_i} \tag{24}$$

the average distortion and coding rate for refinement coding can be calculated as

$$E[\Delta D_i] = 0.25T_{n_i}^2 \tag{25}$$

$$E[\Delta R_i] = 1. \tag{26}$$

The expected R-D slope for refinement coding is thus

$$\lambda_{i,\text{ref}} = \frac{E[\Delta D_i]}{E[\Delta R_i]} = 0.25T_{n_i}^2. \tag{27}$$

Comparing (20) and (27), it is apparent that for the same coding layer n_i, the R-D slope of refinement coding is smaller than that of significance identification whenever the significance probability p_i is above 0.01. This agrees with the result of [9] that in general the significance identification should be placed before the refinement coding.

We may also model the *a priori* probability distribution of coefficient w_i to be Laplacian. In that case, the R-D slope for significance identification and refinement becomes

$$\lambda_{i,\text{sig}} = f_s(p_i)g_{\text{sig}}(\sigma, T_{n_i})T_{n_i}^2 \tag{28}$$

$$\lambda_{i,\text{ref}} = 0.25g_{\text{ref}}(\sigma, T_{n_i})T_{n_i}^2 \tag{29}$$

where σ is the variance of Laplacian distribution which can also be estimated from the already coded coefficients, and

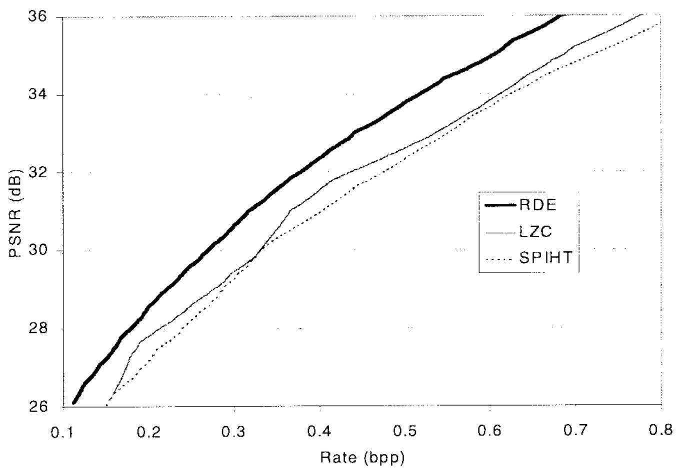


Fig. 11. Rate-distortion curve of RDE, LZC, and SPIHT.

$g_{\mathrm{sig}}(\sigma, T)$ and $g_{\mathrm{ref}}(\sigma, T)$ are Laplacian modification factors in the form of

$$g_{\mathrm{sig}}(\sigma, T) = \frac{1}{2.25}\left(0.75 + \frac{3\sigma}{T} - \frac{3e^{-T/\sigma}}{1-e^{-T/\sigma}}\right) \tag{30}$$

$$g_{\mathrm{ref}}(\sigma, T) = 4\left(0.75 + \frac{2\frac{\sigma}{T}e^{-T/\sigma} - \frac{\sigma}{T}(1+e^{-2T/\sigma})}{1-e^{-2T/\sigma}}\right). \tag{31}$$

However, experiments show that the additional performance improvement provided by the Laplacian probability model is minor. Since the uniform probability model is much simpler to implement, it is used throughout the experiment.

Because the probability of significance p_i is discretely determined by the QM-coder state, and the quantization step size T_{n_i} associated with the coding layer n_i is also discrete, both the R-D slope of significance identification (20) and refinement (27) have a discrete number of states. For fast calculation, (20) and (27) may be precomputed and stored in a table indexed by the coding layer n_i and the QM-coder state. Computation of the R-D slope is thus only a lookup table operation. The R-D slope of refinement needs one entry per coding layer. The R-D slope of significance identification needs two entries per coding layer and per QM-coder state, as each QM-coder state may correspond to the probability of significance p_i (if the most probable symbol is 1) or the probability of insignificance $1 - p_i$ (if the most probable symbol is 0). Therefore, the total number of entries M in the lookup table is

$$M = 2KL + K \tag{32}$$

where K is the maximum coding layer, L is the number of states in the QM-coder. In the current implementation, there are a total of 113 states in the QM-coder, and a maximum of 20 coding layers. This brings up a lookup table of size 4540.

(a)

(b)

(c)

Fig. 12. Original image of (a) boat, (b) gold, and (c) Lena.

D. Coefficient Selection

The second key step in RDE is selecting the coefficient with the maximum expected R-D slope. This may be done through an exhaustive search or sorting over all candidate bits. However, such approach will be computational expensive. In this implementation, a threshold based approach is used. The concept is to setup a series of decreasing R-D slope thresholds $\gamma_0 > \gamma_1 > \cdots > \gamma_n > \cdots$, and to scan the whole image repeatedly. The symbols with R-D slope between γ_n and γ_{n+1}

Fig. 13. Experimental results of the Barbara image. (a) Original. Coded image at 0.125 b/pixel with (b) RDE 26.1 dB.

Fig. 13. (*Continued.*) Experimental results of the Barbara image. Coded image at 0.125 b/pixel with (c) LZC 25.3 dB and (d) SPIHT 25.1 dB.

TABLE II
COMPARISON OF RDE VERSUS LZC AND SPIHT ON IMAGES LENA, BARBARA, BOATS, AND GOLD

Image	Rate (bpp)	LZC PSNR(dB)	SPIHT PSNR(dB)	RDE PSNR(dB)	RDE Gain vs LZC(dB)	RDE Gain vs SPIHT(dB)
Lena	1	40.1	40.4	40.3	0.2	-0.1
	0.5	37.1	37.2	37.2	0.1	0.0
	0.25	34.1	34.1	34.2	0.1	0.1
	0.125	31.1	31.1	31.3	0.2	0.2
Barbara	1	37.5	37.6	38.1	0.6	0.5
	0.5	32.6	32.3	33.1	0.5	0.8
	0.25	28.6	28.2	29.1	0.5	0.9
	0.125	25.3	25.1	26.1	0.8	1.0
Boats	1	41.2	41.1	41.6	0.4	0.5
	0.5	36.9	36.5	37.0	0.1	0.5
	0.25	33.1	32.5	33.2	0.1	0.7
	0.125	29.9	29.3	30.0	0.1	0.7
Gold	1	37.5	37.6	37.7	0.2	0.1
	0.5	34.0	34.1	34.3	0.3	0.2
	0.25	31.4	31.3	31.6	0.2	0.3
	0.125	29.3	29.0	29.5	0.2	0.5

are encoded at iteration n. The threshold based rate-distortion optimization sacrifices a little bit performance as symbols with R-D slope between γ_n and γ_{n+1} cannot be distinguished. However, the coding speed is much faster as the search for the maximum R-D slope is avoided.

The entire coding operation of RDE can be shown in Fig. 10, where the left part shows the main operation flow, and the right part shows a blown up of R-D slope calculation and symbol coding. Since the symbols of significance identification and refinement are treated differently, they are depicted at separate branches in R-D slope calculation and symbol coding. The operation can be described step by step as follows.

1) *Initialization*: The image is decomposed by the wavelet transform. The initial R-D slope threshold γ is set to γ_0, with

$$\gamma_0 = \frac{1}{16} T_0^2. \tag{33}$$

2) *Scanning*: The entire image is scanned top-down from the coarsest scale to the finest scale. Within each scale, subbands are coded sequentially with order LL (if the most coarse scale), LH, HL and HH. The coder follows the raster line order within the subband.
3) *Calculation of the expected R-D slope*:
The expected R-D slope is calculated for the candidate bit of each coefficient. Depending on whether the coefficient is significant, the expected R-D slope is calculated according to (20) or (27). Note that the calculation of the R-D slope is only a lookup table operation indexed by the QM-coder state and the coding layer n_i.
4) *Coding decision*: The calculated R-D slope is compared with the current threshold γ. If the R-D slope is smaller than γ of the current iteration, the coding proceeds to the next coefficient. Only the candidate bit with R-D slope greater than γ is encoded.
5) *Coding of the candidate bit*: Depending again on whether the coefficient is significant, the candidate bit is coded with significance identification or refinement. The QM-coder with context designated in Fig. 7 is used for significance identification. A fixed probability arithmetic coder is used to encode the sign and refinement. The sign bit is encoded right after the coefficient becomes significant.
6) *Coding rate check*: The coder checks if the assigned coding rate is reached. If not, the coder goes back to step 3.
7) *Iteration*: After the entire image has been scanned, the R-D slope threshold is reduced by a factor of α:

$$\gamma \leftarrow \gamma/\alpha. \tag{34}$$

In the current implementation, α is set to 1.25. The coder then goes back to step 2 and scans the image again.

III. EXPERIMENTAL RESULTS

Extensive experiments are performed to compare RDE with other existing algorithms. The test images are Lena, boats, gold, and Barbara, which are shown in Figs. 12 and 13(a). The image Lena is of size 512 $\times$ 512, all other images are of size 720 $\times$ 576. The images are decomposed by a 5-level 9-7 tap biorthogonal Daubechies filter with symmetric boundary extension [14]. It is then compressed by the layered zero coding (LZC, proposed by Taubman and Zakhor in [3]), the set partitioning in hierarchical trees (SPIHT, proposed by Said and Pearlman in [2]), and the rate-distortion optimized embedding (RDE), respectively. The well-respected SPIHT coder is used here as a reference of the state of the art coder. LZC uses the same method for context-coding of bitplanes as that of RDE. In essence, RDE shuffles the bitstream of LZC and improves its embedding performance. The comparison between RDE and LZC therefore shows the particular improvement of R–D optimization. We set the initial probability of QM-coder in RDE to be equilibrium, (i.e., the probabilities of one of all contexts are equal to 0.5). No prestatistics of image is used. The compression ratio in the experiment is chosen to be 8 : 1 (1.0 b/pixel), 16 : 1 (0.5 b/pixel), 32 :1 (0.25 b/pixel) and 64 : 1 (0.125 b/pixel). Since all three coders are embedded coders, the coding can be stopped at the exact bit rate.

The comparison results is shown in Table II, where the coding rate is shown in column 2, the peak signal-to-noise ratio (PSNR) of LZC and SPIHT are shown in columns 3 and 4, and the PSNR of RDE and its gain over LZC and SPIHT are shown in columns 5, 6, and 7, respectively. We also plot the R-D performance curve of the Barbara image in Fig. 11, where the R-D curves of RDE, LZC and SPIHT are plotted with the bold, solid and dotted line, respectively. The R-D curve in Fig. 11 is dense, as we calculate one PSNR point every increment of few bytes. RDE apparently outperforms both LZC and SPIHT. The performance gain of RDE over LZC ranges from 0.1 to 0.8 dB, with an average of 0.3 dB. The gain shows the performance improvement achieved by the rate-distortion optimization. From Fig. 11, it can be observed that the R-D performance curve of RDE is also much smoother than that of LZC. The effect is a direct result of rate-distortion optimization. With the embedded bitstream organized by decreasing rate-distortion slope, the slope of the resultant performance curve decreases gradually, results in the smooth looking R-D curve of RDE. Unlike Fig. 1, the RDE still outperforms LZC at high bit rate due to the fact that a floating 9-7 wavelet filter is used in the experiment. There are infinite bitplanes in the transform coefficients, so gain still can be observed at high bitrate. In another aspect, the context adaptive arithmetic coder in use adjusts its symbol probability estimation based on the past coding pattern under the same context, its performance is thus very slightly affected by the order of coding, which also attributes to the performance difference of RDE and LZC at high bitrate. The performance gain of RDE over SPIHT ranges from -0.1 to 1.0 dB, with an average of 0.4 dB.

The RDE, LZC, and SPIHT coded Barbara images at 0.125 b/pixel are shown in Fig. 13. The subjective appearances of the three images are close. Although the RDE coded Barbara does reveal a little more details in the texture regions, especially around the tie and trousers of Barbara. Due to the use of R-D optimization, RDE allocates the bit budget smartly and encodes the wavelet coefficients a little better, which results in the slightly improved subjective appearance in Fig. 13.

IV. Conclusions and Extensions

In this paper, we propose a rate-distortion optimized embedded coder (RDE). RDE substantially improves the performance of embedding at every possible truncation point by coding first the symbol with the steepest R-D slope. That is, at each coding instance, RDE spends the bits to the coding symbol with the largest distortion decrease per coding bit. For synchronization between the encoder and the decoder, RDE uses the expected R-D slope, which can be calculated by both the encoder and the decoder. It also takes advantage of the probability estimation table of the QM-coder so that the calculation of the R-D slope is just one lookup table operation.

Currently, the distortion used in RDE is measured by the mean square error (MSE). However, MSE does not reflect the visual quality of the image. We are working toward a visual weighted RDE which, instead of optimizing MSE, optimizes the visual quality at each truncation point. Research is also being conducted to calculate the expected R-D slope for symbols of a group of bits, so that RDE can be extended to the embedded zerotree wavelet coding (EZW) or the set partitioning in hierarchical trees (SPIHT). Another area of improvement is coding postprocessing [16], which may be used to reduce the ringing artifact and improve the subjective quality of the decoded image.

References

[1] J. Shapiro, "Embedded image coding using zerotree of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

[2] A. Said and W. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, June 1996.

[3] D. Taubman and A. Zakhor, "Multirate 3-D subband coding of video," *IEEE Trans. Image Processing*, vol. 3, pp. 572–588, Sept. 1994.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991, ch. 13.

[5] L.-J. Lin, A. Ortega, and C.-C. J. Kuo, "Rate control using spline-interpolated R-D characteristics," in *Proc. SPIE: Visual Communication and Image Processing*, Orlando, FL, Apr. 1996, vol. 2727, pp. 111–122.

[6] K. Ramchandran, A. Ortega, and M. Vetterli, "Bit allocation for dependent quantization with applications to multiresolution and MPEG video coders," *IEEE Trans. Image Processing*, vol. 3, pp. 533–545, Sept. 1994.

[7] K. Ramchandran, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, pp. 160–175, Apr. 1993.

[8] Z. Xiong and K. Ramchandran, "Wavelet packet-based image coding using joint space-frequency quantization," in *Proc. 1st IEEE Int. Conf. Image Processing*, Austin, TX, Nov. 13–16, 1994.

[9] J. Li, P. Cheng, and C.-C. J. Kuo, "On the improvements of embedded zerotree wavelet (EZW) coding," in Proc. *SPIE: Visual Communication and Image Processing*, Taipei, Taiwan, R.O.C., May 1995, vol. 2601, pp. 1490–1501.

[10] W. Pennebaker and J. Mitchell, IBM, "Probability adaptation for arithmetic coders," U.S. Patent 5 099 440, Mar. 24, 1992.

[11] D. Duttweiler and C. Chamzas, "Probability estimation in arithmetic and adaptive Huffman entropy coders," *IEEE Trans. Image Processing*, vol. 4, pp. 237–246, Mar. 1995.

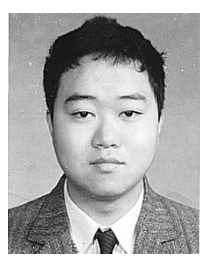
[12] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York: Van Nostrand Reinhold, 1993.

[13] J. L. Mitchell, *MPEG Video Compression Standard*. London, U.K.: Chapman & Hall, 1997.

[14] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205–230, Apr. 1992.

[15] J.-K. Li, J. Li, and C.-C. Jay Kuo, "Layered DCT still image compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 440–443, Apr. 1997.

[16] J. Li and C.-C. Jay Kuo, "Coding artifact removal with multiscale postprocessing," in *Proc. IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997, vol. 1, pp. 45–48.



Jin Li (SM'94–M'95) received the M.S. and Ph.D. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1991 and 1994, respectively.

Since October 1996, he has been Member of Technical Staff at the Digital Video Division, Sharp Laboratories of America, Camas, WA. From 1994 to 1996, he served as a Research Associate in the Integrated Media Science Center, Department of Electrical Engineering—Systems, University of Southern California, Los Angeles. He has been active in the areas of image processing and multimedia communication during the past five years and has published over 30 technical papers in various journals and conferences on the topics of fast motion estimation, advanced motion compensation, embedded wavelet coding, rate-distortion codec model, wavelet packet coding, scalable coding, fractal, vector quantization, image enhancement, coding preprocessing and postprocessing, coding distortion measure, and multiresolution graphic model and geometry compression.

Dr. Li serves as an area editor for *the Journal of Visual Communication and Image Representation*. He received the Distinctive Ph.D. Award from Tsinghua University in 1994 and the Young Investigator Award from SPIE and IS&T Visual Communication and Image Processing in 1998.

Shaw-Min Lei (S'87–M'88–SM'95) received the B.S. and M.S. degrees from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1980 and 1982, and the Ph.D. degree from the University of California, Los Angeles, in 1988, all in electrical engineering.

From 1982 to 1984, he was an Instructor of electrical engineering at Naval Academy, Taiwan. From August 1988 to October 1995, he was with Bellcore, Red Bank, NJ, where he had worked in both video compression and wireless communication areas. He is currently with Sharp Laboratories of America, Camas, WA, where he is a manager of the Video Coding and Communication Group. His current research interests include video/image compression, coding, processing and communication, multimedia communication, wireless communicaions, data compression, and error control coding. He is the author or co-author of more than 30 technical papers.

Dr. Lei is co-recipient of the Best Paper Award of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 1993. He has been awarded three patents.