A Wavelet-Based Two-Stage Near-Lossless Coder

Sehoon Yea, *Student Member, IEEE*, and William A. Pearlman*, *Fellow, IEEE*

***Abstract*— In this paper, we present a two-stage near-lossless compression scheme. It belongs to the class of "lossy plus residual coding" and consists of a wavelet-based lossy layer followed by arithmetic coding of the quantized residual to guarantee a given L^{∞} error bound in the pixel domain. We focus on the selection of the optimum bit rate for the lossy layer to achieve the minimum total bit rate. Unlike other similar lossy plus lossless approaches using a wavelet-based lossy layer, the proposed method does not require iteration of decoding and IDWT (Inverse Discrete Wavelet Transform) in succession to locate the optimum bit rate. We propose a simple method to estimate the optimal bit rate, with a theoretical justification based on the critical rate argument from the rate-distortion theory and the independence of the residual error.**

EDICS Category: COD-OTHR, COD-LSSI, MOD-SRCE

I. Introduction

Image compression techniques are usually classified into two categories—lossy and lossless. The former achieves high compression ratios by virtue of various strategies effectively retaining visually relevant information, while the latter achieves rather modest compression ratios as a result of conservatively maintaining all the information. In some applications such as medical and scientific imaging, lossless compression is traditionally used for fear of losing critical information. It is often argued that loss of seemingly minor details subject to a subsequent medical diagnosis or scientific postprocessing can lead to significant legal liabilities or incorrect diagnoses. On the other hand, many in the medical image compression community argue that lossy compression of medical images is necessary and helpful in the long run [6]. They argue that technology itself must be considered because the overwhelming quantity of medical data requires the remote storage of hard-copy films, which frequently results in loss or damage and always requires significant time to locate and transfer. Thus, one may well argue that, rather than thinking in terms of the dichotomy between lossless and lossy, we must seek out alternative ways to effectively compromise the conflicting requirements. In the face of this reality, more and more users in medical and scientific communities are accepting "near-lossless" methods as a means of trading off between compression ratios and distortion so that higher compression ratios can be achieved with small enough distortion to ensure sufficient accuracy for specific purposes.

There are many excellent wavelet-based lossy image compression schemes under L^2 norm in the literature [1], [3], [14], [15], [16]. However, none of these coders which are L^2 approximation schemes can offer a tight bound on the maximum reconstruction error of each pixel due to the global nature of the L^2 norm which averages the reconstruction error over the entire image. High enough PSNR may mean that the reconstructed image is visually indistinguishable from the original. But for many applications, high image fidelity means more than satisfying the eyes, as discussed above. Specifying how much and what type of "near-lossless" distortion can be allowed for a specific purpose is a largely open problem and varies from application to application. The most popular definition of "near-losslessness" is that no pixel is changed in magnitude by more than δ gray levels compared with the original, where δ is a nonnegative integer indicating the error tolerance. Let us denote an $N_x \times N_y$ gray scale image by a two-dimensional array $I(x, y)$ of integers, where $0 \leq x < N_x$ and $0 \leq y < N_y$, and a compressed version (an approximation) of it by $\tilde{I}(x, y)$. Then the purpose of near-lossless compression is to obtain an $\tilde{I}(x, y)$ such that the following relation is guaranteed.

$$\|I - \tilde{I}\|_{\infty} = \max_{0 \leq x < N_x, 0 \leq y < N_y} |I(x, y) - \tilde{I}(x, y)| \leq \delta, \quad (1)$$

where δ is the error tolerance. We consider a near-lossless coding problem with a two-stage coder [7], where the amount of distortion, as defined by (1), is guaranteed quantitatively. The first stage consists of a wavelet-based progressive lossy coder such as SPIHT or JPEG-2000. For the second stage, we use an arithmetic coder to encode the residual.

Since it is difficult to derive a meaningful relation between distortions in the wavelet domain and in the pixel domain in the L^{∞} error sense, pixel domain techniques such as predictive coding have been the most popular choice in practice [11], [13]. Furthermore, since prediction-based coders are very competitive in lossless and high bit-rate coding, it is expected to be so in many near-lossless coding scenarios, when δ is small (typically 1 or 2). Unfortunately, this is no longer true as δ gets larger since the quality of prediction deteriorates as will be discussed in Section II-A. Also, predictive coders do not offer the advantage of progressive transmission as with wavelet-based coders. Thus it is highly desirable to combine the advantages of both worlds to provide the convenience of embedded bit-stream along with the guaranteed maximum distortion in the L^{∞} sense in the pixel domain. Indeed, there have been various attempts at achieving such a goal in the near-lossless/lossless coding problem. In [7], the authors

This research was carried out at Rensselaer Polytechnic Institute and was supported by the National Geospatial-Intelligence Agency under Award No. NMA201-00-1-2008.

S. Yea is now with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA; E-mail: yea@merl.com.

W. A. Pearlman is with the Elec., Comp. & Syst. Engrg. Dept., Rensselaer Polytechnic Institute, Troy, NY 12180-3590,USA; E-mail: pearlw@ecse.rpi.edu.

compare various near-lossless schemes with a focus on the wavelet-based two-stage scheme. A similar approach is taken in [9] with a focus on L^∞ error scalability. Marpe *et al.* [8] also propose to use a two-stage wavelet-packet based approach to lossy/lossless coding. However, none of the aforementioned works addresses the problem of bit rate assignment between the two encoding stages except through exhaustive search by repeatedly encoding the residuals generated by subtracting lossy reconstruction images at various lossy layer coding rates from the original.

In this paper, we elaborate on the choice of efficient bit rate assignment for the wavelet-based first-stage lossy-layer coder in the two-stage near-lossless setting. It does not directly rely on any analytical model for the source, yet works reasonably well for most real sources. In Section II, we give a review of relevant approaches proposed in the literature. Section III describes the proposed near-lossless coding method, followed by discussions in Section IV on some key observations underlying our proposed approach. Then we proceed to show experimental results in Section VI and conclude the paper in Section VII.

II. Review of Literature

In this section, we provide an overview of relevant literatures in near-lossless compression.

A. Predictive-Coding Based Approach

The early work of Chen and Ramabadran [5] makes use of differential pulse code modulation (DPCM) and a uniform scalar quantizer to ensure the maximum error of ±1. Ke and Marcellin [4] generalized it to any discrete L^∞ error of $\pm\delta$. The authors used a DPCM coding scheme incorporating entropy-minimization of the quantized prediction errors. Figure 1 shows a schematic diagram of a DPCM coder. The residual error e is obtained from the predicted image $\hat{I}$ as :

$$e = I - \hat{I}, \tag{2}$$

and quantized to $\hat{e}$ providing a near-lossless reconstruction image $\tilde{I}$:

$$\tilde{I} = \hat{I} + \hat{e}. \tag{3}$$

Thus, we have

$$\begin{aligned} I - \tilde{I} &= I - \hat{I} - \hat{e} \\ &= e - \hat{e}. \end{aligned} \tag{4}$$

In other words, the reconstruction error is equal to the quantization error introduced by the quantizer in the feedback loop. We would like to mention two things. One is that it is trivial to upper bound the maximum reconstruction error in the framework of DPCM as obvious from (4). The other is that as the maximum error δ becomes larger, the performance of the DPCM-based near-lossless scheme will, in general, deteriorate since a larger δ means a coarser quantization which will lead to degraded predictor performance.

On the other hand, Wu *et al.* [11] proposed the near-lossless CALIC (Context-based, Adaptive Lossless-Image Codec) which could support L^∞ error-constrained compression. They extended their state-of-the-art nonlinear-predictive lossless coding scheme to near-lossless cases and achieved a very competitive bit rate performance given an L^∞ bound. They even incorporated a bias-cancellation technique to compensate for the degraded prediction with a large value of δ.

Finally, we have to mention JPEG-LS [13], the new ISO/ITU standard for lossless and near-lossless compression of continuous-tone images. It is based on the LOCO-I algorithm developed at HP Labs and has low computational complexity and memory requirements. It also uses context-based entropy coding (Golomb coding) of the quantized prediction residuals. It shows a competitive performance with other near-lossless coders especially at very high bit rates corresponding to small δ's.

However, the aforementioned predictive coding techniques are non-scalable. Thus, in many practical scenarios where progressive reconstruction of images is regarded as necessary, they are of limited utility.

B. Wavelet Based Approach

Since there is no simple relationship between the quantization error in the wavelet domain and the L^∞ error in the pixel domain, it is generally presumptuous to expect that an L^2-oriented wavelet coder will also perform well in terms of the L^∞ error. In fact, there is a rather complicated relationship between the quantization step-size for the wavelet coefficients and the bound on L^∞ error in the pixel domain [10], [12]. However, it is a very conservative one. For example, choosing the quantization step-size for the wavelet coefficient as predicted by the formula targeting the L^∞ error of 3 would usually result in the L^∞ error of 1 or 2. In other words, it leads to a significant “over-encoding”— use of unnecessarily high bit rate — for a given maximum error.

We will discuss three approaches to L^∞ error-constrained compression based on wavelet coders. The first approach centers on reducing the aforementioned conservatism in the derived relationship between quantization errors in the two domains. Karry *et al.* [10] proposed a filterbank-based image coding technique where at least a given percentage of the reconstruction errors are below the required L^∞ error. Their algorithm is based on the argument that the reconstruction error distribution resulting from encoding the wavelet coefficients tends to behave as a Gaussian. Alecu *et al.* [12] also relied on similar assumptions and proposed a lifting-based implementation to realize an L^∞ error scalable wavelet coder. However, judging from the experimental results they show in the papers, both of these approaches still tend to highly “over-encode” to achieve a certain L^∞ error bound.

Next, a very simple way to attaining near-lossless compression, especially with a small δ, is that of prequantization [7].

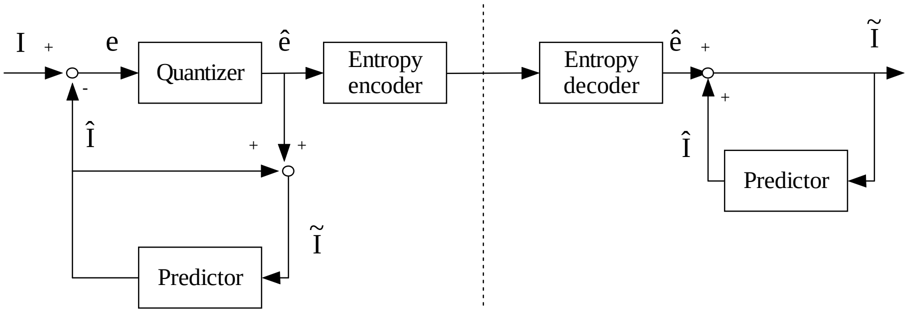


Fig. 1. Block diagram of a DPCM coder

At the encoder, the quantization index is generated as follows for each pixel:

$$l = \lfloor \frac{x+\delta}{2\delta+1} \rfloor \tag{5}$$

where x is a pixel value and $\lfloor \cdot \rfloor$ denotes the integer part of the argument. At the decoder, the pixel value is reconstructed as:

$$\hat{x} = l \times (2\delta + 1) \tag{6}$$

In this scheme, the quantization indices are losslessly compressed using an integer-to-integer wavelet coding scheme such as SPIHT with S+P filters [2] and JPEG-2000 with integer filters [17]. Obviously, it will lead to an embedded bitstream which incrementally refines in the L^2 sense up to the lossless reconstruction of the prequantized image. Two problems with this method can be pointed out. One is that, as with the predictive coding schemes, its performance degrades rapidly as δ becomes larger [7]. The other is that use of an integer wavelet usually implies somewhat compromised PSNR performance at lossy bit rates [19], [20], [21].

Finally, another way to reach the goal of near-lossless compression via wavelet is based on the so-called "lossy-plus-lossless" approach [7], [8], [9]. It is composed of a wavelet-based lossy coder in the first stage followed by a residual encoder in the second stage. In this approach, one usually quantizes the residual layer with a step size of 2δ and encode the quantization indices losslessly [7], [8]. Indeed, this is the one we chose to follow in this paper. In [7], Ansari *et al.* compare various near-lossless compression schemes with a special attention to wavelet-based schemes. Their work makes use of the lossy reconstruction for context-based entropy coding of the quantized residual layer. It was observed that a too low or high lossy layer bit rate leads to non-minimal total bit rate, suggesting the existence of optimal lossy rate. However, they had to iterate encoding at various lossy bit rates in order to find such an optimal lossy bit rate, because the bit rate of the context-based entropy coding of the residual could be, in general, known only after the actual coding on the pixel-domain residual is performed. Marpe *et al.* [8] proposed to use different wavelet-packets for the first or second stage in order to better fit the characteristics of the source or the residual to be compressed. Their method needs an exhaustive search for the best wavelet-packet basis for the lossy or the residual layer. Besides, it was limited only to lossless cases and its potential effectiveness in a near-lossless scenario remains to be seen. Certainly, all of the above-mentioned schemes may be useful when users specify the lossy layer encoding rate up to which they want to have a progressive (in L^2 error sense) bitstream and later want to receive the residual layer with as little bit rate as possible utilizing the correlation between the residual and the lossy layer (*scenario I*). However, there are situations where users have no *a priori* idea of an appropriate bit rate for the lossy layer. All that they need might be a reasonably good scalable (in L^2 error sense) lossy reconstruction, with an optional near-lossless residual layer with the minimum total bit rate (*scenario II*). The proposed scheme is the only one in the literature that attempts to meet the need of scenario II efficiently, as well as being a computationally very economical alternative with comparable performance to the above-mentioned schemes.

III. Proposed Method

A. Structure of the Two-Stage Near-Lossless Coder

Figure 2 shows a schematic block diagram of the two-stage near-lossless coder used in this work. In the figure, we assume any (bi-)orthogonal wavelet encoder based on successive bit-plane encoding. Let I and $\hat{I}$ represent the original image and the lossy reconstruction, respectively. At the encoder, the lossy reconstruction is subtracted from the original input image to form a residual e. It is readily confirmed from the figure that the following relations hold:

$$I = \hat{I} + e \tag{7}$$
$$\tilde{I} = \hat{I} + \hat{e} \tag{8}$$

where $\tilde{I}$ is the near-lossless reconstruction and $\hat{e}$ is the quantized residual. Therefore, it follows that $\|e - \hat{e}\|_\infty \leq \delta$ is equivalent to $\|I - \tilde{I}\|_\infty \leq \delta$ and the residual e is uniformly quantized as follows to guarantee the maximum error no larger than δ:

$$\hat{e}_{id} = \begin{cases} \lfloor \frac{e+\delta}{2\delta+1} \rfloor & , \quad e > 0 \\ \lfloor \frac{e-\delta}{2\delta+1} \rfloor & , \quad e < 0 \end{cases} \tag{9}$$

where $\lfloor \cdot \rfloor$ denotes the integer part of the argument. The generated quantization index $\hat{e}_{id}$ is then losslessly encoded by a simple arithmetic coder without incorporating any context model. Hence, $\hat{I}_{en}$ is transmitted as the lossy layer and losslessly coded quantization index, $\hat{e}_{id-en}$, is sent as the residual layer. At the decoder, we decode $\hat{e}_{id-en}$ by arithmetic decoding to yield $\hat{e}_{id}$ followed by a dequantizer defined as follows in order to guarantee $\|e - \hat{e}\|_\infty \leq \delta$.

$$\hat{e} = (2\delta + 1) \cdot \hat{e}_{id} \tag{10}$$

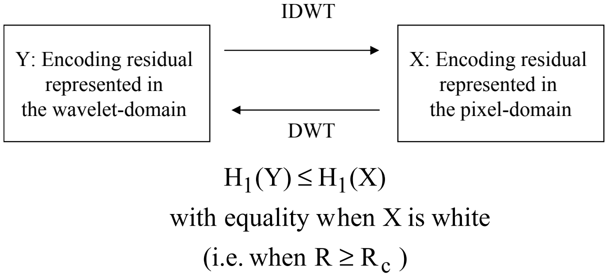
By adding the lossy reconstruction $\hat{I}$ and the dequantized residual $\hat{e}$, we obtain the final near-lossless reconstruction $\tilde{I}$ which guarantees $\|I - \tilde{I}\|_\infty \leq \delta$.

As already mentioned in Section II, in a two-stage near-lossless coding scheme, efficient encoding of the given quantization indices could be performed, for example, by building an efficient context model exploiting the correlation between the lossy reconstruction and the quantized residual indices [7], [9] or performing a search for a wavelet packet basis for the residual [8]. However, if we want the total (lossy plus residual) bit rate to be minimized for any given δ, then such approaches will not achieve this goal without exhaustive iteration with various lossy reconstruction bit rates. The method to be presented is the only one that determines this optimal first stage lossy rate during encoding without exhaustive iteration.

B. Critical Rate in Source Coding

Figure 3 shows the central idea of our method to determine the optimal first-stage bit rate. It is well-known from rate-distortion theory that when the bit rate (R) for the lossy reconstruction ($\hat{I}$) becomes larger than the 'critical rate' (R_c), the so-called 'backward test-channel' model holds and the encoding residual of a source becomes i.i.d. Gaussian [22], [24] under the MSE (mean squared error) distortion measure. Stated in a different way, when the encoding MSE distortion becomes smaller than the 'critical distortion (D_c)' for a given source, the source can be represented as a sum of two independent terms, one of which being an i.i.d. Gaussian noise, thereby rendering the sub-critical distortion-rate function to coincide with the Shannon Lower Bound (SLB). Now note that the wavelet domain residuals can be regarded as the wavelet transform of the pixel domain residual. Since the pixel domain encoding residual is i.i.d. when the lossy rate is above R_c, wavelet transforming it does not change its first order entropy [28]. This means that the first-order entropy $H_1(Y)$ of the wavelet-domain residual and that of its inverse, i.e. the pixel-domain residual's entropy $H_1(X)$ are equal beyond R_c. Above this critical rate R_c, the i.i.d. residual lacks the structure that a good lossy encoder (e.g., SPIHT) can take advantage of, so the coding efficiency of such a coder tends to become at most as good as and usually worse than that of first-order entropy coding. Thus, if we assume the use of a first-order entropy coder for the pixel-domain residual as was described in Section III-A, the total bit rate curve will have a lossy layer bit rate or a small flat region of bit rates where it attains its minimum.

For example, Figure 4(a) compares, with the Lena image, the first-order entropies $H_1(X)$ (bit rate of an actual first-order arithmetic coder for the pixel-domain residual X) and $H_1(Y)$ (the first-order entropy of the corresponding wavelet-domain residual Y calculated by the Algorithm 1 to be described in Section V) of the residuals in both domains as well as the corresponding total bit rates obtained by adding the lossy layer bit rate R on each of them. When R is below $R_c \approx 0.61$ bpp, the actual total bit rate curve ($H_1(X) + R$) moves downward as R increases since the use of SPIHT is more efficient than switching to a first-order pixel-domain residual coder in those bit rate regions due to the remaining memory in the source. However, as R approaches R_c, it reaches its minimum (or minima) and starts to increase when SPIHT becomes less efficient than the first-order entropy coder beyond R_c. Now, it is important to realize that it is also beyond this critical rate R_c that the residual rates (i.e. $H_1(X)$ and $H_1(Y)$) as well as the corresponding total bit rates in both domains start to coincide. In other words, the two total bit rate curves (i.e. 'actual' and 'estimate') as illustrated in Figure 4(a) start to coincide and move upward at the optimal lossy bit rate — ideally R_c. This phenomenon forms the basis of our method of identifying the optimal lossy rate point and, interestingly, holds similarly when we quantize the residual according to (9) for a non-zero L^∞ error δ. In the sequel, we will refer to the critical lossy bit rate either as R_c or $\hat{R}_c$ for lossless (δ=0) or near-lossless ($\delta > 0$) coding, respectively.



IDWT
Y: Encoding residual represented in the wavelet-domain
X: Encoding residual represented in the pixel-domain
DWT
H1(Y) ≤ H1(X)
with equality when X is white
(i.e. when R ≥ Rc)

Fig. 3. First-order entropies in the two domains

Finally, observe that our method can be seen as a way of finding out the 'critical rate' for practical sources. Although the notion of SLB has been proved to be useful and applicable to quite a general class of sources and problem settings [25], [26], [27], finding out the 'critical rate (R_c)' of a *practical* encoder achieving the 'critical distortion (D_c)' for a given *practical* source has been somewhat elusive. In this regard, our observations underlying the proposed method may shed some light from a practical viewpoint.

IV. Analysis of the Key Observations

In this section, an analysis is presented of the convergence phenomena regarding the probability distribution of encoding residuals in both the wavelet and the pixel domains. This will provide us with an 'under-the-hood' understanding of the key observations made in the previous section as well as a theoretical ground for the algorithm we propose in the next section.

A. Encoding Residual of Wavelet Coders

It is well-known that wavelet coefficients are well-modelled by a Generalized Gaussian Distribution (GGD) [29]. Since large coefficients are encoded first in typical wavelet coders, it is expected that the distribution of the *encoding residual* in

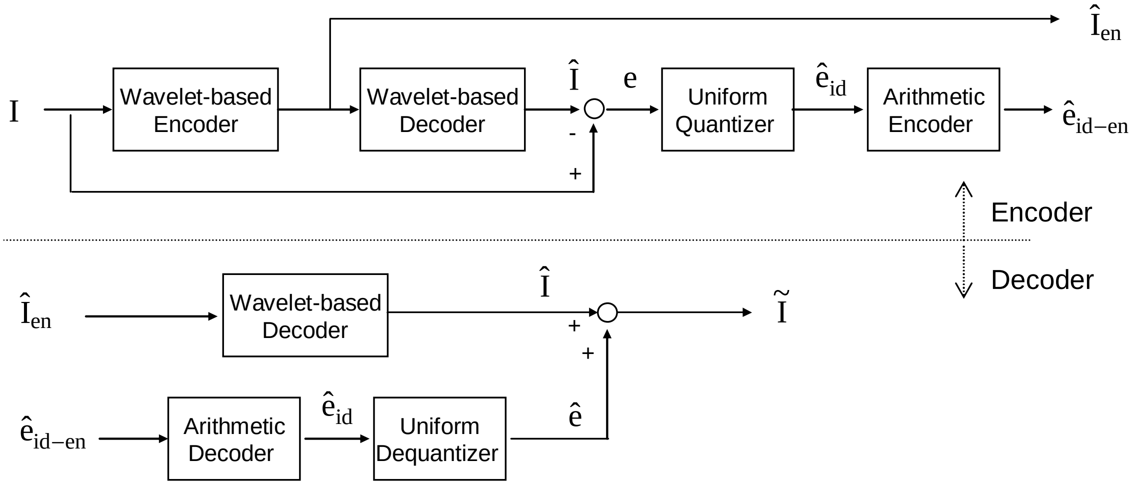
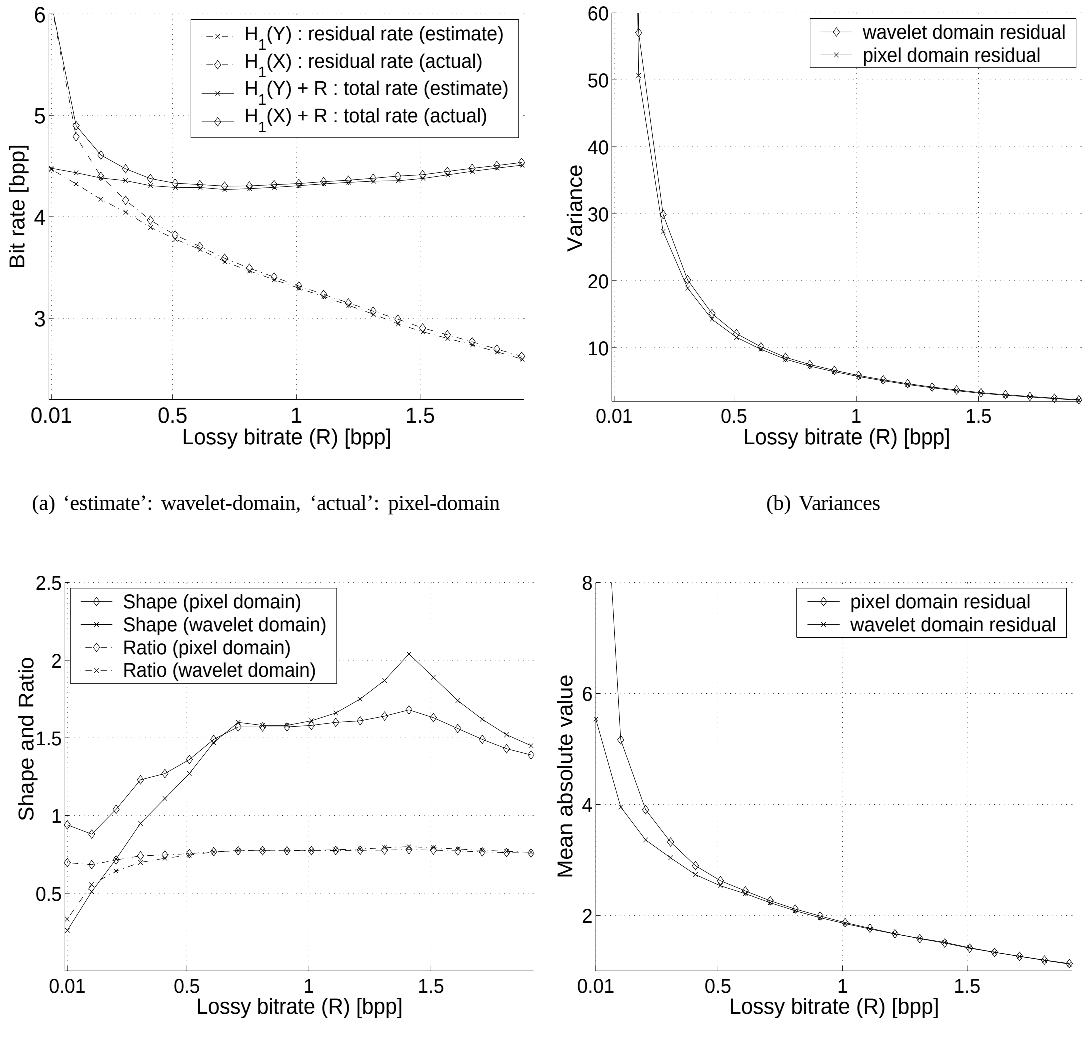


Fig. 2. Proposed Two-Stage Near-Lossless Wavelet Coder



(a) ‘estimate’: wavelet-domain, ‘actual’: pixel-domain

(b) Variances

(c) Ratio (η) & shape (ν) parameters

(d) Mean absolute values

Fig. 4. Bit rates and the GGD modelling parameters of residuals for Lena (‘estimate’ by wavelet-domain residual vs. ‘actual’ rates & parameters of pixel-domain residual)

the wavelet domain will still fit well into a GGD (usually with a shorter tail than the distribution of the wavelet coefficients itself). Indeed, it was observed through numerous experiments that the residual distributions both in the pixel and the wavelet domains could be reasonably well-modelled by GGD's.

A GGD $f(x)$ for a random variable X is defined as follows:

$$f(x) = ae^{-|bx|^{\nu}}, \tag{11}$$

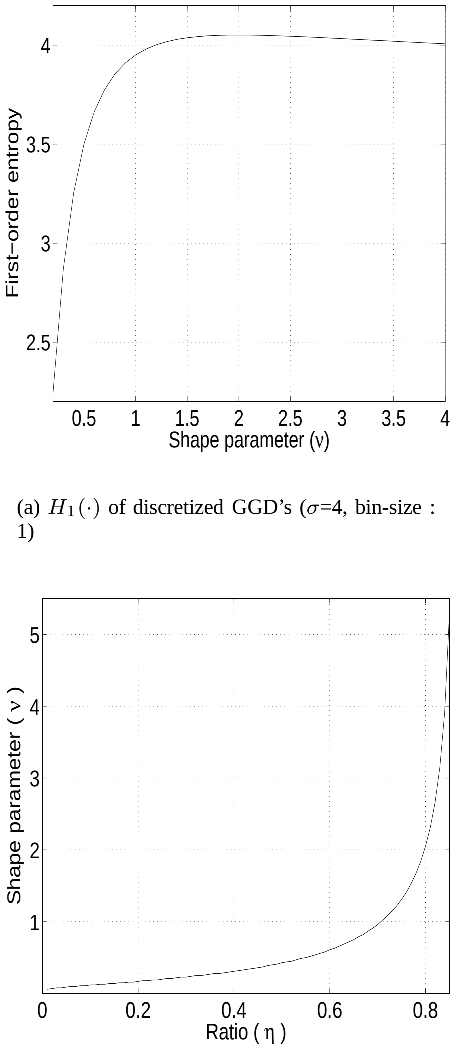
where $a \equiv \frac{b\nu}{2\Gamma(1/\nu)}$, $b \equiv \frac{1}{\sigma}\sqrt{\frac{\Gamma(3/\nu)}{\Gamma(1/\nu)}}$, $\Gamma()$ is the Gamma function defined as $\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt$, ν is the shape parameter and σ is the standard deviation of the source X. Thus a GGD is completely specified by σ and ν. Here the shape parameter ν can be estimated using the following equation [29].

$$\nu = G^{-1}(\eta), \tag{12}$$

where $G(x) \equiv \frac{\Gamma(2/x)}{\sqrt{\Gamma(1/x)\Gamma(3/x)}}$ and $\eta \equiv \frac{E(|X|)}{\sigma}$ is a normalized mean absolute value. Here, $\nu = 1$ and $\nu = 2$ correspond to Laplacian and Gaussian distributions, respectively. Also note that a Gaussian attains the maximum differential entropy among all special cases of GGD's given a variance. In general, as ν becomes large, it approaches the uniform distribution, while it becomes a sharply-peaked distribution as ν becomes small.

B. Convergence of Residual Distributions

Recall from Section III-B that the first-order entropies of the encoding residuals in both the pixel and the wavelet domains become the same when the encoding rate of the lossy layer (R) is above a certain 'critical' bit rate R_c. Now, notice from Figure 5(a) that, given a variance, the shape parameter (ν) of a GGD is related to its first-order entropy in a one-to-one manner when c is over the range of 0 to 2, which could cover practically all the possible shape parameter values for the encoding residual distributions in our experiments. This implies the shape parameters of the residuals in both domains will actually converge when R is above $R_c \approx 0.61$ bpp, assuming the variances are the same in both domains. Indeed, when we use the 9/7 biorthogonal filter for the SPIHT as a first-stage encoder, the variances of the residuals in the two domains are not exactly the same, but they are close enough to be regarded as the same, as illustrated in Figure 4(b) for Lena (see also [18] for a relevant discussion). Now, since a GGD is completely specified by its variance σ^2 and shape parameter ν, it follows from the convergence of c that the encoding residuals in both domains converge in probability distribution when the lossy layer encoding rate R is above R_c. Also observe that, as the function $G(\cdot)$ in (12) relating the shape parameter ν and the ratio η (the normalized mean absolute value) is one-to-one as shown in Figure 5(b), the values of η in both domains should also approach each other, as should the mean absolute values ($E(|X|)$) as a consequence. Figures 4(c) and 4(d) illustrate such a situation with the Lena image, where the shape parameters and the corresponding ratios (Figure 4(c)) as well as the mean absolute values (Figure 4(d)) of the residuals of both domains indeed converge when R is above $R_c \approx 0.61$ bpp. Finally, notice from Figure 5(b) that as η approaches 0.8, the corresponding value of the shape parameter ν changes abruptly beyond 2.0, although the corresponding change in its first-order entropy is very small as shown in Figure 5(a). This explains the poor match of the shape parameters around 1.41 bpp in Figure 4(c) as compared with the decent matches of corresponding 'ratios' around the same bit rate.



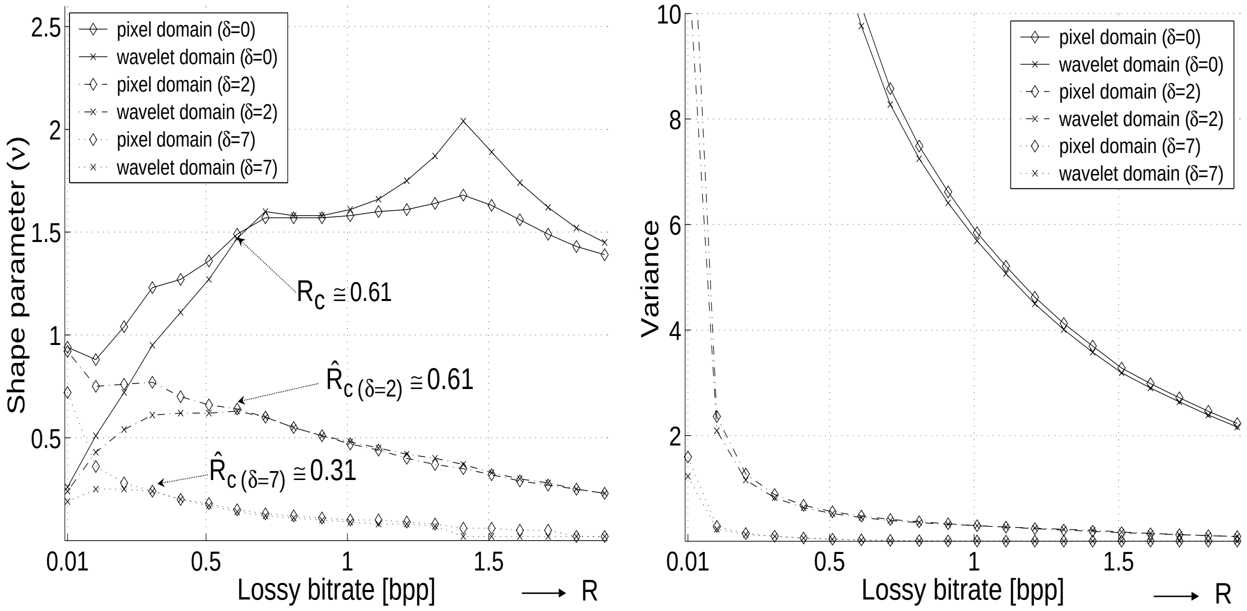
(a) $H_1(\cdot)$ of discretized GGD's (σ=4, bin-size : 1)

(b) Ratio (normalized mean-absolute-value) vs. shape (ν)

Fig. 5. Generalized Gaussian Distribution

C. Convergence of Quantized Residual Distributions

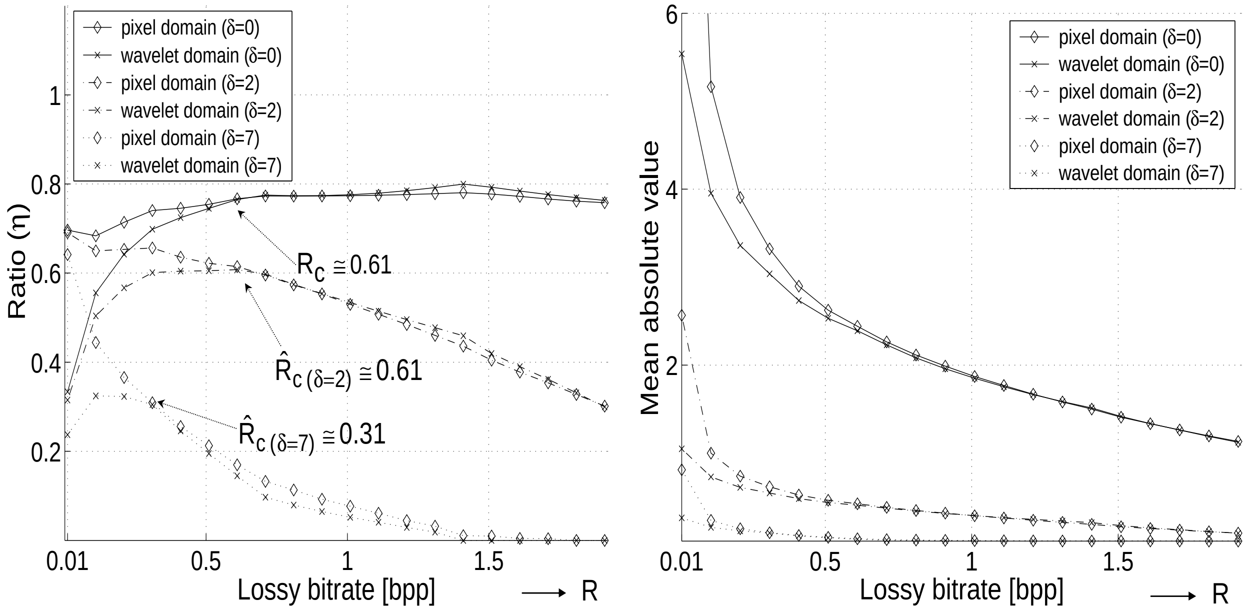
Now that we have discussed the convergence phenomena of the first-order entropies (Section III-B) and the residual probability distributions themselves (Section IV-B) when $\delta = 0$, it would be natural to ask if the same or similar results apply to the cases with $\delta > 0$. In other words, one may ask the following two questions: (In the sequel, the term 'quantized residual' and 'quantization indices' will be used interchangeably, since they are related in a one-to-one manner by (10).)



(a) Shape parameter ν in (12)

(b) Variance of pixel & wavelet domain quantized residuals

Fig. 6. Shape parameter and variance of the residuals for Lena



(a) Ratio η (normailized mean absolute value) in Section IV-A

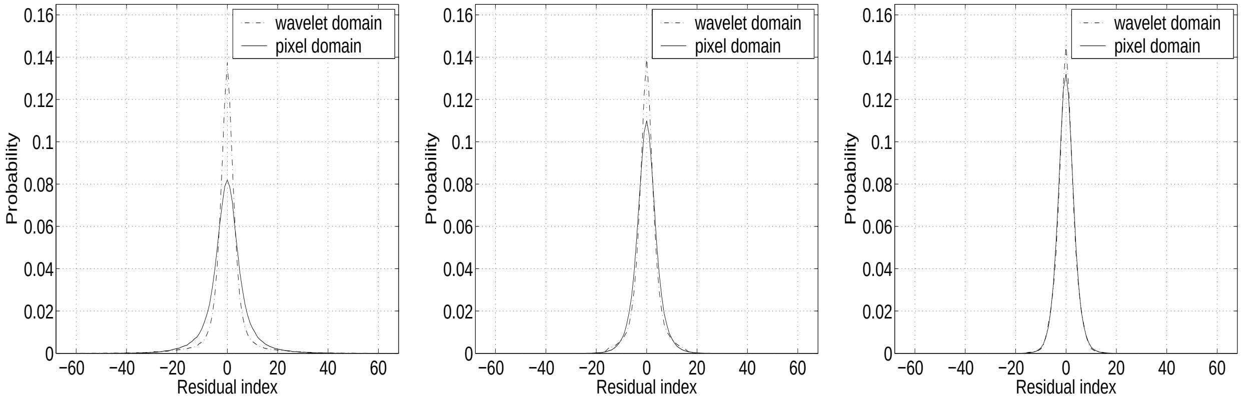
(b) Mean absolute values of pixel & wavelet domain quantized residuals

Fig. 7. Ratio and mean absolute value of residuals for Lena

1) Suppose the pixel-domain residual X needs to be quantized according to (9) for the case of $\delta > 0$ to yield the quantization indices $\hat{X}$ and we do the same quantization on the wavelet-domain residual Y to obtain $\hat{Y}$, will $H_1(\hat{X})$ and $H_1(\hat{Y})$ be the same beyond a certain rate $\hat{R}_c$?

2) Will the 'quantized' residuals in both domains converge in probability distribution beyond a certain rate $\hat{R}_c$?

First, observe that the answer to 1) is not obvious because the quantized residual indices in one domain can no longer be seen as a DWT (or IDWT) of those in the other due to the presence of quantization. Hence we cannot directly apply the 'critical rate' argument used in Section III-B for the case with $\delta = 0$. However, note that if the answer to



(a) δ=0 , lossy layer rate=0.11

(b) δ=0 , lossy layer rate=0.31

(c) δ=0 , lossy layer rate=0.51

Fig. 8. Convergence of residual distributions for Lena (δ=0)

2) is 'yes', it follows that the answer to 1) will be also in the affirmative since the identical probability distributions will have the same entropy. Therefore, we will try to address the second question, thereby drawing an answer to the first one as well. To begin with, we will show some experimental evidences suggesting the answer to 2) is in the affirmative. For example, plots of the shape parameter ν, the variance, the ratio η and the mean-absolute-value each against the lossy bit rate R are shown in Figures 6(a), 6(b), 7(a) and 7(b), respectively in the case of the Lena image with different δ's. Evidently, the 'quantized' ($\delta > 0$) residuals also converge in probability distribution at different $\hat{R}_c$'s depending on δ's since the variances are almost the same in both domains and the shape parameters converge as the lossy bit rate R increases just like the case with $\delta = 0$. However, the reason for such a convergence is not obvious. One may argue that the 'quantized' residual distributions will also converge since the underlying 'unquantized' distributions already did. However, this is usually not the case since the quantized ($\delta > 0$) residual distributions in the two domains usually converge in probability distribution *before* the unquantized ($\delta = 0$) counterparts do. In other words, $\hat{R}_c \leq R_c$ in general, as can be readily checked from Figures 6(a) and 7(a).

To illustrate this point more clearly, Figures 8, 9 and 10 show pmf's (probability mass functions) of the quantized residuals of Lena for the cases of δ =0, 2 and 7, respectively. We can confirm faster convergence of the residual distributions with a larger δ.

Next, we present a theorem to explain why the 'quantized' versions of the encoding residuals in both domains converge in probability distribution, with a faster speed than their 'unquantized' counterparts.

Definition 4.1 (Refinement/processing of a distribution): Given a probability distribution P_X on $\boldsymbol{X}$, divide $\boldsymbol{X}$ into k mutually disjoint sets $\hat{\boldsymbol{X}}_1, \hat{\boldsymbol{X}}_2, ..., \hat{\boldsymbol{X}}_k$ satisfying

$$\boldsymbol{X} = \bigcup_{i=1}^{k} \hat{\boldsymbol{X}}_i.$$

Define a new distribution $P_{\hat{X}}$ as

$$P_{\hat{X}}(\hat{x}_i)=\sum_{x\in\hat{\boldsymbol{X}}_i} P_X(x),$$

where $\hat{x}_i \equiv f(x)$, $x \in \hat{\boldsymbol{X}}_i$, and f is a many-to-one mapping such as a uniform quantizer defined in (9). Then P_X is called a *refinement* of $P_{\hat{X}}$ and $P_{\hat{X}}$ is called a *processing* of P_X.

Now, we need a lemma regarding the divergence $D(P_X||P_Y)$ of the distributions P_X and P_Y of two random variables X and Y, respectively.

Lemma 4.2: Let $P_{\hat{X}}$ and $P_{\hat{Y}}$ be the processing of P_X and P_Y, respectively. Then

$$D(P_X||P_Y) \geq D(P_{\hat{X}}||P_{\hat{Y}})$$

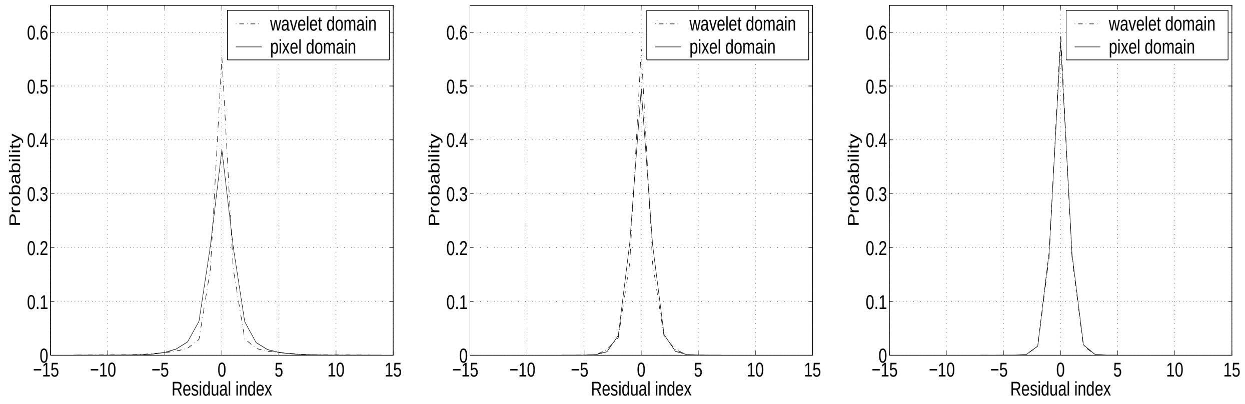
Proof: See the Appendix (also see, for example, p.300 of [23]). ■

Theorem 4.3: Let $\hat{Y}$ and $\hat{X}$ be the quantization indices obtained from uniformly quantizing the random variables Y and X, respectively (Y: wavelet-domain error residual, X: pixel-domain error residual). Then $P_{\hat{Y}}$ converges to $P_{\hat{X}}$ at least as fast as P_Y does to P_X.

Proof: Observe that quantization partitions the event space of the original random variable. Thus the distribution of the quantization indices is nothing but a *processing* of the original distribution. Since $D(P_Y||P_X) \geq 0$ with equality iff $P_Y(y) = P_X(x)$ and we have $P_Y(y) \rightarrow P_X(x)$, it follows from lemma 4.2 that $P_{\hat{Y}}(\hat{y}) \rightarrow P_{\hat{X}}(\hat{x})$ at least as fast as its unquantized counterpart. ■

V. On the fly procedure for optimal lossy layer bit rate

In this section, based on the observations and the analyses in previous sections, we present an 'on-the-fly' algorithm to locate the optimal lossy layer bit rate. Recall from Sections III-B and IV-C that the first-order entropy of the quantized pixel-domain encoding residual ($H_1(\hat{X})$) is the same as that of the quantized wavelet-domain counterpart ($H_1(\hat{Y})$) when the first-stage encoding rate R is above a certain rate $\hat{R}_c$. (Assume hereafter the term 'quantized' includes the case with $\delta = 0$ as a special case.) Thus we only need to know $H_1(\hat{Y}) + R$ in order to estimate the actual total bit rate $H_1(\hat{X}) + R$ when R is above $\hat{R}_c$. Since we have the wavelet domain residual ($\hat{Y}$)

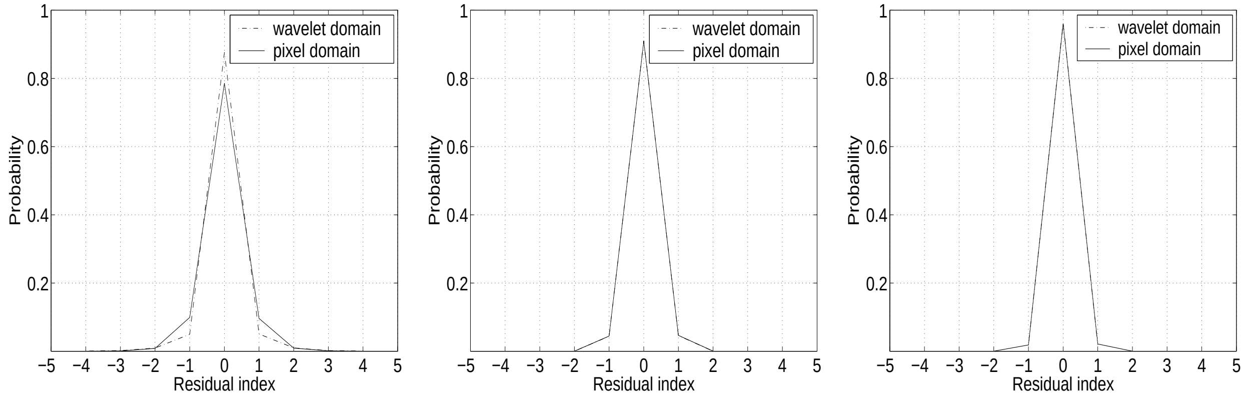


(a) δ=2 , lossy layer rate=0.11

(b) δ=2 , lossy layer rate=0.31

(c) δ=2 , lossy layer rate=0.51

Fig. 9. Convergence of residual distributions for Lena (δ=2)



(a) δ=7 , lossy layer rate=0.11

(b) δ=7 , lossy layer rate=0.31

(c) δ=7 , lossy layer rate=0.51

Fig. 10. Convergence of residual distributions for Lena (δ=7)

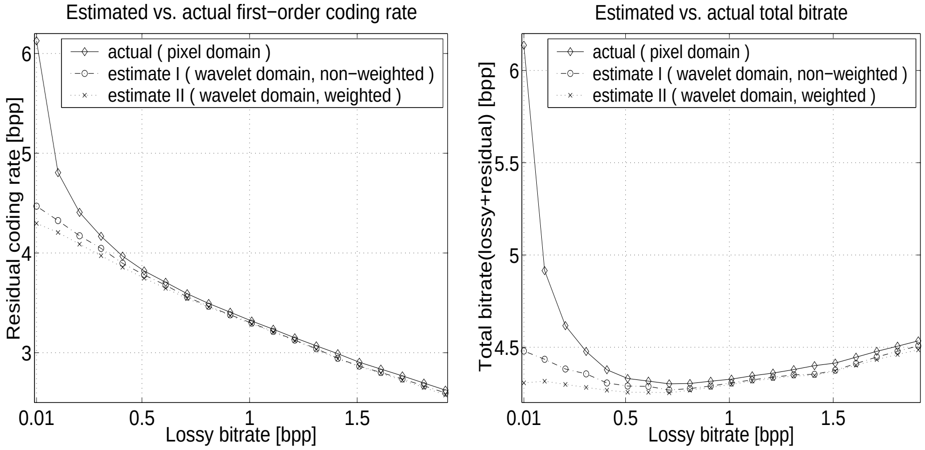
in the memory of the first-stage encoder while progressively encoding the source, we can calculate its first-order entropy ($H_1(\hat{Y})$) 'on the fly'. This implies we do not need to generate lossy reconstructions at various bit rates in order to get the quantized pixel-domain residuals ($\hat{X}$) at corresponding bit rates. Thus, potential savings in computation obtained thereby can turn out to be significant and it is indeed one of the most important advantages of our scheme. In other wavelet-based two-stage schemes mentioned in previous sections, one first needs to encode a source at a sufficiently high bit rate, and then decode the compressed bitstream at various bit rates to generate the lossy layers and their corresponding residual layers. This means one needs to take the inverse wavelet transforms to generate the lossy reconstructions at various bit rates and also actually perform the (context-based) entropy coding of the corresponding residual possibly with trial and error. Obviously such a process will be quite time-consuming especially for 3D data sets such as medical or motion video sequences.

Algorithm 1 describes an 'on-the-fly' estimation procedure of the first-order entropy of the quantized wavelet domain residual $(H_1(\hat{Y}))$ as we successively encode a source with SPIHT. The idea of the algorithm is that only two symbol probabilities (P_i's) of the quantized wavelet-domain residual need to be updated at any particular moment of successive bit-plane encoding in order to keep track of the corresponding change in the first-order entropy ($H_1(\hat{Y})$), where P_i is given as the frequency count (FR_i) of the corresponding quantized wavelet-domain residual value divided by the total number (N) of the wavelet coefficients. Thus, it is readily understood that the proposed estimation algorithm requires almost negligible computational effort.

A few comments are in order regarding the implementation of the algorithm. First, note that the uniform quantization rule given by (9) (i.e., the near-lossless quantization formula) does not work if e is non-integer. For example, if e =1.3 and δ =1, we get $\hat{e}_{id}$ =0 and $\hat{e}$ =0. But this leads to $\|e-\hat{e}\| = 1.3 > \delta = 1$. In fact, when estimating the first-order entropy $H_1(\hat{Y})$ in Algorithm 1, we found it necessary to round the floating point value of the wavelet domain residual to the nearest integer before applying the uniform quantization rule given by (9). Second, when it comes to evaluating the first-

Algorithm 1 Iterative First-Order Residual-Entropy Estimation ‘on the fly’

```
Step 1: Given a maximum-error bound $\delta$, initialize the first-order entropy of the quantized wavelet coefficients as follows:
    $H_1(\hat{Y}) = -\sum_i P_i \log P_i$
, where $P_i$ is the probability of the $i$-th symbol for the quantized wavelet coefficients.
Step 2: Everytime a wavelet coefficient is further encoded via updating the significance/refinement information inside the SPIHT, save the symbol index of the quantized value of that wavelet coefficient before updating as $old$ and find the corresponding one after updating as $new$.
Step 3: Subtract the contribution of the above symbols from the first-order entropy estimate as follows:
    $H_1(\hat{Y}) \Leftarrow H_1(\hat{Y}) + P_{old} \log P_{old} + P_{new} \log P_{new}$
Step 4: Update the symbol probabilities of the symbols $old$ and $new$ as follows:
    $FR_{old} = FR_{old} - 1$ , $FR_{new} = FR_{new} + 1$
    $P_{old} = FR_{old}/N$ , $P_{new} = FR_{new}/N$
,where $FR_i$ is the frequency count of the $i$-th symbol and $N$ is the total number of the wavelet coefficients.
Step 5: Add the new contribution of the updated symbol probabilities to the first-order entropy estimate as follows:
    $H_1(\hat{Y}) \Leftarrow H_1(\hat{Y}) - P_{old} \log P_{old} - P_{new} \log P_{new}$
Step 6: Repeat steps 2 through 5 everytime a wavelet coefficient is further encoded.
```



(a) Residual layer rate when $\delta = 0$ and $R_c \approx 0.61$ bpp

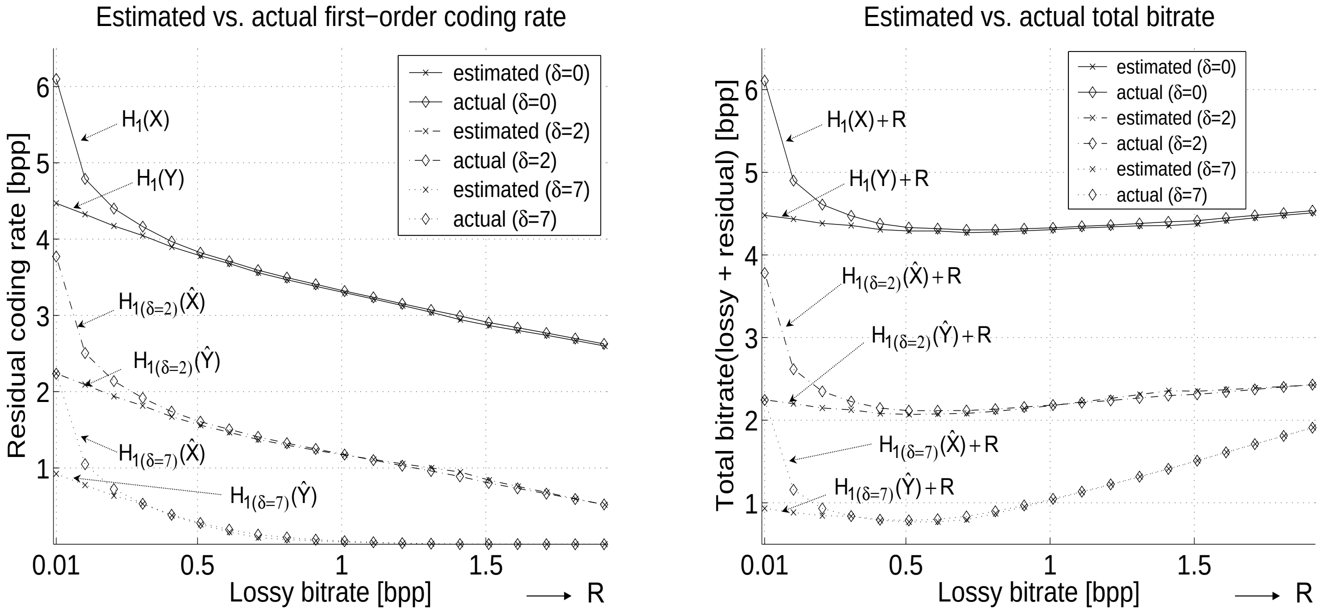
(b) Total bit rate when $\delta = 0$ and $R_c \approx 0.61$ bpp

Fig. 11. Comparison of first-order entropy calculation methods for Lena

order entropy of a wavelet transformed image, one may think of a weighted sum of the first-order entropy of each subband defined as $\sum_{j=1}^{N} \frac{1}{M_j} H_{1,j}(Y)$, where $H_{1,j}$ is the first-order entropy of the j-th subband, N is the total number of subbands and $M_j = 4^l$ is the decimation factor when the j-th subband belongs to the l-th decomposition level in a typical dyadic wavelet transform. This means we need to keep N separate pmf’s (probability mass functions) — equivalently, sets of symbol counts — in Algorithm 1, given N subbands. However, we chose to use only a single pmf for the entire subbands. This way we can not only save memory and further reduce the already trivial computational burden but also better detect the optimal lossy bit rate as will be illustrated in the following. Figure 11 compares two methods of calculating the first-order entropy of the wavelet domain residual for Lena. ‘Estimate I’ in the figure corresponds to the method we chose (i.e. use of only one pmf), while ‘Estimate II’ corresponds to the weighted first-order entropy described above. Note the first-order entropy of ‘Estimate I’ is larger than that of ‘Estimate II’ when the first-stage bit rate (R) is below the ‘critical’ rate R_c. In fact, this is true regardless of the image and can be easily understood by the following inequality:

$$\begin{aligned}\sum_{j=1}^{N} \frac{1}{M_j} H_{1,j}(Y) &= \sum_{j=1}^{N} \frac{1}{M_j} H_1(Y|Y \in j\text{-th subband})\\ &\leq \sum_{j=1}^{N} \frac{1}{M_j} H_1(Y)\\ &= H_1(Y) \cdot \sum_{j=1}^{N} \frac{1}{M_j} = H_1(Y) \qquad (13)\end{aligned}$$

where the inequality comes from the fact conditioning reduces entropy. Considering that the first-order entropy of the pixel-domain residual is (significantly) larger than that of the



(a) Optimum lossy rate: 0.61 bpp ($\delta = 0$), 0.61 bpp ($\delta = 2$), 0.61 bpp ($\delta = 7$)

(b) Optimum lossy rate : 0.71 bpp ($\delta = 0$), 0.61 bpp ($\delta = 2$), 0.51 bpp ($\delta = 7$)

Fig. 12. Bit rates for Lena, X: pixel-domain residual, Y:wavelet-domain residual

wavelet-domain counterpart when R is below R_c, use of a single pmf for the wavelet-domain residual always helps better detect the optimal lossy bit rate as it makes the curve more convex by slightly 'lifting up' the total bit rate estimation curve toward the actual one as shown in Figure 11(b). Also notice that as the residual becomes i.i.d., the two methods of calculating the first-order entropy yield the same value since conditioning does not reduce the entropy of an i.i.d. distribution.

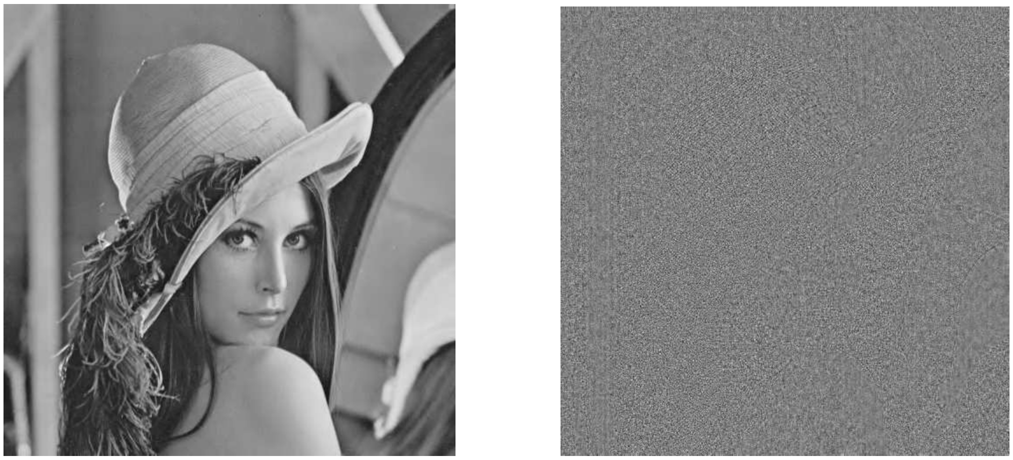
VI. Experimental Results and Discussion

This section shows the results of experiments to demonstrate the validity of our approach. First, the effectiveness of our 'on-the-fly' algorithm to locate the optimal lossy rate is shown and some issues are discussed. Next, we compare its near-lossless/lossless compression performance with other schemes in the literature. To facilitate comparison with the results of other coders available in the literature, we chose to use two representative natural images (Lena and Barbara). However, keep in mind that the observations and conclusions drawn here apply to most real sources. We used SPIHT as the first-stage coder and an adaptive arithmetic encoder [30] as the residual coder. Here 'adaptive' means the symbol probabilities are adjusted on the fly when encoding the residual. We used only one probability model within the adaptive arithmetic coder for all the pixels in a given residual layer and the typical encoding rate was very close to its first-order entropy.

A. Determination of Optimal Lossy Layer Bit Rate

In Figure 12(a), we have plotted the residual coding rates ($H_1(\cdot)$) versus the lossy bit rates (R) for the lossless ($\delta = 0$) and the near-lossless ($\delta = 2, \delta = 7$) cases. Figure 12(b) shows its corresponding total bit rate ($H_1(\cdot) + R$) versus the lossy bit rate (R) curves. To be more precise, $H_1(\hat{X})$ is the actual coding rate of the quantized pixel-domain residual by the arithmetic coder which always was virtually the same as its first-order entropy, while $H_1(\hat{Y})$ is the first-order entropy of the quantized wavelet-domain residual calculated 'on the fly' according to Algorithm 1. We can see the convergences of the residual (Figure 12(a)) and total (Figure 12(b)) coding rates in both domains above the rate R_c (lossless) or $\hat{R}_c$ (near-lossless).

Recall from Section III-B that it is more efficient to keep successively encoding the source by SPIHT than switching to a first-order entropy coder for the pixel domain residual while it has memory (i.e. not i.i.d.). However, beyond R_c, the encoding residual has almost no structure as exemplified in Figures 13 and 14, which show the lossy reconstructions and the corresponding residuals of Lena and Barbara at the optimal lossy bit rates, respectively. Therefore, attempting to further refine the source beyond rate R_c with SPIHT, which is not efficient for encoding small, independent random variables, leads to at least as large as and usually larger (thus, sub-optimal) total bit rates than those obtained by switching to a first-order residual coder. This explains why there exists a lossy bit rate or a small flat region of lossy bit rates around R_c where a minimum total bit rate (lossy plus residual bit rates) is achieved by the proposed two-stage scheme. Indeed we could confirm this with Lena for the case of $\delta = 0$ from Figure 12(b). The same observations basically apply to the case when $\delta > 0$. However, there is a subtle difference in this case. Although the quantized residuals $\hat{X}$ and $\hat{Y}$ converge in



(a) PSNR = 38.80 dB at 0.71 bpp

(b) Lossless residual (δ = 0)

Fig. 13. Lossy-layer reconstruction and the corresponding residual of Lena at optimal rate

probability distribution beyond the critical lossy bit rate $\hat{R}_c$, it is not obvious whether they necessarily become independent thereafter, since our argument regarding the convergence of $\hat{X}$ and $\hat{Y}$ in Section IV-C did not rely upon the independence of the quantized ($\delta > 0$) residual. In fact, by comparing Figure 12(a) with Figure 12(b) for the case of $\delta = 7$, we can see the critical rate $\hat{R}_c$ is 0.31 bpp, while the minimum total bit rate is achieved at the lossy bit rate R of 0.51 bpp. Since it would be still reasonable to expect the optimal lossy bit rate in the proposed method to correspond to the point where the pixel domain quantized residual $\hat{X}$ is independent, this result means that the critical rate $\hat{R}_c$ (, where the convergence of $\hat{X}$ and $\hat{Y}$ occurs) preceded the actual optimal lossy bit rate, when $\delta > 0$. This was a general tendency especially with a large δ, but poses no problem with detecting the optimal lossy bit rate anyway as we have the convergence of bit rate curves well before the optimal point is reached.

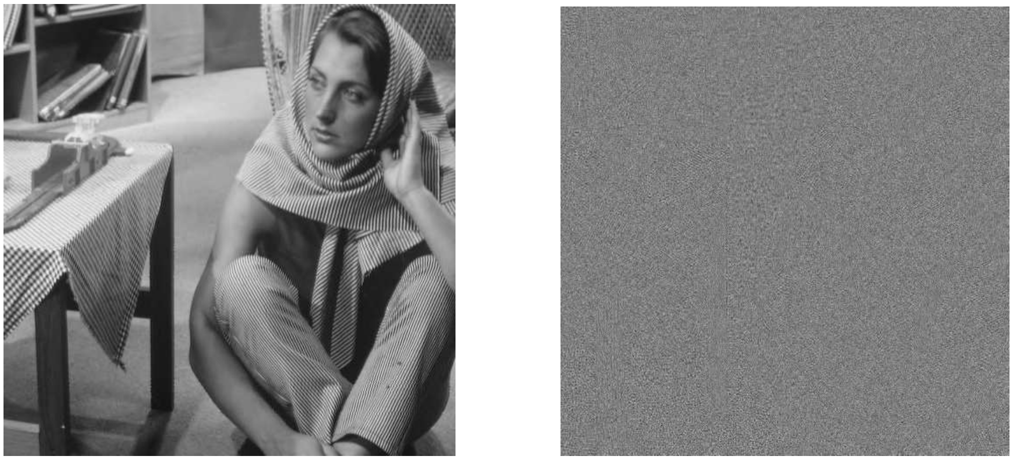
Next, Table I shows the accuracy of the method presented to locate the optimal lossy layer bit rate. In the table, the rows with ‘Actual rate by iteration’ correspond to the rates obtained by iteratively decoding the given image at every increment of 0.1 bpp starting from 0.01 bpp to find the optimal lossy bit rate (the 1st column for each image) and the actual minimum total (lossy plus residual) rate (the 2nd column for each image) obtained at the chosen optimal lossy rate. Also, the rows with ‘Estimated rate on the fly’ correspond to the estimated optimal lossy rates (estimation step size : 0.01 bpp) and the corresponding estimated total rates obtained using the proposed ‘on-the-fly’ algorithm. The total rates in the parentheses of these rows refer to the *actual* total rates obtained by stopping the SPIHT encoding at the *estimated* optimal lossy rate and encoding the quantized residual with an arithmetic coder. Note no iteration is needed in this case and the actual total bit rates thus obtained (i.e. the numbers in the parentheses) are very close to the actual minimum total bit rates obtained through iteration.

Image name		Lena		Barbara	
Tolr.	Method	bpp	PSNR	bpp	PSNR
$\delta = 0$	JPEG-LS	4.25	∞	4.86	∞
	CALIC	**4.10**	∞	**4.59**	∞
	Prequant. S+P [7]	4.20	∞	4.71	∞
	Iter. SPIHT + context-AC [7]	N/A	∞	N/A	∞
	SPIHT + AC (proposed)	4.30	∞	4.90	∞
$\delta = 1$	JPEG-LS	2.72	49.90	3.30	49.89
	CALIC	**2.59**	49.89	**3.07**	49.89
	Prequant. S+P	2.78	49.90	3.27	49.89
	Iter. SPIHT + context-AC	2.69	49.89	3.31	49.89
	SPIHT + AC	2.77	49.89	3.38	49.90
$\delta = 2$	JPEG-LS	2.09	45.15	2.65	45.14
	CALIC	**1.95**	45.16	**2.42**	45.14
	Prequant. S+P	2.25	45.11	2.72	45.12
	Iter. SPIHT + context-AC	2.02	45.16	2.65	45.17
	SPIHT + AC	2.12	45.17	2.72	45.16
$\delta = 4$	JPEG-LS	1.54	40.11	2.02	40.04
	CALIC	1.29	40.27	**1.77**	40.11
	Prequant. S+P	1.74	39.89	2.17	39.89
	Iter. SPIHT + Context-AC	**1.28**	40.59	1.91	40.54
	SPIHT + AC	1.36	40.65	1.97	40.52
$\delta = 6$	JPEG-LS	1.24	37.17	1.67	36.99
	CALIC	0.96	37.57	**1.40**	37.21
	Prequant. S+P	1.48	36.60	1.86	36.68
	Iter. SPIHT + Context-AC	**0.86**	38.54	1.48	38.59
	SPIHT + AC	0.92	38.76	1.52	38.56
$\delta = 7$	JPEG-LS	1.14	35.99	1.54	35.82
	CALIC	0.85	36.56	**1.28**	36.10
	Prequant. S+P	1.40	35.34	1.76	35.43
	Iter. SPIHT + Context-AC	**0.73**	37.95	1.35	38.10
	SPIHT + AC	0.79	38.27	1.37	38.07

TABLE II
COMPARISON WITH OTHER CODERS FOR LENA & BARBARA

B. Near-Lossless Compression Performance

Table II compares the performance of the proposed coder with JPEG-LS [13], CALIC [11], prequant. S+P [7] (prequantization followed by lossless compression of quantization indices as explained in Section II-B) and ‘Iter. SPIHT + Context-AC’ [7] (a wavelet-based two-stage near-lossless coder that needs iteration to find the optimal first-stage bit rate and uses a context-based entropy coding



(a) PSNR = 40.62 dB at 1.61 bpp

(b) Lossless residual (δ = 0)

Fig. 14. Lossy-layer reconstruction and the corresponding residual of Barbara at optimal rate

Test conditions		Lena		Barbara	
Tolerance	Method	Optimal lossy	Total at optimal	Optimal lossy	Total at optimal
$\delta = 0$	Actual rate by iteration	0.71	4.30	1.61	4.90
	Estimated rate on the fly	0.71	4.27 (act:4.30)	1.51	4.87 (act:4.90)
$\delta = 1$	Actual rate by iteration	0.71	2.77	1.31	3.38
	Estimated rate on the fly	0.71	2.75 (act:2.77)	1.29	3.35 (act:3.38)
$\delta = 2$	Actual rate by iteration	0.61	2.12	1.11	2.72
	Estimated rate on the fly	0.61	2.07 (act:2.12)	1.11	2.67 (act:2.72)
$\delta = 4$	Actual rate by iteration	0.51	1.36	1.11	1.97
	Estimated rate on the fly	0.43	1.33 (act:1.37)	0.95	1.97 (act:1.98)
$\delta = 6$	Actual rate by iteration	0.51	0.92	1.11	1.52
	Estimated rate on the fly	0.43	0.91 (act:0.94)	0.89	1.51 (act:1.55)
$\delta = 7$	Actual rate by iteration	0.51	0.79	1.11	1.37
	Estimated rate on the fly	0.51	0.77 (act:0.79)	1.15	1.33 (act:1.38)

TABLE I

ACCURACY OF OPTIMAL RATE ESTIMATION FOR LENA & BARBARA, ALL NUMBERS ARE IN [BPP]

for residuals) for Lena and Barbara images. In terms of both PSNR and the total bit rate, the wavelet-based two-stage coders including our proposed one were always comparable (with small δ's) or better (with large δ's) compared with JPEG-LS or the prequant. S+P scheme. It is encouraging to see that the two-stage wavelet coders were quite comparable to JPEG-LS even at the high bit rates (i.e. small δ's), since they provide (partially) L^2 error scalable bitstream unlike JPEG-LS. Similar observations apply when comparing against CALIC, although CALIC outperforms all the other coders in the table in terms of the bit rate for the Barbara image regardless of δ. Note the extra savings in total bit rates achieved by the 'Iter. SPIHT + Context-AC' come at the expense of high-complexity context-based entropy coding of the residual, which requires exhaustive iteration to locate the optimal lossy bit rate. Alternatively, one can use our algorithm to locate a starting lossy layer bit rate in order to reduce the number of iterations needed when 'Iter. SPIHT + Context-AC' is adopted to achieve better compression.

Lastly, observe that it is reasonable to expect the i.i.d. residual to be obtained at a lower lossy bit rate if the lossy layer coder is improved. In other words, by improving the lossy coder's R-D performance, we would be able to achieve the same 'critical distortion' at a lower rate, thereby reducing the total bit rate. As an illustration, Table III shows a result of incorporating JPEG-2000 style context models [17] to improve the sign/refinement coding within SPIHT (see the rows 'SPIHT (with sign/rfn) + AC'). Especially with Barbara image, we could see noticeable improvement in total bit rates. This demonstrates a possible way of further improving the performance of the proposed method, which is quite flexible in the sense that incorporating any improvement into the lossy layer codec (e.g. better transform, novel context-modelling for entropy coding, motion-compensation for motion video etc.) to achieve better performance in the lossy layer usually translates into improved near-lossless/lossless performance.

VII. CONCLUSION

In this paper, we presented a practical method of choosing the lossy layer bit rate so that the sum of lossy and residual coding rate is minimized in a wavelet-based two-stage near-lossless coding scheme. We proposed to use a simple arithmetic coding of the quantized residual to meet an L^∞

Image name		Lena		Barbara	
Tolerance	Method	bpp	PSNR	bpp	PSNR
$\delta = 0$	JPEG-LS	**4.25**	∞	4.86	∞
	SPIHT + AC	4.30	∞	4.90	∞
	SPIHT (with sign/rfn) + AC	4.29	∞	**4.84**	∞
$\delta = 2$	JPEG-LS	**2.09**	45.15	**2.65**	45.14
	SPIHT + AC	2.12	45.17	2.72	45.16
	SPIHT (with sign/rfn)+ AC	2.10	45.18	2.67	45.17

TABLE III

USE OF CONTEXT-MODELS FOR IMPROVED SIGN & REFINEMENT CODING WITHIN SPIHT

error requirement, following the lossy reconstruction layer consisting of a wavelet-based successive bit-plane encoder, such as SPIHT, SPECK [16], or JPEG-2000. The method capitalizes upon the notion of ‘critical rate’ well-known from rate-distortion theory as well as the ‘entropy reduction’ property of subband decomposition, and works reasonably well for most real sources. The most notable advantage of the proposed scheme is its simplicity compared with its predecessors in the literature. We introduced a very simple algorithm that allows one to locate the optimum lossy reconstruction bit rate on-the-fly without any time-consuming iteration of decoding, IDWT, and actual residual encoding for a given L^∞ error bound. Its performance in terms of bit rates in near-lossless and lossless coding was comparable to that of other state-of-the-art near-lossless coders proposed in the literature.

APPENDIX

Proof of Lemma 4.2: By log-sum inequality,

$$\begin{aligned}
& \sum_{x \in \hat{\boldsymbol{X}}_i} P_X(x) \log \frac{P_X(x)}{P_Y(x)} \\
\geq \; & (\sum_{x \in \hat{\boldsymbol{X}}_i} P_X(x)) \log \frac{\sum_{x \in \hat{\boldsymbol{X}}_i} P_X(x)}{\sum_{x \in \hat{\boldsymbol{X}}_i} P_Y(x)} \\
= \; & P_{\hat{X}}(\hat{x}_i) \log \frac{P_{\hat{X}}(\hat{x}_i)}{P_{\hat{Y}}(\hat{x}_i)} \qquad (14)
\end{aligned}$$

$$\begin{aligned}
D(P_X||P_Y) &= \sum_{i=1}^{k} \sum_{x \in \hat{\boldsymbol{X}}_i} P_X(x) \log \frac{P_X(x)}{P_Y(x)} \\
&\geq \sum_{i=1}^{k} P_{\hat{X}}(\hat{x}_i) \log \frac{P_{\hat{X}}(\hat{x}_i)}{P_{\hat{Y}}(\hat{x}_i)} \\
&= D(P_{\hat{X}}||P_{\hat{Y}})
\end{aligned}$$

■

REFERENCES

[1] A. Said and W. Pearlman, “New, fast and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243-250, Jun. 1996.

[2] A. Said and W. Pearlman, “An image multiresolution representation for lossless and lossy compression,” *IEEE Trans. Image Processing*, vol. 5, pp. 1303-1310, Sept. 1996.

[3] J.M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Trans. Signal Processing*, vol. 41, pp. 3445-3462, Dec. 1993.

[4] L. Ke and M. Marcellin, “Near-lossles image compression: minimum-entropy, constrained-error DPCM,” *IEEE Trans. Image Processing*, vol. 7, pp. 225-228, Feb. 1998.

[5] K. Chen and T. Ramabadran, “Near-lossless compression of medical images through entropy-coded DPCM,” *IEEE Trans. Med. Imag.*, vol. 13, pp. 538-548, Sept. 1994.

[6] P. Cosman, R. Gray, and R. Olshen, “Evaluating quality of compressed medical images: SNR, subjective rating, and diagnostic accuracy,” *Proc. IEEE*, vol. 82, pp. 919-932, Jun. 1994.

[7] R. Ansari and N.Memon, and E. Ceran, “Near-lossless image compression techniques,” *J. Electronic Imaging*, vol. 7, pp. 486-494, Jul. 1998.

[8] D. Marpe, G. Blattermann, J. Ricke, and P. Maab, “A two-layered wavelet-based algorithm for efficient lossless and lossy image compression,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 1094-1102, Oct. 2000.

[9] A. Krivoulets, “A method for progressive near-lossless image compression,” *Proc. of IEEE Int. Conf. Image Processing*, vol. 3, pp. 185-188, 2003.

[10] L. Karray, P. Duhamel, and O. Rioul, “Image coding with an l^∞ norm and confidence interval criteria,” *IEEE Trans. Image Processing*, vol. 7, pp. 621-631, May 1998.

[11] X. Wu and P. Bao, “L^∞ constrained high-fidelity image compression via adaptive context modeling,” *IEEE Trans. Image Processing*,vol. 9, pp. 536-542, Apr. 2000.

[12] A. Alecu, A. Munteanu, P. Schelkens, J. Cornelis, and S. Dewitte, “Wavelet-based fixed and embedded L-infinite-constrained image coding,” *J. of Electronic Imaging*, vol. 12, pp. 522-538, Jul. 2003.

[13] M. Weinberger, G. Seroussi, and G. Sapiro,“The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS,” *IEEE Trans. Image Processing*, vol. 9, pp. 1309-1324, Aug. 2000.

[14] D. S. Taubman, “High performance scalable image compression with EBCOT,” *IEEE Trans. Image Processing*, vol. 9, pp. 1158-1170, Jul. 2000.

[15] S. Hsiang and J. W. Woods, “Embedded image coding using zeroblocks of subband/wavelet coefficients and context modeling,” in *Proc. 2000 IEEE Int. Symp. Circuits and Systems*, vol. 3, pp. 662-665, May 2000.

[16] W. A. Pearlman, A. Islam, N. Nagaraj and A. Said, “Efficient, low-complexity image coding with a set-partitioning embedded block coder,” *IEEE Trans. Circuits Syst. Video Technol.*, vol.14, pp. 1219-1235, Nov. 2004.

[17] D. S. Taubman and M. W. Marcellin, *JPEG2000 : Image Compression Fundamentals, Standards and Practice*. Norwell, MA, Kluwer,2002.

[18] B. E. Usevitch, “A tutorial on modern lossy wavelet image compression : foundations of JPEG 2000,” *IEEE Signal Processing Mag.*, vol. 18, pp. 22-35, Sept. 2001.

[19] J. Reichel, G. Menegaz, M. J. Nadenau, and M. Kunt, “Integer wavelet transform for embedded lossy to lossless image compression,” *IEEE Trans. Image Processing*, vol. 10, pp. 383-392, Mar. 2001.

[20] M. D. Adams and F. Kossentini, “Reversible integer-to-integer wavelet transforms for image compression : performance evaluation and analysis,” *IEEE Trans. Image Processing*, vol. 9, pp. 1010-1024, Jun. 2000.

[21] Z. Xiong, X. Wu, S. Cheng and J. Hua, “Lossy-to-lossless compression of medical volumetric data using three-dimensional integer wavelet transforms,” *IEEE Trans. on Medical Imaging*, Vol. 22, pp. 459-470, March 2003.

[22] T. Berger, *Rate Distortion Theory*. Prentice-Hall, 1971.

[23] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, N.Y., 1991.

[24] J. Pan and T. R. Fischer, “Two-stage vector quantization - lattice vector quantization,” *IEEE Trans. Inform. Theory*, vol. 41, pp. 155-163, Jan. 1995.

[25] T. Linder and R. Zamir, “On the asymptotic tightness of the Shannon lower bound,” *IEEE Trans. Inform. Theory*, vol. 40, pp. 2026-2031, Nov. 1994.

[26] R. Zamir, “Gaussian codes and Shannon bounds for multiple descriptions,” *IEEE Trans. Inform. Theory*, vol. 45, pp. 2629-2636, Nov. 1999.

[27] R. Zamir and T. Berger, “Multiterminal source coding with high-resolution,” *IEEE Trans. Inform. Theory*, vol. 45, pp. 106-117, Jan. 1999.

[28] P. Rao and W. A. Pearlman, “On entropy of pyramid structures,” *IEEE Trans. Inform. Theory*, vol. 37, pp. 407-413, Mar. 1991.

[29] K. Sharifi and A. Leon-Garcia, “Estimation of Shape Parameter for Generalized Gaussian Distributions in Subband Decompositions of Video,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 52 - 56, Feb. 1995.

[30] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression,” *Communication of ACM*, vol. 30, pp.520-540, Jun. 1987.