

PAPER

Lossless-by-Lossy Coding for Scalable Lossless Image Compression

Kazuma SHINODA^{†*}, *Member*, Hisakazu KIKUCHI^{†a)}, *Fellow*, and Shogo MURAMATSU[†], *Member*

SUMMARY This paper presents a method of scalable lossless image compression by means of lossy coding. A progressive decoding capability and a full decoding for the lossless rendition are equipped with the losslessly encoded bit stream. Embedded coding is applied to large-amplitude coefficients in a wavelet transform domain. The other wavelet coefficients are encoded by a context-based entropy coding. The proposed method slightly outperforms JPEG-LS in lossless compression. Its rate-distortion performance with respect to progressive decoding is close to that of JPEG2000. The spatial scalability with respect to resolution is also available.

key words: *image compression, lossless compression, scalable coding, embedded coding, wavelet transform, context modeling*

1. Introduction

As is well known, most lossless image coders operate in the picture domain to take full advantage of spatial correlation among pixel values. A sophisticated predictive scheme is developed on the neighborhood of a target pixel, and the prediction error is encoded by entropy coding. JPEG-LS [1] is a representative of predictive lossless image compression, which is based on the LOCO-I [2] algorithm. JPEG-LS uses the median edge detector and a simplified Golomb coding and offers an excellent compromise between high compression performance and low computational complexity. Some of the recent lossless coding are based on adaptive prediction, and have higher compression rates than those of JPEG-LS. TMW [3] incorporates a multiple-pass modeling. MRP [4] is based on a devise of a sophisticated predictor that minimizes the entropy of the prediction residual. Their compression rates are quite high at the expense of a lot of computational complexity. Unfortunately, their output bit streams are not scalable by any means.

In contrast, some of lossy image coders are developed in the wavelet transform domain since the advent of EZW [5]. SPIHT [6] and SPECK [7] have efficient embedded coding algorithms which describe significant coefficients by using the inter-band correlation or intra-band correlation in the wavelet transform domain. In S+P [8], lossy compression is implemented with the aid of SPIHT, while lossless compression is differently performed by a context-based en-

trophy coding. EBCOT [9], [10] defines a small piece of coding unit by the bitplane stripe in code-blocks so that a variety of data-partitioning and an optimal truncation of insignificant code units may be developed even after compression. They are very effective with respect to compression rates. In addition, embedded coding offers partial decoding and incremental decoding. The encoded bit stream is organized as a collection of separate information entities that are sorted in decreasing order of importance. Decoding of such a bit stream can be stopped at any point in the bit stream with the maximum contribution to a particular importance measure such as signal-to-noise ratio (SNR). Embedded coding is hence possible to achieve lossless coding at the extreme condition. However, it is less effective than predictive coding such as JPEG-LS in lossless compression.

On the other hand, the application areas of lossless image coding are mostly biased to office-use and professional use rather than public use. Editing of broadcast/publishing contents, archiving of valuable observations and experiments, and medical imaging as well as their transmission are some examples. Also, a strong demand has emerged in telepathology [11], [12]. It is a virtual slide that is the microscopic digital image of a whole slide of a biopsy lesion. For instance, if a 15 × 15-mm square sample is imaged by an objective lens of 40 times magnification factor onto an imaging device of 0.39 microns/pixel pitch [13], it generates a 1.5 giga pixel-digital image having 38k × 38k resolution, which amounts to 4.5 GB in raw RGB format. In such an application, the spatial scalability is indispensable as well as high lossless compression performance and fast operation. A combination of lossless coding and the spatial scalability is requested as well as progressive decoding capability to help quick browsing, searching and observation within a tolerable time and space.

Unfortunately JPEG-LS does not offer any particular support for scalability. While JPEG2000 [9], [10] offers excellent scalability, there seems to be room for a possible negotiation between the computational complexity and the coding performance in the lossless mode.

This paper presents a scalable lossless image coding that offers the spatial and SNR scalability and progressive decoding. The coding method is developed based on lossy coding of wavelet coefficients to get scalability and high performance in progressive decoding. The residual information for lossless coding is compressed by a context-based entropy coding. The proposed method consists of three steps. A reversible wavelet transform is the first step. In the second

Manuscript received December 19, 2007.

Manuscript revised May 12, 2008.

[†]The authors are with the Department of Electrical and Electronic Engineering, Niigata University, Niigata-shi, 950-2181 Japan.

*Presently, with Display Business Division, Sony Corporation.

a) E-mail: kikuchi@eng.niigata-u.ac.jp

DOI: 10.1093/ietfec/e91-a.11.3356

step, large wavelet coefficients in magnitude are encoded by an embedded coding. In the third step, the other small coefficients are compressed by a context-based arithmetic coding.

The rest of the paper is organized as follows. Section 2 describes the proposed algorithm in detail. Comparisons with JPEG-LS and JPEG2000 are presented in Sect. 3. Conclusions follow in Sect. 4.

2. Lossless-by-Lossy Coding

The idea of *lossless-by-lossy* is not always new. The authors knew it, when they wrote this paper. In the early literature [14], their lossless image coding is made of the JPEG-compressed lossy data and the residual data, and the residual information is compressed by a 3D adaptive prediction and an adaptive arithmetic coding. Another example is the *lossy plus lossless* scheme presented in [15].

While lossless compression is the common objective for all of them, this paper adds another: progressive lossy-to-lossless decoding. It implies that the decoding bit rate is variable by ceasing at any position of a single bit stream, and that the original image is losslessly decodable from the same bit stream. As the number of decoded bits increases, the pixel accuracy is progressively improved at a decoder. The decoded image resolution is also selectable among the full resolution and several subband sizes.

2.1 Outline of the Coding Method

As a result of multi-level wavelet transforms along horizontal and vertical directions, a subband decomposition of the wavelet coefficients is generated. Among them, large coefficients in magnitude are encoded by a sort of embedded coding [16], [17], because their population is sparse and both their values and locations have to be encoded. Threshold values for a significance test are the power of two for ease of implementation. Once a significant coefficient has been found against a threshold, its approximation is incrementally improved as the threshold value decreases. Those bits produced by the large-coefficient coding are fed to arithmetic coding to form a primary bit stream.

In the proposed coding method, a crossover threshold denoted by T_c plays an important role to control two types of encoding for large and small coefficients. Throughout the paper, if the magnitude of a wavelet coefficient is larger than or equal to T_c , the coefficient is said to be large, unless otherwise. Once the primary encoding for large coefficients against a particular crossover threshold has been completed, the other small coefficients are left to be coded in a different way.

It is worth noting the following treatment in order to avoid a possible confusion. No matter how many lower bits may be left in the large coefficients, when the significance test has reached the crossover threshold, they are encoded before encoding the small coefficients. The degree of scalability is controlled by the crossover threshold value. The decoding bit rate in lossy decoding is arbitrarily selectable

so as to obtain a desirable image rendition with respect to bit rate, pixel accuracy (SNR) and resolution, as far as the bit-plane of the decoded bits is higher or equal to the crossover threshold.

2.2 Subband Shift in the Wavelet Transform

The 9/7-tap integer biorthogonal wavelet system [18]–[20] is applied to a given image. A pair of impulse responses of the lowpass filter and the highpass filter are of the sequences as follows. $\{1, 0, -8, 16, 46, 16, -8, 0, 1\}$ and $\{1, 0, -9, 16, -9, 0, 1\}$, which are divided by 64 and 16, respectively. The lifting computation makes the transform reversible and is shared between decomposition and synthesis for lossless decoding. In lifting, the input data is split into even and odd sequences denoted by $c(n)$ and $d(n)$, respectively. The prediction step in lifting is computed by

$$d(n) \leftarrow d(n) - \left\lfloor \frac{-c(n-1) + 9c(n) + 9c(n+1) - c(n+2)}{16} \right\rfloor \quad (1)$$

to define the detail component, where $\lfloor x \rfloor$ stands for the largest integer that does not exceed x . The update step in lifting is then given by

$$c(n) \leftarrow c(n) + \left\lfloor \frac{d(n-1) + d(n)}{4} + \frac{1}{2} \right\rfloor \quad (2)$$

to define the coarse component. The dc level shifting is implemented after the multi-level transform for saving computational burden. Only the coefficients in the coarsest LL subband are subtracted by their average so that all of the coefficients in every subband may be zero-mean samples.

As observed in the expansion of lifting expressions into a plain form in raw samples, the dc gain of the lowpass component is unity, while the highest-frequency gain of the highpass component is two. Although these imbalanced gain factors are irrelevant to lossless decoding, they can cause the lowpass component to be underestimated in the evaluation of significance against coefficient approximation so that coding efficiency is degraded in progressive decoding.

To compensate the discrepancy in these gain factors, the lowpass component is multiplied by $\sqrt{2}$ and the highpass component is divided by the same factor. Since multi-level transforms are applied horizontally and vertically, a two-dimensional subband decomposition is generated, resulting in energy compaction into different subbands. LL subband is thus shifted by one bit upward with respect to HL and LH subbands. The different gain factors cancel out at HL and LH subbands because of a combination of lowpass and highpass. HH subband has to be relatively shifted by one bit downward with respect to HL and LH. The relative scaling factors among four types of subbands are hence described by $\{LL, HL, LH, HH\} = \{2, 1, 1, 0.5\}$. The resulting subband weighting is illustrated in Fig. 1. Actually, the scaling factor of 0.5 is not implemented. Instead, the highest HH subband is identified with the basement level, and all

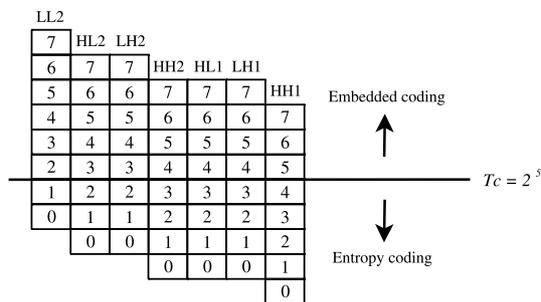


Fig. 1 Subband shift.

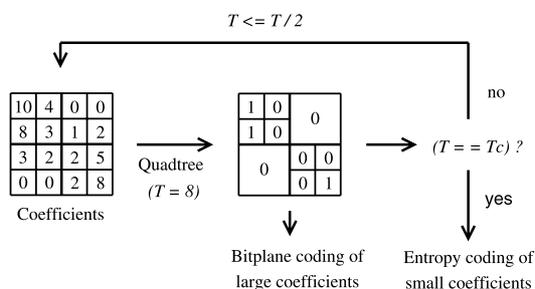


Fig. 2 Outline of the coding method.

the other subbands are relatively and equally up-shifted by one bit.

2.3 Coding of Large Coefficients

Large coefficients are encoded by a sort of embedded bitplane coding with quadtrees. At first, all of the subbands are assumed to belong to an insignificant block. The significance of every coefficient is tested against a prescribed threshold value denoted by T to find large coefficients. The threshold value T for the significance test is initially set to the bitplane indicated by the most significant bit of the maximum coefficient in magnitude. The location of a significant coefficient is identified by using a quadtree. The quadtree coding proceeds in such a way that, if a coefficient in a particular block has been found significant, the block is labeled as significant and is partitioned into four sub-blocks so that the locations of the sub-blocks (or coefficients) may be identified. When the scanning and testing of all significant coefficients have been completed against the present threshold, it is halved, and the same procedure is repeated. The procedure terminates when the threshold value for the significance test exceeds the crossover threshold.

As an example, the coding algorithm is illustrated in Fig. 2. The significance of coefficients is tested against $T = 8$. As a result, the quadtree takes place in two levels as shown in the figure. The significant coefficients marked by '1' are to be coded by the bitplane coding for further refinement. If $T \geq T_c$, the process is repeated after the renewal of T by its half.

When the significance of a coefficient has been detected, its location and sign bit are encoded as soon as possible. During this quadtreeing process, the list of signifi-

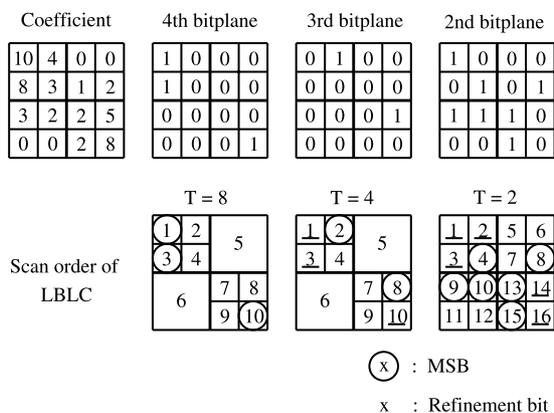


Fig. 3 Scanning order in successive significance tests.

cant pixels (LSP) [6], [7] is not defined unlike SPIHT and SPECK. Instead, every coefficient stays at its original location as it has been, regardless of being significant or insignificant. The coding process is hence simplified. It also allows the successive refinement for significant coefficients to progress in the same order as the significance discovery in which coarser subbands have been tested earlier.

Within a subband, a quadtree node is conditioned by the state of the 8-connected neighboring nodes with respect to the target node. The state is binary; it is 1, if at least three of them are one, otherwise the state is 0. The target node is encoded by an adaptive arithmetic coding depending on the state. While the context modeling of quadtrees is simple, the spatial correlation in a particular level of quadrissection is exploited in arithmetic coding for precise probability estimation.

The quadtree-based bitplane coding is terminated, when significant coefficients against the prescribed crossover threshold value cannot be found any more. The treatment of significant bits and refinement bits of large coefficients is illustrated in Fig. 3, where $T_c = 2$ is assumed. At first, three significance check bits for large coefficients are found to be significant against $T = 8$ during three quadrissections in two levels. Such a significance detection is marked with a circle in the figure. The coefficient scanning order is labeled in sequential numbers. That is, four sub-blocks produced by a quadrissection are traveled as the same as for subband scanning such as LL, HL, LH, and HH. Inside a single sub-block, coefficients are visited by the raster scanning. However, those three significance bits are not encoded, since the same information has been encoded as the location information in the quadtree. All of the lower bits with respect to their significance thresholds are actually encoded as the refinement bits. In Fig. 3, when T is decreased to 4, two coefficients are newly found to be significant. The refinement bits of those three large coefficients that have been detected earlier are encoded, and their bitplane locations are labeled by underscored numbers in Fig. 3.

Just before the large coefficient coding mode is switched over to the small coefficient coding mode, the remaining bits of those large coefficients at lower bitplanes are

encoded with the bitplane coding as the same as for their upper bits. The only difference from the upper bitplane coding is simply to skip the significance test so that insignificant locations are never visited again, while the refinement of large coefficients goes on. For example, in Fig. 3, since $T_c = 2$, their LSBs will be encoded in the pure refinement pass without significance tests.

Since the decoding procedure for large coefficients shares the same procedure as the encoding, decoding can be stopped either at any bit rate or at any SNR to make a desirable image rendition, as long as the decoded bits belong to the large-coefficient coding.

2.4 Coding of Small Coefficients

A wavelet transform provides signal energy compaction in a few subbands. If those highly energy-concentrated subbands are separated from the others, a significant extent of statistical bias is created in the coefficient distribution. This property has been exploited in the large-coefficient coding. On the other hand, the small coefficients widely distribute across and inside subbands, and populate as much more than large coefficients. They are almost uncorrelated from sample to sample and their correlation characteristics differ little between different subbands. Furthermore, as the subband decomposition comes to a coarser level, the number of coefficients in a subband decreases. It causes a difficulty in the probability estimation for entropy coding.

To overcome the poor correlation and the possibility of insufficient samples, hierarchical structure is introduced in the context modeling. Adaptive arithmetic coding is applied for the entropy coding of the small coefficients. The raster scanning is applied to the entire set of all subbands. It means that all of the small coefficients are assumed to be simple elements in a single set that covers an area of the whole picture. The magnitude of small coefficients distributes in the range of $[0, T_c - 1]$. Since the sign of small coefficients are encoded by a binary arithmetic coding as same as for large coefficients, the coding model has at most T_c symbols.

In this work, causal neighbors around a target coefficient and their parent coefficient are referenced in context modeling. Figures 4 and 5 illustrate two types of the context model which are referred to as block context and neighbor context, respectively. The modeling number depends on a combination of the block context and the neighbor context.

Consider the whole image block partitioned in 2×2 blocks. In Fig. 4, NW , N , NE , W , and P denote those blocks at the north-west, north, north-east, west, and its parent locations. Each block has the average value over four coefficients. The model parameter M_B at the current block X_B is defined by

$$M_B = \begin{cases} 0, & \text{for } \hat{X}_B < 1 \\ \lfloor \log_2 \hat{X}_B \rfloor + 1, & \text{otherwise} \end{cases} \quad (3)$$

where

$$\hat{X}_B = \frac{NW_B + N_B + NE_B + W_B + P_B}{5}. \quad (4)$$

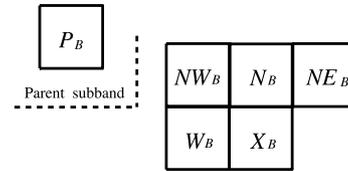


Fig. 4 Block context.

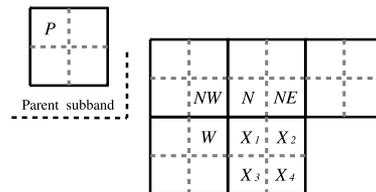


Fig. 5 Neighbor context.

If \hat{X}_B exceeds the crossover threshold T_c , it is clipped onto T_c . M_B is almost equivalent to the MSB position of the average value in a high-order context where $5 \times 4 = 20$ reference samples would be involved. Figure 5 shows a neighbor context that is formed by the nearest neighbors to a target coefficient. Each coefficient X_I ($I = 1, 2, 3$, or 4) has a different model parameter M_{Ni} in the target block X_B . It is defined as follows.

$$M_{Ni} = \begin{cases} 0, & \text{for } \hat{X}_I < 1 \\ \lfloor \log_2 \hat{X}_I \rfloor + 1, & \text{otherwise} \end{cases} \quad (5)$$

where

$$\hat{X}_I = \frac{NW + N + NE + W + P}{5}. \quad (6)$$

The average in Eq. (6) is modified to a 4-sample average for X_4 , because its north-east reference is unavailable. The context modeling class M_I of a coefficient X_I is now specified by a hierarchical characterization through M_B and M_{Ni} as follows.

$$M_I = M_B (\log_2 T_c + 1) + M_{Ni} \quad (7)$$

where $(\log_2 T_c + 1)$ is identical to the value excursion of M_{Ni} . A small coefficient X_I is encoded by the arithmetic coding model M_I . Note that the coding models for a subband differ from those for other different subbands. Separate coding models are applied to a different set of arithmetic coders. A combination of M_B and M_{Ni} is helpful to avoid the problem of context dilution [21], [22] caused by insufficient sample counts in individual contexts. The role of the block context is not only to make a sort of high-order reference but also to limit the number of context models.

2.5 Computational Complexity

The encoding complexity of the proposed method is briefly discussed. While the bitplane coding is somewhat expensive, only a few most significant bitplanes against a

crossover threshold are encoded in the large coefficient coding. The less significant bits of large coefficients are encoded without any overhead, because neither quadtreeing nor a significance test is necessary. As will be later shown, a typical crossover threshold value is $T_c = 2^5$. In this case, the number of encoded bitplanes is effectively five or less for 8-bit depth images, because $8 + 2 - 5 = 5$, where the two-bit increase is due to the highpass gain factor explained in Sect.2.2. This is valid, even if the subband shift is applied, since the coarser a subband is visited, the narrower the area becomes. The information in those data typically amounts to one tenth or less out of the full-length lossless bit stream. The other information is involved with small coefficients that are simply entropy-coded.

This makes a clear contrast to JPEG2000, where all bitplanes are encoded with the aid of sophisticated context modeling. In addition, the post-compression rate-distortion optimization [9] is evoked for the final bit truncation in EBCOT. In summary, the proposed method is considered to have lower complexity than JPEG2000.

3. Experimental Results

The proposed method is evaluated on several grayscale images. Their thumbnail images are displayed in Fig. 6. The lifting computation of the 9/7-tap reversible wavelet transform is applied in five levels and a reflection extension is implemented at the image border. The quadtree bits and the refinement bits of large coefficients are encoded by the same coding model. The signs of coefficients are encoded by a different coding model as a binary string. The number of symbols in the coding model for small coefficients is given by the crossover threshold value. Bit rates have been actually obtained by the adaptive arithmetic coding [23].

Figure 7 shows the experimental bit rates against crossover threshold values, that are represented in the bit-plane number. At the crossover threshold of 2^8 on the left side edge in the figure, the context-based entropy coding is fully applied to all data. In contrast, on the right side edge at $T_c = 2^0$, all of the wavelet coefficients are encoded by the embedded bitplane coding. The total bit rate shows a shallow dip. The value of the crossover threshold where the minimum bit rate appears is insensitive to different image contents, and it is around 2^5 . The plots suggest that the embedded coding of large coefficients and the context-based entropy coding for remaining data complement each other for lossless image coding.

As pointed out in Introduction, the inter-pixel correlation is helpful in lossless compression. In contrast, the zero-tree coding [5], [6] of the bitplanes of wavelet coefficients is very effective for a sparse distribution of significant bits. Hence, as the crossover threshold value decreases up to medium bitplanes, the coding efficiency improves. However it is not so efficient when significant bits are densely populated, because it decreases the chances that a parent node is a zero-tree root [5] for its four child nodes. Note that a zero-tree root is conceptually equivalent to quadtree



Fig. 6 Test images.

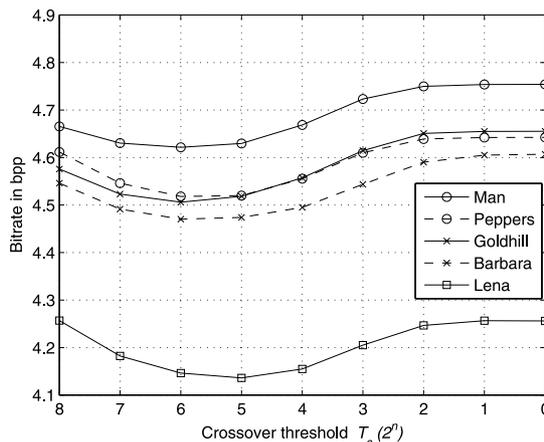


Fig. 7 Bit rates at several crossover thresholds.

pruning. On the other hand, the lower bitplanes of natural scene images show a random nature. Owing to this empirical fact, the zero-tree coding and quadtree-based coding will have little significance against the lower bitplanes. When the crossover threshold value decreases very close to the smallest limit, many bits appear to be significant and it results in a denser distribution of significant bits in lower bitplanes. By contrast, since those disordered lower bits are actually less-significant parts of many correlating multiple-valued coefficients, a context-based entropy coding of multiple-valued coefficients can still do to some degree, unless the excursion of coefficient values reduces too small. As a result of these facts, an optimum compromise will appear in making a negotiation between two different methods for lossy-to-lossless compression.

The compression performance is compared with three typical image coders. The first is JPEG-LS that is a representative among lossless predictive coders in the picture domain. The second is JPEG2000 that is in turn a representative among lossy/lossless scalable coders. In the experiments, the JLSREF version 1.00 [24] and JJ2000 version 4.1 [25] are used. JPEG2000 has been operated with its default parameters in the experiment. The third is S+P [8],

Table 1 Comparison of lossless bit rates in bpp. The integer in parentheses denotes the crossover bitplane.

Image	JPEG-LS	JPEG2000 (default)	JPEG2000 (int 9/7)	S+P	LBLC up-shifted	LBLC shift-free
Cameraman (256x256)	4.314	4.596	4.631	4.476	4.494 (4)	4.475 (4)
Girl (256x256)	4.581	4.749	4.715	4.530	4.508 (5)	4.481 (4)
MoonSurface (256x256)	5.082	5.296	5.288	5.048	5.074 (5)	5.034 (5)
Airplane (512x512)	3.817	4.047	4.003	3.914	3.909 (5)	3.883 (5)
Baboon (512x512)	6.036	6.147	6.126	5.929	5.906 (6)	5.869 (6)
Lena (512x512)	4.237	4.338	4.299	4.163	4.136 (5)	4.120 (5)
Peppers (512x512)	4.513	4.665	4.662	4.583	4.518 (6)	4.508 (5)
Barbara (720x576)	4.691	4.639	4.521	4.527	4.470 (6)	4.448 (5)
Goldhill (720x576)	4.477	4.639	4.638	4.557	4.506 (6)	4.481 (5)
Man (1024x1024)	4.694	4.852	4.826	4.658	4.622 (6)	4.609 (5)
Average	4.644	4.797	4.771	4.639	4.614	4.591

Table 2 Lossless and lossy coding performance at several crossover thresholds on *Lena*.

T_c (bitplane)	Lossless bit rate (bpp)	Lossy bit rate (bpp)	PSNR (dB)
14	4.413	-	-
13	4.412	0.000	14.52
12	4.410	0.001	16.81
11	4.402	0.003	19.21
10	4.380	0.008	21.67
9	4.336	0.020	24.13
8	4.257	0.047	26.78
7	4.183	0.100	29.70
6	4.146	0.205	32.64
5	4.136	0.405	35.59
4	4.155	0.836	38.68
3	4.205	1.753	42.68
2	4.247	2.955	47.87
1	4.257	3.961	55.19
0	4.256	4.256	Infinity

and it is a context-based entropy coder which is based on a progressive coding in the transform domain.

The lossless bit rates are listed in Table 1, where the proposed method is referred to as LBLC. The lossless bit rates in rightmost two columns in the table are followed by (b) , and b denotes the bitplane corresponding to the crossover threshold at which value the lowest bit rate is obtained. *Up-shifted* refers to the subband shifted operation, while *shift-free* refers to the other optional operation where the subband shift is suppressed. The bit rates for JPEG2000 are listed in two columns. The first of them refers to the lossless mode of JPEG2000 that is based on the reversible 5/3-tap wavelet system [9], [10]. The other of them is the result obtained by replacing the wavelet system with that of Eqs. (1), (2). One can see that the improvement in bit rates caused by the different wavelet transform is about 0.025 bpp on average.

As is observed, the proposed method slightly outperforms the other three codecs on average. Although the subband shifted LBLC is slightly inferior to the shift-free operation, its rate-distortion performance is fine as will be shown in Fig. 9.

Table 2 shows lossless and lossy coding performance

**Fig. 8** Example of progressive decoding. *Lena* has been encoded with $T_c = 2^5$, and is decoded at 0.405 bpp, resulting in PSNR = 35.59 dB. The displayed image is a demonstration of spatial scalability having a 256×256-pixel rendition of which resolution is a half the original.

at several crossover thresholds on *Lena*. The lossy bit rate in the table is a part of information involved with the full bit stream of which lossless bit rate is found in the same row. When the bit stream is partially decoded at a particular lossy bit rate, the decoded image has the quantization error listed in the column of PSNR. As the lossy bit rate increases, PSNR monotonically increases and finally reaches infinity, where the lossless decoding is presented. When the crossover threshold is set at the 5th bitplane on *Lena*, the best lossless bit rate of 4.136 bpp has been attained. At the same time, if it is partially decoded at 0.405 bpp, a high-fidelity image recovery is obtained with PSNR of 35.59 dB.

Figure 8 is such a decoded image of which rendition is of a half the original resolution. To demonstrate the spatial scalability as well as partial decodability, it is thus rendered with a fractional bit rate out of 0.405 bpp. No objectionable artifacts are visually perceived in the image. As demonstrated in this typical example, and by paying attention to Fig. 7, one can conclude that it is reasonable in practice to

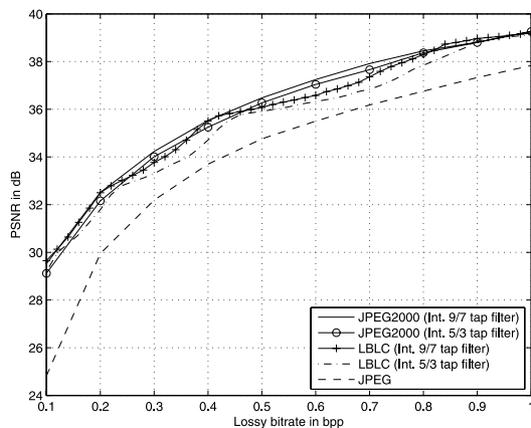


Fig. 9 Lossy bit rates vs. PSNR on *Lena*.

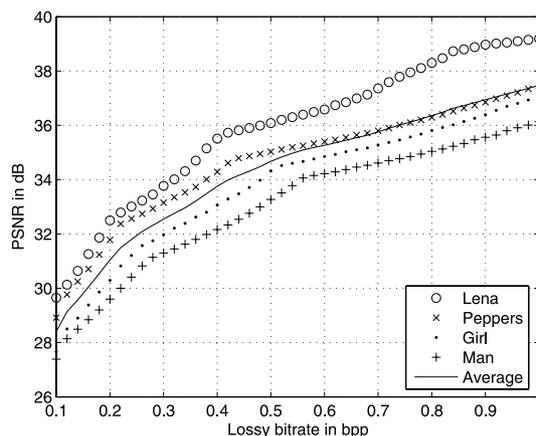


Fig. 10 Lossy bit rates vs. PSNR for different images.

choose the crossover threshold around 2^5 for acceptable visual appearance in progressive decoding.

The rate-distortion curves for LBLC, JPEG2000, and JPEG [26] are shown in Fig. 9. A pair of plots for JPEG2000 correspond to two different wavelet systems, and have been obtained with the 9/7-tap and 5/3-tap reversible wavelet transforms. The former wavelet is the same given by Eqs. (1), (2), and the latter is the default system in accordance with the recommendation for lossy-to-lossless seamless coding. Our proposed codec, LBLC, has been also operated with the same two wavelet systems. The 9/7-tap reversible system results in slightly better performances in both codecs of LBLC and JPEG2000. The performance of the proposed method equipped with the 9/7-tap system is found to be competitive to that of the lossy-to-lossless mode of JPEG2000.

Rate distortion plots for some images are presented in Fig. 10 just for reference purposes to possible variations among different image contents.

The total encoding time for *Lena* on an Intel Core 2 Duo processor (E6600, 2.4 GHz) is listed in Table 3, where two columns for JJ2000 and Kakadu [27] are also included for reference purposes. The rightmost three columns apply

Table 3 Comparison of lossless encoding time at some crossover thresholds.

Image	Lossless encoding time (sec)				
	JJ2000 ver. 4.1	Kakadu ver. 5.2	LBLC		
			$T_c =$ MSB	$T_c =$ 32	$T_c =$ 1
Cameraman	0.58	0.03	0.02	0.02	0.04
Girl	0.55	0.03	0.02	0.02	0.04
MoonSurface	0.65	0.04	0.02	0.02	0.04
Airplane	1.04	0.07	0.09	0.09	0.16
Baboon	0.99	0.11	0.10	0.12	0.22
Lena	0.82	0.08	0.09	0.09	0.18
Peppers	0.71	0.08	0.09	0.09	0.17
Barbara	1.07	0.12	0.14	0.16	0.30
Goldhill	1.24	0.12	0.14	0.14	0.30
Man	1.57	0.31	0.40	0.42	0.77

to the proposed system.

A half or more processing time is spent for adaptive arithmetic coding in the proposed system. The coding time in the case of $T_c = 1$ corresponds to that of purely embedded coding without the context-based coding. It is approximately twice long to that of the normal operation provided with a mixture of the embedded coding and the context-based coding. There is very little difference between the cases of $T_c = \text{MSB}$ and 32, since a small number of code bits are produced in the embedded coding, if $T_c \geq 2^5$.

As seen in the table, the encoding speed of the proposed codec referred to as LBLC is faster than that of JJ2000, because JJ2000 is written in Java and LBLC is in C. The encoding time of LBLC is quite close to that of Kakadu. It is quite reasonable, since Kakadu has been brushed up during its history. In contrast, neither optimization nor improvement has been implemented for speeding up LBLC. The processing speed highly depends on different code implementations and an exact comparison on encoding time is beyond the scope of this paper.

4. Conclusions

A lossless image coding that is scalable with respect to resolution, bit rate and PSNR has been presented. It consists of a reversible wavelet transform, quadtree-based bitplane coding, and context-based arithmetic coding with a hierarchical modeling. In lossless compression, the proposed method slightly outperforms JPEG-LS and the lossless operation of JPEG2000. With respect to the rate-distortion performance in progressive decoding it is competitive to the JPEG2000 lossy-to-lossless mode. The progressive decoding capability for scalable resolution, PSNR, and bit rates is equipped with the losslessly encoded bit stream. Even if those scalabilities are limited, they are considered to offer an acceptable level in practical use. The encoding complexity of the proposed method is estimated to be slightly lower than that of JPEG2000.

An efficient treatment of super high-definition images and a quantization scheme for perceptual quality control are left as open issues for coming applications.

Acknowledgements

The authors thank anonymous reviewers for their helpful comments to improve the quality of presentation of this work.

References

- [1] ISO/IEC 14495-1:1999, Information technology—Lossless and near-lossless compression of continuous-tone still images: Baseline, Dec. 1999.
- [2] M.J. Weinberger, G. Seroussi, and G. Sapiro, “The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS,” *IEEE Trans. Image Process.*, vol.9, no.8, pp.1309–1324, Aug. 2000.
- [3] B. Meyer and P. Tischer, “TMW—A new method for lossless image compression,” *Proc. 1997 Picture Coding Symp. (PCS’97)*, pp.533–538, Berlin, Germany, Sept. 1997.
- [4] I. Matsuda, Y. Umezū, N. Ozaki, J. Maeda, and S. Itoh, “A lossless coding scheme using adaptive predictors and arithmetic code optimized for each image,” *IEICE Trans. Inf. & Syst. (Japanese Edition)*, vol.J88-D-II, no.9, pp.1798–1807, Sept. 2005.
- [5] J.M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Trans. Signal Process.*, vol.41, no.12, pp.3445–3462, Dec. 1993.
- [6] A. Said and W.A. Pearlman, “A new, fast and efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits Syst. Video Technol.*, vol.6, no.3, pp.243–250, June 1996.
- [7] 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, no.11, pp.1219–1235, Nov. 2004.
- [8] A. Said and W.A. Pearlman, “An image multiresolution representation for lossless and lossy compression,” *IEEE Trans. Image Process.*, vol.5, no.9, pp.1303–1310, Sept. 1996.
- [9] D. Taubman, “High performance scalable image compression with EBCOT,” *IEEE Trans. Image Process.*, vol.9, no.7, pp.1158–1170, July 2000.
- [10] D.S. Taubman and M.W. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards, and Practice*, Kluwer Academic Publishers, Boston, 2002.
- [11] R.S. Weinstein, M.R. Descour, L. Chen, A.K. Bhattacharyya, A.R. Graham, J.R. Devis, K.M. Scott, L. Richter, E.A. Krupinski, J. Szymus, K. Kayser, and B.E. Dunn, “Telepathology overview: From concept to implementation,” *Human Pathology*, vol.32, no.12, pp.1283–1299, Dec. 2001.
- [12] U. Catalyurek, M.D. Beynon, C. Chang, T. Kurc, A. Sussman, and J. Saltz, “The virtual microscope,” *IEEE Trans. Inf. Technol. Biomed.*, vol.7, no.4, pp.230–248, Dec. 2003.
- [13] I. Tofukuji, A study on the qualifications of microscopic imaging for intra-operative diagnosis telepathology, Ph.D. Dissertation, Niigata University, 2004.
- [14] S. Takamura and M. Takagi, “Lossless image compression with lossy image using adaptive prediction and arithmetic coding,” *Proc. Data Compression Conf.*, pp.166–174, March 1994.
- [15] N.D. Memon, K. Sayood, and S.S. Magliveras, “Simple method for enhancing the performance of lossy plus lossless image compression schemes,” *J. Electronic Imaging*, vol.2, no.3, pp.245–252, July 1993.
- [16] K. Shinoda, H. Kikuchi, and S. Muramatsu, “A study on a lossless image compression by lossy coding,” *Proc. 20th Picture Coding Symp. of Japan (PCSJ2005)*, pp.69–70, Hamamatsu, Japan, Nov. 2005.
- [17] K. Shinoda, H. Kikuchi, and S. Muramatsu, “A lossless-by-lossy approach to lossless image compression,” *IEEE Int. Conf. on Image Processing (ICIP 2006)*, pp.2265–2268, Atlanta, USA, Oct. 2006.
- [18] A. Calderbank, I. Daubechies, W. Sweldens, and B.L. Yeo, “Lossless image compression using integer to integer wavelet transforms,” *Proc. IEEE Int. Conf. on Image Processing (ICIP’97)*, vol.1 of 3, pp.596–599, Washington, DC, USA, 1997.
- [19] M.D. Adams and F. Kossentini, “Reversible integer-to-integer wavelet transforms for image compression: Performance evaluation,” *IEEE Trans. Image Process.*, vol.9, no.6, pp.1010–1024, June 2000.
- [20] G. Strang and T.Q. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, Cambridge, MA, USA, 1997.
- [21] X. Wu, “Lossless compression of continuous-tone images via context selection, quantization, and modeling,” *IEEE Trans. Image Process.*, vol.6, no.5, pp.656–664, May 1997.
- [22] X. Wu and N. Memon, “Context-based, adaptive, lossless image coding,” *IEEE Trans. Commun.*, vol.45, no.4, pp.437–444, April 1997.
- [23] I.H. Witten, R.M. Neal, and J.G. Cleary, “Arithmetic coding for data compression,” *Commun. ACM*, vol.30, no.6, pp.520–540, 1987.
- [24] “LOCO-I/JPEG-LS software, JPEG-LS reference encoder—ver. 1.00, (c) Copyright Hewlett-Packard Company, 1995–1999, HP Labs LOCO-I/JPEG-LS home page,” available at: <http://www.hpl.hp.com/loco/>, Feb. 2007.
- [25] “An implementation of the JPEG 2000 standard in java, version 4.1,” available at: <http://jj2000.epfl.ch/>, Feb. 2007.
- [26] ITU-T T.81, ISO/IEC 10918-1:1994, *Digital Compression and Coding of Continuous-Tone Still Images: Requirements and Guidelines*, Feb. 1994.
- [27] <http://www.kakadusoftware.com/>



Kazuma Shinoda was born in Niigata, Japan in 1983. He received B.E. and M.E. degrees in electrical and electronic engineering from Niigata University, Niigata, in 2005 and 2007, respectively. In 2007, he joined Sony Corporation, Atsugi, where he works for the research and development of master monitor displays. His research interests include digital signal processing and image/video processing.



Hisakazu Kikuchi was born in Niigata in 1952. He received B.E. and M.E. degrees from Niigata University, Niigata, in 1974 and 1976, respectively, and Dr. Eng. degree in electrical and electronic engineering from Tokyo Institute of Technology, Tokyo, in 1988. From 1976 to 1979 he worked at Information Processing Systems Laboratory, Fujitsu Ltd., Tokyo. Since 1979, he has been with Niigata University, where he is a Professor in the Department of Electrical and Electronic Engineering. During

the 1992–1993 academic year, he was a visiting scholar in the Electrical Engineering Department, University of California, Los Angeles, USA. He holds a visiting professorship at Chongqing University of Posts and Telecommunications and Nanjing University of Information Science and Technology, both in China, since 2002 and 2005, respectively. His research interests include digital signal processing, image/video processing, and wavelets as well as UWB communication systems. He is a Member of ITE (Institute of Image Information and Television Engineers of Japan), IIEEJ (Institute of Image Electronics Engineers of Japan), CADM (Society of Computer-Assisted Diagnosis of Medical Images), IEEE and SPIE. He served the chair of Circuits and Systems Group, IEICE, in 2000 and the general chair of Digital Signal Processing Symposium, IEICE, in 1998 and Karuizawa Workshop on Circuits and Systems, IEICE, in 1996, respectively.



Shogo Muramatsu was born in Tokyo, Japan in 1970. He received B.E., M.E. and Dr. Eng. degrees in electrical engineering from Tokyo Metropolitan University in 1993, 1995 and 1998, respectively. In 1997, He joined Tokyo Metropolitan University. In 1999, he joined Niigata University, where he is currently an Associate Professor in the Department of Electrical and Electronic Engineering. He was a Visiting Researcher at MICC & VIP Laboratory, University of Florence, Italy, during the 2003–2004

academic year. His research interests are in digital signal processing, multirate systems, image processing and VLSI architecture. He is a Member of ITE (Institute of Image Information and Television Engineers), IIEEJ (Institute of Image Electronics Engineers of Japan), IPSJ (Information Processing Society of Japan), and IEEE.