An Image Multiresolution Representation for Lossless and Lossy Compression

Amir Said, *Member, IEEE*, and William A. Pearlman, *Senior Member, IEEE*

***Abstract*—We propose a new image multiresolution transform that is suited for both lossless (reversible) and lossy compression. The new transformation is similar to the subband decomposition, but can be computed with only integer addition and bit-shift operations. During its calculation, the number of bits required to represent the transformed image is kept small through careful scaling and truncations. Numerical results show that the entropy obtained with the new transform is smaller than that obtained with predictive coding of similar complexity. In addition, we propose entropy-coding methods that exploit the multiresolution structure, and can efficiently compress the transformed image for progressive transmission (up to exact recovery). The lossless compression ratios are among the best in the literature, and simultaneously the rate *versus* distortion performance is comparable to those of the most efficient lossy compression methods.**

I. Introduction

THERE are important image applications where some processing (e.g., subtraction, filtering, contrast enhancement, etc.) should be applied to archived or transmitted images. In those cases, lossy compression methods may destroy some of the information required during processing, or add artifacts that lead to erroneous interpretations. Quite frequently, the user of those applications wants to have total control of the precision in which the image pixels are represented and prefers to have the image compressed with a lossless (or reversible) method. Lossless compression is also indicated for images obtained at great cost, such as space and medical images, when it is unwise to discard any information that may be useful later. Nevertheless, images are frequently visually inspected, and it is interesting to have a compression scheme that simultaneously allows fast inspection and, only when necessary, exact recovery of the image. Traditionally, the user had to choose different coding methods depending whether the highest compression or fast inspection was desired [2]. In this paper, we propose efficient coding methods that achieve both of those objectives.

We consider two schemes of progressive transmission (or recovery) for fast inspection: *progressive fidelity* and *progressive resolution.* In the first case, we assume that the image is observed at its full size. Initially, only the main components of the image are transmitted and shown, and some form of interpolation is used to cover the missing details, which are progressively added. This way the image quality is gradually improved until perfect reconstruction. In the progressive resolution transmission scheme, an image with reduced resolution (to be displayed in a small size) is transmitted first. Afterward, the information to obtain images with increasing dimension or resolution (i.e., larger viewing area or pixels/area) from the smaller versions is transmitted. This is useful when several images are simultaneously displayed for inspection (e.g., in a 4×4 array), and later magnified or processed.

Some of the most effective methods for lossless compression use linear predictive coding [1], [2], which has been adopted for lossless compression in the Joint Photographic Experts Group (JPEG) still picture compression standard [3]. This form of compression is usually defined for a single resolution, and in a way that the image can only be recovered in its entirety, which impedes fast inspection. Several *ad hoc* methods have been proposed for lossless compression with progressive fidelity transmission [4], [5], but their performance is much inferior to the lossy compression methods. Recently, a tree-structured vector quantizer was proposed by Effros *et al.* [6] for progressive-fidelity transmission, which should provide good quality images at low bit rates but is not efficient for lossless compression. More efficient fast inspection can be obtained with the *lossy-plus-residual* methods [1], [20].

Excellent lossy compression results have been obtained using the wavelet transform [7], [8]. In an image context, it produces a multiresolution representation, which has been shown to be naturally suited for progressive transmission. One multiresolution transform for lossless compression is known in the medical imaging community as the *sequential transform* (S-transform) [1], [9], [11]. Another method that enables progressive resolution transmission is called *hierarchical interpolation* (HINT) [2], [12]. These transformations are fairly efficient, but some studies [2] show that they may not be as effective as predictive coding.

In this paper, we propose a new multiresolution transformation for both lossless and lossy compression called the *S+P-transform,* the "P" for "prediction." Numerical results show that the S+P-transform yields more compression than single-resolution linear predictive coding methods of similar complexity, and can be calculated with a very small computational effort. Furthermore, we propose entropy-coding methods that exploit the multiresolution structure and that can efficiently compress the S+P-transformed image for progressive-

Manuscript received August 12, 1994; revised January 22, 1996. This paper was presented in part at the SPIE Symposium on Visual Communications and Image Processing, Cambridge, MA, Nov. 1993. The associate editor coordinating the review of this paper and approving it for publication was Dr. Amy R. Reibman.

A. Said is with the Faculty of Electrical Engineering, State University of Campinas (UNICAMP), Campinas, SP 13081 Brazil.

W. A. Pearlman is with the Dept. of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: pearlman@ecse.rpi.edu).

Publisher Item Identifier S 1057-7149(96)06027-7.

1057–7149/96$05.00 © 1996 IEEE

INTEL-1057
9,641,841

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

resolution transmission. For progressive-fidelity transmission we propose an embedded coding method, and show that its rate distortion function is comparable to those of the most efficient lossy compression methods. The compression rates obtained with both types of progressive transmission are among the best in the literature. Thus, we show that, with the proper image transformation, fast inspection schemes can be readily combined with lossless compression, resulting in a negligible penalty in both compression efficiency and coding complexity.

This paper is organized as follows. Section II describes the general form of the S+P-transform. In Section III, we consider the optimization of some of its parameters using a frequency-domain analysis, and compare its entropy with those obtained by other transforms. In Section IV, we propose coding methods for progressive-resolution transmission, and compare the lossless compression rates with those obtained with other entropy-coding methods. The coding method for progressive-fidelity transmission, together with rate *versus* peak signal-to-noise ratio (PSNR) results, are presented in Section V. Section VI contains information about accessing the source code of the codecs and concludes the paper.

II. The S+P-Transform

We now present the S-transform, which is similar to the Haar multiresolution image representation [10]. There are different definitions of the S-transform in the literature, but most differ only in some implementation details.

A sequence of integers $c[n]$, $n = 0, \cdots, N-1$, with N even, can be represented by the two sequences

$$l[n] = \left\lfloor \frac{c[2n] + c[2n+1]}{2} \right\rfloor, \quad n = 0, \cdots, \frac{N}{2} - 1$$
$$h[n] = c[2n] - c[2n+1], \quad n = 0, \cdots, \frac{N}{2} - 1 \tag{1}$$

where $\lfloor \cdot \rfloor$ corresponds to downward truncation. The sequences $l[n]$ and $h[n]$ form the S-transform of $c[n]$. Since the sum and difference of two integers correspond to either two odd or two even integers, the truncation is used to remove the redundancy in the least significant bit. The division and downward truncation can be done with a single bit-shift. Note that $l[n]$ and $h[n]$ can use the memory space used by $c[n]$.

The inverse transformation is

$$c[2n] = l[n] + \left\lfloor \frac{h[n]+1}{2} \right\rfloor$$
$$c[2n+1] = c[2n] - h[n]. \tag{2}$$

To see the advantage of this representation, let us consider that $c[n]$ is a zero-mean random sequence with

$$E\{c^2[n]\} = \sigma^2$$
$$E\{c[n]c[n+1]\} = \sigma^2 \rho \tag{3}$$

where $E\{\cdot\}$ is the expectation operator. If we disregard the truncation, we have

$$E\{l^2[n]\} = \sigma^2 \frac{1+\rho}{2}$$
$$E\{h^2[n]\} = 2\sigma^2(1-\rho). \tag{4}$$

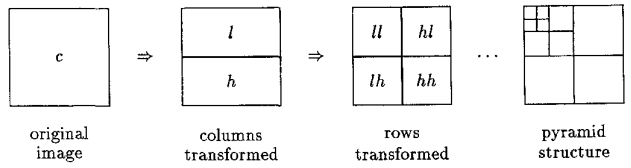


Fig. 1. Construction of an image multiresolution pyramid from one-dimensional (1-D) transformations.

If $c[n]$ is formed with the gray-scale values in an image row or column, we have $\rho \approx 1$, and $h[n]$ normally has a very small variance, while the variance of $l[n]$ is approximately equal to the variance of $c[n]$. Furthermore, we have

$$\tfrac{1}{2}\left(E\{l^2[n]\} + E\{h^2[n]\}\right) = \tfrac{1}{4}\sigma^2(5 - 3\rho) \tag{5}$$

which means that the average variance of $l[n]$ and $h[n]$ is smaller than the variance of $c[n]$ when $\rho > 1/3$. Thus, even though we expect to have $E\{l[n]l[n+1]\} < \sigma^2\rho$, it is usually advantageous to apply the same decomposition to $l[n]$.

The two-dimensional (2-D) transformation is done by applying the transformation (1) sequentially to the rows and columns of the image, as shown in Fig. 1. The coefficients corresponding to ll in Fig. 1 are the mean of 2×2 pixel blocks, and they form another image with half the resolution. The same transformation is applied to these reduced resolution "mean images" to form a hierarchical pyramid [7].

Note that the maximum number of bits required to represent each pixel in the ll images does not change with each transformation. For example, if the gray-level original image has 8 b/pixel, the reduced ll image also has 8 b/pixel. On the other hand, the other pixels require a signed representation with a larger number of bits. Except for the truncations in (1) and (2), this transformation corresponds to a subband decomposition [11]. The low resolution (ll) images are formed with mean values (a form of lowpass filtering), which reduces aliasing, and is superior to unfiltered subsampling used by linear interpolation methods [2], [12].

The S-transform is simple, can be very efficiently calculated, and significantly reduces the first-order entropy. However, it leaves a residual correlation between the highpass components, which is due to aliasing from the low-frequency components of the original image. Hence, we could expect an improvement if better filters were used. However, arithmetic operations with integer numbers create a statistical dependence in the least significant bits, which is irrelevant for lossy compression, but that must be removed for efficient lossless compression. This means that for lossless compression we must always pay attention to the truncation.

To solve this problem, we can use the fact that predictive coding does not have to be linear for perfect reconstruction. Hence, the prediction value can be truncated to an integer. Thus, we can improve the S-transform with predictive coding. However, instead of using prediction in the final S-transformed pyramid, in the S+P-transform we use, during each one-dimensional (1-D) transformation, some values of $l[n]$ and $h[n]$ to estimate the value of a given $h[n_o]$. Calling the estimates

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

$\hat{h}[n]$, the differences $h_d[n]$, where

$$h_d[n] = h[n] - \left\lfloor \hat{h}[n] + \frac{1}{2} \right\rfloor, \quad n = 0, 1, \cdots, \frac{N}{2} - 1 \quad (6)$$

replace $h[n]$, forming a new transformed image with smaller first-order entropy. No estimation is subtracted from the sequence $l[n]$ because it forms the reduced-resolution image, which can be later transformed with the same method.

Defining

$$\Delta l[n] = l[n-1] - l[n] \quad (7)$$

the general form of our estimation is[1]

$$\hat{h}[n] = \sum_{i=-L_0}^{L_1} \alpha_i \Delta l[n+i] - \sum_{j=1}^{H} \beta_j h[n+j]. \quad (8)$$

We use $\Delta l[n]$ instead of $l[n]$ to have zero-mean estimation terms, and thus there is no need to subtract the mean from $c[n]$. Note that the index i can be negative because $l[n]$ is not replaced by a prediction error. To simplify the notation we disregard, for now, the image borders.

During the inverse 1-D transformation, the prediction can be added following a reverse order

$$h[n] = h_d[n] + \left\lfloor \hat{h}[n] + \frac{1}{2} \right\rfloor, \quad n = \frac{N}{2} - 1, \frac{N}{2} - 2, \cdots, 0 \quad (9)$$

so that the values of $h[n]$ required to calculate the prediction for the current n have already been recovered. The inverse 1-D S-transform (2) is calculated after the sequence $h[n]$ is recovered. The 2–D S+P-transform is also implemented by applying the 1-D S+P-transformation sequentially to the columns and rows of the image. However, note that (6) is not linear due to the truncation, and this makes the order of transformations important. For instance, if the transformation was applied first to the columns and then to the rows, the inverse transformation must be applied first to the rows and then to the columns. In short, the inverse transformation algorithm is just like the transformation algorithm running "backward."

III. Selection of the Predictor Coefficients

We have studied three schemes for the determination of the predictor coefficients α_i and β_j in (8): minimum entropy, minimum variance, and frequency domain design. The coefficients that minimize the (first-order) entropy can be found with the Nelder–Mead simplex algorithm [13], but their calculation requires a computational effort too large for practical applications. They are used as a benchmark, to evaluate the performance of the other schemes. The coefficients that minimize the variance of $h_d[n]$ can be found by solving Yule–Walker equations [2]. However, this common approach does not necessarily minimize the entropy of the S+P-transformed image, and numerical results have shown that even with high-order adaptive predictors the minimum-variance schemes were inferior to fixed predictors designed in the frequency-domain, the approach that is explained in the rest of this section.

If we disregard the truncations, we can combine (1) and (6)–(8), and see that $h_d[2n]$ can be regarded as the output of a noncausal FIR filter to the input sequence $c[n]$, which is subsampled by a factor of two. The z-transform of the filter response is

$$F(z) = (z-1)\left[\frac{(z^{-1}+1)^2}{2} \sum_{i=-L_0}^{L_1} \alpha_i z^{2i} - \sum_{j=1}^{H} \beta_j z^{2j} - 1\right]. \quad (10)$$

It is unusual to have a noncausal response with predictive coding, but as explained in Section II, this is possible because the values of $l[n]$ are not replaced by the prediction.

Since most of the image energy is normally concentrated in the low frequencies, to reduce the variance of $h_d[n]$ we should select a filter with a strong attenuation in the low frequencies. However, due to the structure of (10) and the requirements for a reversible transformation, stronger attenuation in the low frequencies inevitably leads to a larger gain in the high frequencies. For example, by selecting the gain $F(z)|_{z=-1}$, we can obtain sets of frequency responses like those shown in Fig. 2. In theory the choice for the best predictor depends on the image's characteristics: smooth and noiseless images are better compressed using the filter with the largest attenuation in the low frequencies, while noisy and very detailed images require a filter with small gain at the high frequencies. However, it has been observed that the entropy has a low sensitivity to the predictor parameters, and that predictor parameter choice is not critical [14]. Thus, there are good "universal" predictors, i.e., those that are effective for a broad class of images (e.g., portraits, landscape, medical, etc.). After extensive tests with different types of images, we selected the two predictors with coefficients listed in Table I. Predictor A has the smallest computational complexity, B is indicated for natural images, and C for very smooth medical images. They provide very good performance, with the additional advantage that bit-shifts can be used instead of multiplications or divisions. For instance, predictor B can be calculated as

$$\hat{h}[n] = \tfrac{1}{8}\{2(\Delta l[n] + \Delta l[n+1] - h[n+1]) + \Delta l[n+1]\}. \quad (11)$$

In the image borders, we use the predictors

$$\hat{h}[0] = \frac{\Delta l[1]}{4}$$
$$\hat{h}\left[\frac{N}{2} - 1\right] = \frac{\Delta l\left[\frac{N}{2} - 1\right]}{4}. \quad (12)$$

We tested the selected predictors on the set of images listed in Table II. The first four are well known, and can be seen in [17], [19], and [20]. The others are medical images. The X-ray images are the same as those used in [6]. The first-order entropy of the S+P-transformed image (six-level pyramid) is

[1] See Section VI for information about how to obtain the source code with the S+P-transform.

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

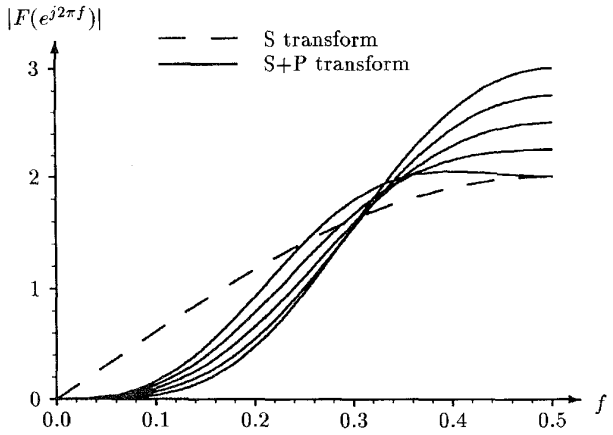


Fig. 2. Frequency response with different predictors in the S+P-transform; there is a compromise between attenuation of the low frequencies and amplification of the high frequencies.

TABLE I
COEFFICIENTS OF THE SELECTED PREDICTORS FOR THE S+P-TRANSFORM

predictor	parameter			
	α_{-1}	α_0	α_1	β_1
A	0	1/4	1/4	0
B	0	2/8	3/8	2/8
C	−1/16	4/16	8/16	6/16

TABLE II
SET OF IMAGES USED

Name	origin	dimension	gray levels
Girl, Couple	USC	256 × 256	256
Lena, Mandrill	USC	512 × 512	256
CT 1 to 4	tomography	512 × 512	4096
XR 1	X-ray	256 × 256	4096
XR 2	X-ray	1024 × 1024	256

shown in Table III. These entropies are the weighted means of the entropies in each of the pyramid levels—a more accurate estimate of the bit rates when adaptive entropy coding is used. The labels A–C in each column indicate the corresponding predictor in Table I. The label P* indicates the minimum-entropy sixth-order predictor found with the Nelder–Mead simplex algorithm for that image. To give a reference for comparisons, Table III also shows the first-order entropy obtained with the lossless JPEG third-order predictors #4 and #7 [3] and with the HINT method [12].

We see that the difference between the S-transforms and S+P-transforms is significant, and that the S+P consistently yields the smallest values. Also, note that the predictors B and C come very close to the optimal, represented by P*. It can be argued that the S+P entropy in some cases is not much smaller than JPEG's. However, the capability of progressive transmission is quite important and should be taken into account. Compared to the HINT method, the S+P-transform is more advantageous because its reduced-resolution

TABLE III
COMPARATIVE EVALUATION OF THE FIRST-ORDER ENTROPY (B/PIXEL) OBTAINED WITH DIFFERENT IMAGE TRANSFORMATIONS. B AND C ARE THE PREDICTORS DEFINED IN TABLE I AND P* IS THE MINIMUM-ENTROPY OPTIMAL PREDICTOR. THE JPEG NUMBERS INDICATE THE DPCM PREDICTORS

image	S transf.	S+P transf.				JPEG		HINT
		A	B	C	P*	4	7	
Girl	5.04	4.71	4.64	4.66	4.63	5.06	4.80	4.74
Couple	4.45	4.23	4.16	4.19	4.15	4.24	4.38	4.47
Lena	4.77	4.41	4.33	4.35	4.31	4.81	4.62	4.53
Mandrill	6.15	5.99	5.91	5.94	5.91	6.43	6.08	6.12
CT 1	4.30	3.74	3.50	3.37	3.34	3.77	4.11	4.04
CT 2	6.62	6.00	5.83	5.67	5.66	5.95	6.40	6.24
CT 3	5.64	5.10	4.95	4.80	4.79	5.32	5.47	5.32
CT 4	5.89	5.25	5.03	4.92	4.90	5.33	5.71	5.58
XR 1	4.91	4.41	4.17	4.08	4.02	4.56	4.68	4.64
XR 2	4.12	4.02	4.10	4.21	3.99	4.99	4.34	4.22
mean	5.18	4.79	4.65	4.60	4.55	5.03	5.05	4.99

images have better quality and because it can be used for progressive-fidelity transmission (see Section V).

IV. ENTROPY CODING FOR PROGRESSIVE-RESOLUTION TRANSMISSION

The progressive-resolution transmission schemes are easily implemented from the multiresolution transform because, in this case, the encoder just has to code the pixels beginning from the highest level of the pyramid. The decoder, after receiving the data up to level l, can recover an image with dimensions 2^l smaller than the original.

For entropy coding the S+P-transform, we use the fact that there is a statistical dependence between pixels of the transformed image that cannot be further reduced by linear predictive methods, but that can be exploited during coding. In practice, we should also pay attention to the complexity of the coding methods and observe that there are components of the transformed image that cannot be efficiently compressed and may be transmitted uncoded. This fact was used to define one entropy-coding method in the JPEG still picture compression standard [3].

In JPEG's method, an integer value is decomposed in three parts: the length in bits, the sign, and a magnitude difference. The magnitude difference is the difference between the actual magnitude and the lowest magnitude in a particular predefined set of transform pixel magnitudes. The length, which is the number of bits needed to express the sign and this magnitude difference, is entropy coded, forming the variable-length code (VLC), and then the sign and the magnitude difference are transmitted uncoded in the variable-length integer (VLI) format. (See [3] for more details.) With this representation, there is a small loss due to the fact that the VLI's are not entropy coded, but with the advantage that the number of VLC symbols is small, which simplifies the entropy-coding process. In other words, with this representation we can get bit rates near those that would be obtained if the complete integers were entropy coded, but with a smaller complexity.

We use the same approach to entropy code the S+P-transform. However, to reduce the loss that must occur with the uncoded transmission of part of the numbers, we propose

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
DEFINITION OF THE MAGNITUDE-SET VARIABLE-LENGTH INTEGER (MS-VLI) REPRESENTATION

magn. set (MS)	amplitude intervals	sign bit	magn. bits
0	[0]	no	0
1	[−1], [1]	yes	0
2	[−2], [2]	yes	0
3	[−3], [3]	yes	0
4	[−5,−4], [4,5]	yes	1
5	[−7,−6], [6,7]	yes	1
6	[−11,−8], [8,11]	yes	2
7	[−15,−12], [12,15]	yes	2
8	[−23,−16], [16,23]	yes	3
9	[−31,−24], [24,31]	yes	3
⋮	⋮	⋮	⋮

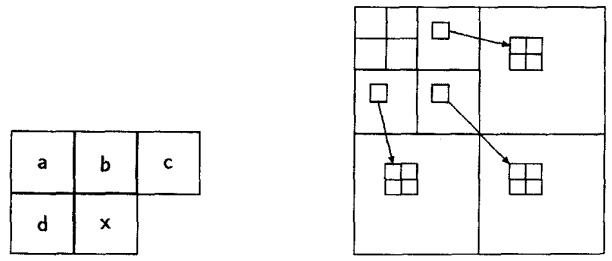
a slightly more complex integer representation, which is defined in Table IV. With this representation, the number of a magnitude set (MS) is entropy coded first, and, depending on its value, it is followed by the sign bit and the magnitude-difference bits. For example, the numbers 15 and −16 are transmitted with the sequences (7, +, 3) and (8, −, 0), respectively.

We used two well-known entropy coding methods in our coding tests: the Huffman and the arithmetic. The first is preferred when the hardware resources are limited and coding/decoding speed is a prime objective [1]. Arithmetic coding is somewhat slower than Huffman, but it is much more versatile and effective. It can be easily made adaptive [15], [16] and also exploit high-order dependencies with the use of conditioning contexts [5], [18]. We used the adaptive arithmetic coding program of Witten *et al.* because its source code is listed in [15].

We propose the three following coding methods that are used to evaluate the true compression rates that can be achieved with the S+P-transform. These methods are very simple, yet remarkably efficient.

Method 1: Here we use an entropy-coding method quite similar to the one used in JPEG's standard for lossless compression [3]. The main differences are 1) the S+P-transform replaces differential pulse code modulation (DPCM) and 2) a Huffman code optimized for each image (two-pass coding) replaces JPEG's fixed Huffman code (one-pass).

Method 2: In this method, the MS-VLI representation of Table IV is used, with the magnitude set (MS) information being arithmetic coded conditioned to the mean MS value of the adjacent pixels. For example, the MS of pixel x in Fig. 3(a) is coded conditioned to the mean MS of the pixels a, b, c, and d. Since the average MS is a growing function of the variance, with this method the encoder can detect the activity (or "local variance") in that region of the transformed image, and code the MS symbol accordingly. This dependence between magnitudes occurs not only locally, but is also observed between pixels in different levels of the hierarchical pyramid.



(a) (b)

Fig. 3. Set of conditioning contexts used by the arithmetic encoder in coding method 2. (a) Adjacent pixels. (b) Parent pixels.

For that reason, the MS value of the pixel in the same spatial orientation, and in the next level of the pyramid, as shown in Fig. 3(b), is also used as a coding context. More formally, let $\mathrm{MS}(i, j)$ be the magnitude-set of pixel (i, j). $\mathrm{MS}(i, j)$ is coded conditioned to the pair $(\overline{\mathrm{MS}}, \mathrm{MS}_p)$, where $\overline{\mathrm{MS}}$ is the mean MS of the adjacent pixels rounded to the next integer, and $\mathrm{MS}_p \equiv \mathrm{MS}(\lfloor i/2 \rfloor, \lfloor j/2 \rfloor)$ is the MS of the "parent" pixel in the pyramid structure. In practice, this conditional coding was done by selecting, for each coded MS, an adaptive model [15] numbered $K = \min(4, \overline{\mathrm{MS}}) + 5 \min(4, \mathrm{MS}_p)$. This limits the number of adaptive models (conditioning contexts) to 25, with each model containing a number of symbols equal to the number of magnitude sets present in that image (typically 16 for 8-b images).

The method above is similar to the use of "buckets" for complexity reduction [18]. However, we went one step further: The sign and magnitude difference were also arithmetic coded, but to increase the efficiency we used a fixed (instead of adaptive) uniform-distribution model, and we also used the fact that the number of symbols inside each set of Table IV is a power of two to substitute some products by bit shifts. This is equivalent to the uncoded transmission of the sign and magnitude-difference bits. This modification yields a 25–35% reduction in the coding/decoding times.

Method 3: This is the method of set partitioning in hierarchical trees (SPIHT), introduced in [22] for progressive-fidelity transmission, and here used with the S+P-transform. Its adaptation for lossless compression is described in Section V. Its bit rates required for perfect reconstruction are shown here to facilitate the comparisons.

The coding results are shown in Table V. All the bit rates presented in that table are calculated from the size of the compressed files, including a small header, and are not entropy estimates. In all tests, the S+P-transform pyramids have six levels. The predictor with smallest entropy was selected for each image, with predictor C used for the medical images CT 1–4 and XR 1, and predictor B used for the other images. For a comparison with other coding methods, Table V also shows the rates presented in [6], [17], and [20].

We can see that the best progressive-resolution results are obtained with Method 2. For reference, the second column in Table V shows the first-order entropy, which is the lower bound for compression methods that do not exploit higher order dependencies, of the S+P-transformed images. Note that

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

TABLE V
LOSSLESS COMPRESSION RATES (B/PIXEL) OBTAINED WITH S+P-TRANSFORM COMBINED WITH THE PROPOSED CODING METHODS (1–3), AND METHODS PROPOSED BY TISCHER *ET AL.* [17], TAKAMURA AND TAKAGI [20], AND EFFRIS *ET AL.* [6]. THE ENTROPY CORRESPONDS TO THE S+P-TRANSFORMED IMAGE

image	1st order entropy	coding method			reference		
		I	II	III	[17]	[20]	[6]
Girl	4.64	4.64	4.53	4.56	—	4.61	—
Couple	4.16	4.27	3.87	3.93	—	4.06	—
Lena	4.33	4.47	4.17	4.19	4.51	—	—
Mandrill	5.91	6.04	5.77	5.79	6.21	—	—
CT 1	3.37	3.47	2.70	2.89	—	—	—
CT 2	5.67	5.82	5.14	5.20	—	—	—
CT 3	4.80	4.94	4.24	4.35	—	—	—
CT 4	4.92	5.05	4.30	4.43	—	—	—
XR 1	4.08	4.24	3.85	3.94	—	—	6.37
XR 2	4.10	4.29	3.96	3.96	—	—	5.73

it can code, in some cases, to rates more than 0.5 b/pixel below the first-order entropy, which is unusual for lossless compression, when the gains are usually modest. On the other hand, the use of arithmetic coding makes it slower than Method 1.

A comparative evaluation of coding methods is particularly difficult for lossless compression of medical images because many published results were obtained using the author's own images. There are also results based on entropy estimates that turned out to be over-optimistic and were later corrected by the authors.

In Table V, the results by Tischer *et al.* [17] were obtained using the JPEG predictive coding scheme, and arithmetic coded with the Q-coder [16]. They used an extension of the method proposed by Todd *et al.* [18]. These results correspond to the best compression rates with 256×256 binary contexts. Similar results were obtained by Rabbani *et al.* [1], [5]. Takamura and Takagi [20] used a lossy-plus-residual method, which allows fast inspection; their lossy image was obtained with JPEG's lossy (discrete-cosine-transform-based) method, and the residual was compressed using multiple adaptive predictors and context-based arithmetic coding. It is worth mentioning that the compression efficiency of their method decreases as the quality of the lossy reproduction increases. Thus, their best results (Table V) were obtained with very low JPEG "quality values." Better lossy reproductions require 0.2–0.5 b/pixel increase in their rates. From this comparative analysis we can conclude that the results obtained with the S+P-transform and the proposed coding methods are among the best known, while simultaneously having low complexity and allowing full use of progressive transmission.

V. PROGRESSIVE-FIDELITY TRANSMISSION

For progressive-fidelity transmission we use the method of set partitioning in hierarchical trees (SPIHT) [22], which is in principle similar to the embedded zerotree wavelet (EZW) algorithm [21]. SPIHT is presently one of the most efficient methods known for lossy compression, both in terms of speed and compression. In addition, it has several other advantages: It is completely adaptive; it is simple to implement; and it

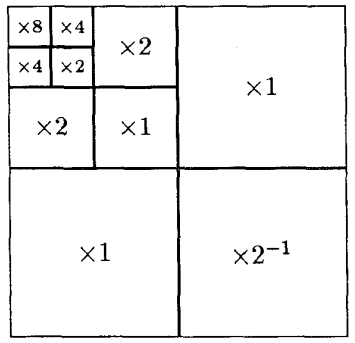


Fig. 4. Factors to be used in the implicit scaling of the S+P-transform pyramid to approximate a unitary transformation.

produces a fully embedded message, i.e., a message corresponding to a rate R_0 bits always forms the first R_0 bits of any message with rate $R_1 \geq R_0$. With embedded coding, at any point in the decoding process it is possible to recover the lossy version with distortion corresponding to the rate of the received message, which allows coding/decoding to exactly the desired rate or distortion.

One important property of the SPIHT algorithm is that it codes information corresponding to individual bits of the transformed image, following a bit-plane ordering. Thus, it shares some characteristics of the well-known bit-plane binary coding methods, and can be used for lossless compression. However, it also has quite distinct characteristics: The bits are not transmitted in the usual line-by-line order; and tree structures are used in such a way that, with a single coded symbol, the decoder can infer that the all bits in a large region of a given bit plane are zero.

The coding efficiency of the SPIHT algorithm comes from exploiting the self-similarity present in the wavelet multiresolution representation—a property also present in the S+P-transformed images. The only reason the S+P-transform cannot be used directly with SPIHT is that, for embedded lossy compression, the transmission priority given by the bit planes will minimize the mean squared error (MSE) distortion only when the transformation is unitary. The S+P-transform is not unitary, but we can get a good approximation considering that, if we had used

$$l[n] = \frac{c[2n] + c[2n+1]}{\sqrt{2}}$$
$$h[n] = \frac{c[2n] - c[2n+1]}{\sqrt{2}} \tag{13}$$

instead of (1), we would have a unitary transformation. Combining the corrective factors in the 2-D transformation, we conclude that the S-transform would be approximately unitary if multiplied by the factors shown in Fig. 4. Since the S-transform coefficients are integers and the scaling factors are powers of two, they can be considered implicitly while coding. Due to these advantages, we used the same scaling factors with the S+P-transform.

In the progressive-fidelity transmission scheme the decoder initially sets the transformed image to zero and updates its

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

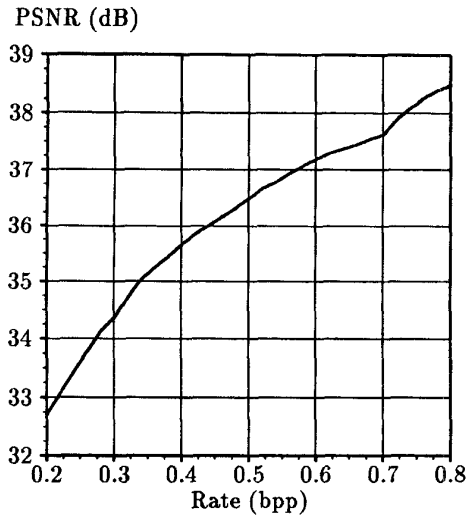


Fig. 5. Performance of coding method 3 for progressive-fidelity transmission. Image: Lena 512×512.

pixel values using the coded message. The decoder can decide at which rate to stop, and then it calculates the inverse S+P-transform to obtain a lossy version of the image. If it continues decoding to the end of the file, the image is recovered exactly. Thus, all SPIHT results presented here were obtained from the same file.

The SPIHT algorithm can be used to code all bit planes to recover the image exactly. However, when it codes the least significant bits, its efficiency decreases, mostly in terms of speed and memory usage. This usually happens for a bit rate when the lossy version of the image is visually indistinguishable from the original, and for that reason we have chosen to use a hybrid method: SPIHT is used to code up to the third least significant bit, and then we use a simplified version of Method 2, described in Section IV, to code the remaining bit planes. A description of the SPIHT algorithm that we used can be found in [22]. The main difference is that here the SPIHT method was changed to disregard the parts of the bit planes that, due to scaling, are identically zero.

The progressive-fidelity transmission results obtained with the Lena image are shown in Fig. 5. The PSNR is defined by

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{255^2}{\mathrm{MSE}} \right) \mathrm{dB}. \tag{14}$$

These rate-versus-PSNR results are excellent, considering the speed of the S+P-transform and decoding algorithm, and the possibility of exact recovery. They are slightly inferior (usually less than 1 dB) to those obtained with the SPIHT algorithm [22] on a wavelet transform, and are practically equal or superior to the EZW method [21] and other much more complex coding methods like subband coding with adaptive vector quantization [23].

The rates for perfect reconstruction are shown in Table V, where the method above is identified as Method 3. Note that these rates are near those obtained with Method 2 for progressive-resolution transmission; i.e., progressive fidelity is achieved with a very small loss in compression, in contrast to the penalty incurred by the lossless tree-structured vector quantization [6, Table V].

(a)

(b)

Fig. 6. Lossy reproduction of the Lena image obtained with coding method 3, for progressive-fidelity transmission. (a) Original. (b) 0.2 b/pixel, PSNR = 32.7 dB.

Fig. 6 shows the lossy version of the Lena image coded with this method at rate 0.2 b/pixel. Like subband-coded images, there are no blocking artifacts and, even though a bit-plane approach was used, the inverse S+P-transformation completely eliminates the "contouring" artifacts usually present in bit-plane coded images.

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.

VI. Summary and Conclusions

We have proposed a new image multiresolution transformation, called the S+P-transform, which is suited for both lossy and lossless compression. It is shown that it can be computed with a small computational effort, using only integer additions and bit shifts. Numerical results show that the entropy of the transformed images is smaller than that obtained with predictive coding methods of similar complexity. In addition, we proposed coding methods that exploit the multiresolution representation for efficient progressive transmission. The methods for progressive-resolution transmission have low complexity, and still can compress the images to rates among the best in the literature. An embedded-coding method was proposed for progressive-fidelity transmission, and it is shown that it yields a rate-versus-distortion curve superior to much more sophisticated vector quantization methods and inferior only to the most efficient lossy compression methods employing wavelet transforms. At the same time, its rates to code the image for lossless recovery are very near those obtained with the progressive-resolution methods.

Thus, we have shown that, with the proper multiresolution representation, it is possible to have compression schemes allowing efficient and fast inspection of the images—in a reduced resolution or in a lossy reconstruction—and simultaneously code the images with performance comparable to the best-known schemes for lossy or lossless compression.

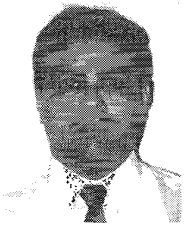
The codec programs with the methods proposed in this paper, including the S+P-transform, can be obtained via anonymous ftp from the host `ipl.rpi.edu`, directory `pub/EW_Code`, with instructions in the file `README` or via the Internet site with URL `http://ipl.rpi.edu/SPIHT`.

Acknowledgment

The authors wish to thank M. Effros for the X-ray medical images and the reviewers for their suggestions.

References

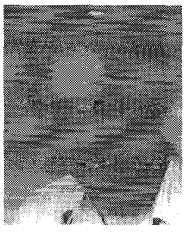
[1] M. Rabbani and P. W. Jones, *Digital Image Compression Techniques.* Bellingham, WA: SPIE, 1991.
[2] G. R. Kuduvalli and R. M. Rangayyan, "Performance analysis of reversible image compression techniques for high-resolution digital teleradiology," *IEEE Trans. Med. Imaging,* vol. 11, pp. 430–445, Sept. 1992.
[3] G. K. Wallace, "The JPEG still picture compression standard," *Comm. ACM,* vol. 34, pp. 30–44, Apr. 1991.
[4] CCITT Draft Recommendation T.82, ISO/IEC Committee Draft 11544, "Progressive bi-level image compression," Sept. 1991.
[5] M. Rabbani and P. W. Melnychuck, "Conditioning contexts for the arithmetic coding of bit planes," *IEEE Trans. Signal Processing,* vol. 40, pp. 232–236, Jan. 1992.
[6] M. Effros, P. A. Chou, E. A. Riskin, and R. M. Gray, "A progressive universal noiseless coder," *IEEE Trans. Inform. Theory,* vol. 40, pp. 108–117, Jan. 1994.
[7] J. W. Woods, Ed., *Subband Image Coding.* Boston, MA: Kluwer, 1991.
[8] Special issue on wavelets and signal processing, *IEEE Trans. Signal Processing,* vol. 41, Dec. 1993.
[9] V. K. Heer and H.-E. Reinfelder, "A comparison of reversible methods for data compression," in *Proc. SPIE–Med. Imaging IV,* 1990. vol. 1233, pp. 354–365.
[10] E. H. Adelson, E. Simoncelli, and R. Hingorani, "Orthogonal pyramid transforms for image coding," in *Proc. SPIE,* Cambridge, MA, Oct. 1987, vol. 845, pp. 50–58.
[11] S. Rănganath and H. Blume, "Hierarchical image decomposition and filtering using the S-transform," in *Proc. SPIE–Med. Imaging II,* 1988, vol. 914, pp. 799–814.
[12] P. Roos, A. Viergever, and M. C. A. van Dijke, "Reversible intraframe compression of medical images," *IEEE Trans. Med. Imaging,* vol. 7, pp. 328–336, Sept. 1988.
[13] W. H. Press, B. P Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Programming.* Cambridge, UK: Cambridge Univ. Press, 1986.
[14] A. Said and W. A. Pearlman, "Reversible image compression via multiresolution representation and predictive coding," in *Proc. SPIE,* Nov. 1993, vol. 2094, pp. 664–674.
[15] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM,* vol. 30, pp. 520–540, June 1987.
[16] W. B. Pennebaker, J. L. Mitchell, G. G. Langdon, Jr., and R. B. Arps, "An overview of the basic principles of the Q-coder adaptive binary arithmetic coder," *IBM J. Res. Develop.,* vol. 32, pp. 717–726, Nov. 1988.
[17] P. E. Tischer, R. T. Worley, A. J. Maeder, and M. Goodwin, "Context-based lossless image compression," *Comp. J.,* vol. 36, pp. 68–77, Jan. 1993.
[18] S. Todd, G. G. Langdon, Jr., and J. Rissanen, "Parameter reduction and context selection for compression of grey-scale images," *IBM J. Res. Develop.,* vol. 29, pp. 188–193, 1985.
[19] K. Sayood and K. Anderson, "A differential lossless image compression scheme," *IEEE Trans. Signal Processing,* vol. 40, pp. 236–241, Jan. 1992.
[20] S. Takamura and M. Takagi, "Lossless image compression with lossy image using adaptive prediction and arithmetic coding," in *Proc. Data Compress. Conf.,* Snowbird, UT, Mar. 1994, pp. 166–174.
[21] J. M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients," *IEEE Trans. Signal Processing,* vol. 41, pp. 3445–3462, Dec. 1993.
[22] 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 Tech.,* vol. 6, pp. 243–250, June 1996.
[23] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing,* vol. 1, pp. 31–48, Jan. 1992.



Amir Said (S'90–M'93–S'93–M'95) received the B.S. and M.S. degrees in electrical engineering from the State University of Campinas, Brazil, in 1985 and 1988, respectively. In 1994, he received the Ph.D. degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY.

After a year at IBM Brazil, he joined the School of Electrical Engineering of the State University of Campinas, Brazil, as a lecturer. In 1994, he became an Assistant Professor. His research interests include image compression, channel coding and modulation, signal processing, and optimization.

While at Rensselaer Polytechnic Institute, Dr. Said was awarded the Allen B. DuMont prize for academic achievement.



William A. Pearlman (S'61–M'64–S'71–M'74–SM'84) received the B.S. degree and M.S. degree in electrical engineering from Massachusetts Institute of Technology, Cambridge, MA, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1974.

Between 1963 and 1974, he was employed by Lockheed Missiles and Space Company, Sunnyvale, CA, and by GTE-Sylvania, Mountain View, CA. He then joined the Department of Electrical and Computer Engineering, University of Madison, WI, as an Assistant Professor and, in 1979, the Electrical, Computer, and Systems Engineering Department at Rensselaer Polytechnic Institute, where he is currently a Professor. His research interests are in information theory; image, video, and audio coding; and image processing.

Dr. Pearlman has served as conference chairman of SPIE's Visual Communications and Image Processing Conference in 1989, and as associate editor of coding for IEEE Transactions on Image Processing. He is a fellow of SPIE.

Authorized licensed use limited to: University of Illinois at Chicago Library. Downloaded on June 06,2024 at 10:59:51 UTC from IEEE Xplore. Restrictions apply.