A Lossless Version of the Hadamard Transform

Wilfried Philips and Koen Denecker
University of Gent, Department of Electronics and Information Systems (ELIS),
Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
philips@elis.rug.ac.be denecker@elis.rug.ac.be

Abstract— **This paper presents a lossless block-transform coder based on the Hadamard transform that achieves a compression ratio as good as that of the current lossless JPEG-standard and about 10 to 20% lower than the currently best state-of-the-art technique Calic. The main advantage of the proposed technique is that it produces an embedded data stream which may be truncated to achieve lossy compression. Preliminary results obtained on the "lena" image show that the proposed scheme produces an image quality that is comparable to that of lossy JPEG for compression ratios up to 4, and only slightly worse for compression ratios in the range 4 to 10. This means that the proposed coder performs well over a wide range of compression ratios and is therefore a useful alternative for the lossy and lossless JPEG-standards in nearly-lossless coding or when lossless and lossy decoding must be supported simultaneously. Finally, the transform part of the proposed scheme requires fewer computations than lossy JPEG (and only slightly more than lossless JPEG).**

Keywords— **Lossless and quasi-lossless image coding, Hadamard transform.**

I. Introduction

MOST lossless compression schemes are based on prediction techniques, or simple variants thereof. Many lossless schemes share the same principle: they compute a linear estimate to predict (final or intermediate) pixel values and round the estimate to the nearest integer to achieve losslessness. Simple examples of such quasi-linear "predictive" techniques include the S+P transform [1], of which many variants have been published under different names, e.g., the "lossless integer wavelet transform" [2] and the modified Haar transform [3]. Interpolation based wavelet techniques [4], [5] are less trivial examples of such quasi-linear predictive techniques. Finally, the current state-of-the-art lossless compression techniques such as Calic [6], Loco-I [7] and Btpc [8] are based on more complicated (non-linear) predictive techniques.

In contrast, most lossy image coding schemes (e.g., lossy JPEG [9], wavelet coders, ...) are not based on prediction but rather on general transforms such as the DCT, the wavelet transform, ...; they apply a linear transform to the input data and then quantize the resulting coefficients. In theory, these techniques can be made lossless by sufficiently finely quantizing the coefficients, or by losslessly coding the error image, but unfortunately both of these strategies result in a negligibly small compression ratio (of the order of 1, or even less). Moreover, fine quantization of the coefficients may work well on one platform, but is difficult to guarantee lossless reconstruction over a variety of different platforms. Therefore it is not surprising that lossless coding schemes are hardly ever based on non-trivial linear transforms, but rather on predictive techniques or simple variants thereof.

One noteworthy exception (i.e., an example of a lossless coder based on a non-trivial transform) is the recently introduced rounding transform [10], a technique which is able to losslessly recover the input data from the quantized transform coefficients for transforms that satisfy certain criteria. In theory, the class of "rounding" transforms is fairly general, but the results published so far are restricted to some very simple transforms, which are similar in nature to the predictive schemes discussed above.

In section II, we present a block-based transform coder based on the Hadamard transform and a sophisticated coefficient quantization/dequantization scheme. The scheme cannot be interpreted in terms of simple prediction and rounding operators. The quantized coefficient data is organized into an embedded data stream which may be decoded completely to achieve lossless compression but which may also be truncated to achieve a much higher compression ratio, at the expense of a lossy reconstruction. In the latter case, the coefficients reconstructed from the truncated data stream approximately equal quantized version of the coefficients of the traditional lossy Hadamard transform [11]. Therefore, in lossy mode the coding properties of the proposed scheme are similar to those of the lossy Hadamard transform.

The experimental results in section III show that in lossless mode the proposed scheme achieves a compression ratio which is as good or even better (depending on the mode of operation) than the current lossless JPEG-standard. The achieved compression ratio is still 10 to 20% lower than the currently best state-of-the-art technique Calic [6], but the proposed technique has not been optimized to the same degree as Calic and has the advantage of producing an embedded data stream. Predictive techniques such as Calic, Loco-I and Btpc can also operate in lossy mode, but the proposed scheme is based on transform coding and it is known that transform coders produce much better images than lossy predictive techniques at higher compression ratios [11]. In fact, the experimental results in section III show that on images of size 512×512, the compressed image quality produced by the proposed scheme is indistinguishable of that of lossy JPEG for compression ratios up to 4, and is only slightly worse for compression ratios in the range 4 to 10.

In conclusion, the proposed coder performs well over a wide range of compression ratios and is therefore a useful alternative for the lossy and lossless JPEG-standards when lossless and lossy decoding must be supported simultaneously. Furthermore, the computational complexity of the transform part of the proposed scheme (no multiplications needed, less than 10 additions per pixel and some bit shift operations) is quite low.

A comparable quantization scheme has also been applied to a colour-decorrelation transform for contone CMYK images for pre-press applications [12], [13]. Using the Karhunen-Loève transform, an optimal 4×4 colour transform matrix is determined for every square block and decomposed into a product of an upper-triangular matrix, a diagonal matrix and a lower-triangular matrix. Careful application of the quantization schemes allows lossless reconstruction.

II. Lossless Hadamard coding

A. The principle of lossless coding

In this section, we present the principle of lossless Hadamard coding and show how lossless reconstruction is performed on one-dimensional, integer valued signals. The extension to two-dimensional signals is straightforward: it can be done through a combination of a lossless transformation on the rows and on the columns of a square block or by rearranging a square block of $N \times N$ into a vector of size N^2 and losslessly transforming this vector. In this paper, we use the former approach.

Firstly, the signal is split up into blocks of length $n = 2^k$; in our experiments, n will be 8. Let H be the Hadamard matrix [14, p. 209] of order n. In the following, $\mathbf{x}$ denotes a column vector of dimension n and x_j its j-th component; Z_n is the space of all $\mathbf{x}$ with n integer components; finally, A is an $n \times n$ matrix and $A_{i,j}$ one of its components. In this paper we only consider the case $n = 8$, which is most appropriate for image coding; however, all of the derivations in this section can be repeated for any other value of n, with similar results.

In transform coding it is most useful to order the base vectors, i.e., the rows of H in order of increasing sequency:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \qquad (1)$$

Typical highly correlated input (column) vectors $\mathbf{x}$ are then transformed into coefficient vectors $\mathbf{a} = H\mathbf{x}$ which tend to be arranged in order of decreasing magnitude. The dynamic range of the output will be 8 times larger than the dynamic range of the input.

We assume that each of the coefficients a_j is quantized by rounding it to an integer multiple $k_j\Delta_j$ of a quantization step Δ_j. Specifically, we assume that k_j is the largest integer for which $k_j\Delta_j \leq a_j$ (similar results hold for other rounding schemes, but lead to more complicated notations). We will denote this quantization operator [.] and therefore $\mathbf{k} = [\mathbf{a}]$.

We will now determine a sufficient condition for the input vector $\mathbf{x}$ to be unambiguously recoverable from its quantized coefficient vector $[\mathbf{a}]$. The input vector will be recoverable iff. H and [.] satisfy:

$$[Hx_1] \neq [Hx_2],\ \forall x_1, x_2 \in \mathrm{Z}_n \text{ with } x_1 \neq x_2. \qquad (2)$$

If [.] and H satisfy this condition, we will say that the operator $[H.]$ is invertible.

To derive the condition, we note that for all $n = 2^k$, H can be decomposed as $H = AB$, where A and B are integer square matrices, with B unimodular (i.e., $|\det(B)| = 1$) and with A triangular. This decomposition is not unique, but a particularly interesting one is the following:

$$A = \begin{bmatrix} 8 & 4 & 4 & 4 & 2 & 0 & 2 & 1 \\ 0 & -4 & 0 & 4 & 2 & 0 & 0 & 1 \\ 0 & 0 & 4 & 4 & 2 & -2 & 2 & 1 \\ 0 & 0 & 0 & -4 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 2 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -2 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 1 & 0 & -1 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}. \tag{4}$$

Now, since $|\det(B)| = 1$, B^{-1} is also an integer matrix (this follows from the explicit matrix inversion formula). This means that for all integer vectors $\mathbf{y}$, there exists an integer vector $\mathbf{x}$ such that $\mathbf{y} = B\mathbf{x}$. Therefore, $[H.]$ is invertible iff. $[A.]$ is invertible. Because A is triangular, the latter condition is equivalent to $\Delta_i \leq |A_{i,i}|$, for $i = 1, \ldots n$ (this can be shown by induction on j, in the order $j = n, n-1, \ldots, 1$).

We now assume that $\Delta_j = |A_{j,j}|$: indeed, there is no point in using quantization steps which are smaller than necessary, because this would increase both the magnitude and hence the entropy of the integer coefficients k_j. The net effect of this quantization is that the $\log_2(\Delta_j)$ least significant bits can be omitted from the transformed coefficients. Compression will benefit a lot from this step since these bits tend to be very noisy.

Since A is an integer matrix, Δ_j is an integer and a_j may be expanded as $a_j = k_j \Delta_j + a'_j$, where $a'_j \equiv a_j \pmod{\Delta_j}$. Dequantizing is then equivalent to finding the numbers a'_j.

Because $\mathbf{a} = A\mathbf{y}$, where $\mathbf{y}$ is an integer vector and A a fixed upper-triangular matrix, and because $|A_{j,j}| = \Delta_j$, we have

$$\begin{aligned} a'_j &\equiv a_j \pmod{\Delta_j} \\ &\equiv \sum_{l=j}^{n} A_{j,l}\, y_l \pmod{\Delta_j} \\ &\equiv \sum_{l=j+1}^{n} A_{j,l}\, y_l \pmod{\Delta_j}. \end{aligned} \tag{5}$$

This shows that a'_j depends only on y_l, $l = j+1, \ldots, n$. Furthermore, by inverting the set of equations $\mathbf{a} = A\mathbf{y}$, it can be seen that y_l depends only on a_m, $m = l, \ldots, n$. By combining these facts, it becomes clear that a'_j is a function of $a_l \pmod{\Delta_j}$, $l = j+1, \ldots, n$. In conclusion, the numbers a'_j can be determined recursively in order of decreasing j, starting with $a'_n = 0$ (because $|A_{n,n}| = 1$). In fact, the following equations may be used:

$$\begin{aligned} a_8 &= k_8 \\ a_7 &= 2k_7 + a_8 \pmod 2 \\ a_6 &= 2k_6 + a_7 \pmod 2 \\ a_5 &= 2k_5 + (a_8 + a_7 - a_6) \pmod 2 \\ a_4 &= 4k_4 + (a_8 + a_7 - a_6) \pmod 4 \\ a_3 &= 4k_3 + (a_6 + a_5 - a_4) \pmod 4 \\ a_2 &= 4k_2 + (a_8 + a_5 - a_4) \pmod 4 \\ a_1 &= 8k_1 + (a_8 + a_7 - a_6 + a_5 \\ &\quad -a_4 + a_3 - a_2) \pmod 8. \end{aligned} \tag{6}$$

From an implementational point of view, it is worth noting that the multiplications may be implemented as bit shifts and that the remaining computations involve simple operations on the least significant bits of the coefficients a_l, which could even be implemented using lookup tables, if desired.

B. Theoretical performance

A useful performance measure of different quantization schemes on Hadamard transformed data is $\mathcal{P} = \log_2\left(\prod_{j=1}^{n} \Delta_j\right)$. Indeed $\mathcal{P}$ is a (very) rough estimate[1] of the number of bits that are saved by quantizing the coefficients $\mathbf{a}$. For the scheme proposed in this paper $\mathcal{P} = n\log_2(n)/2$, i.e., $\mathcal{P} = 12$ for $n = 8$. On the other hand, with classical quantization/dequantization schemes, the Hadamard transform can be lossless only when $\Delta_i = 1$, $i = 1, \ldots, 8$. Indeed, such classical schemes reconstruct $\mathbf{x}$ by simply inversely transforming scaled versions $k_j\Delta_j$ of the quantized coefficients and rounding the result. In order for such schemes to be lossless, Δ_j must not exceed 1, hence $\mathcal{P} = 0$. Therefore, for $n = 8$ the scheme in this paper theoretically requires about $12/8 = 1.5$ fewer bits per coefficient than classical schemes. For the coding of two-dimensional signals, we expect that the gain in bit rate will be smaller than twice the gain in one dimension because the signal will already be partially decorrelated after one step (see sect. III for an example).

[1] The estimate would be exact if the least significant bits of the coefficients a_j were completely random.

Note that there are many possible ways to factor H into a triangular and a unimodular matrix. Each of these factorizations yields a different quantization/dequantization scheme. However, all these schemes have the same performance $\mathcal{P} = n\log_2(n)/2$: indeed, for all of them $\mathcal{P} = \det A$ on the one hand, and $\det A = \det H = n^{n/2}$ on the other.

The particular scheme in this paper has an important advantage over others: it strips off more of the least significant bits from low order coefficients than it does from high order coefficients. As low order unquantized coefficients tend to have a larger dynamic range than high order coefficients, the scheme tends to equalize the dynamic range of the quantized coefficients of different orders. Typically, this results in quantized coefficients which can all be represented using 8 bits when the original data has a resolution of 8 bits.[2] This is an important advantage because the quantized coefficients are entropy coded in practice and because the complexity of entropy coders tends to grow with the dynamic range of the coefficient data.

C. The image codec

The currently implemented image coder operates on blocks of 8×8 pixels. The columns of each block are transformed and rounded as explained in section II-A. The resulting quantized coefficient vectors $\mathbf{k}$ are grouped into an intermediate coefficient block and the rows of this block are then processed in a similar fashion. This results in a final coefficient block for each input block.

The final coefficient blocks are then encoded using an adaptive arithmetic coder in several passes; in each of these passes one bit of the magnitude (absolute value) of each coefficient is encoded, starting with the most significant bits. The sign bit of a coefficient is encoded along with the first non-zero magnitude bit of this coefficient. When encoding a bit of a coefficient, all of the previously encoded bits of these coefficients are used as a context. The arithmetic coder is inspired by the ideas in [15].

The data stream produced by the encoder is clearly embedded: if it is truncated only information about some least significant bits of the coefficients is lost.

III. Experimental results

The proposed lossless Hadamard transform was applied to three gray scale images (8 bit per pixel), which were subsequently decoded to verify its losslessness. Table I shows that the compression ratios for the proposed scheme are very close to those of lossless JPEG (Ljpg) and about 10 to 20% lower than those of Calic, which is the best lossless technique we know of. The table also shows that the compression ratio of the proposed scheme exceeds that of Ljpg, when the Hadamard transform is preceded by an additional prediction step. However, the extra prediction step probably limits the usefulness of the proposed scheme for progressive decoding (as it does in e.g., Calic and Ljpg); therefore, we omit the prediction step in the remainder of this paper.

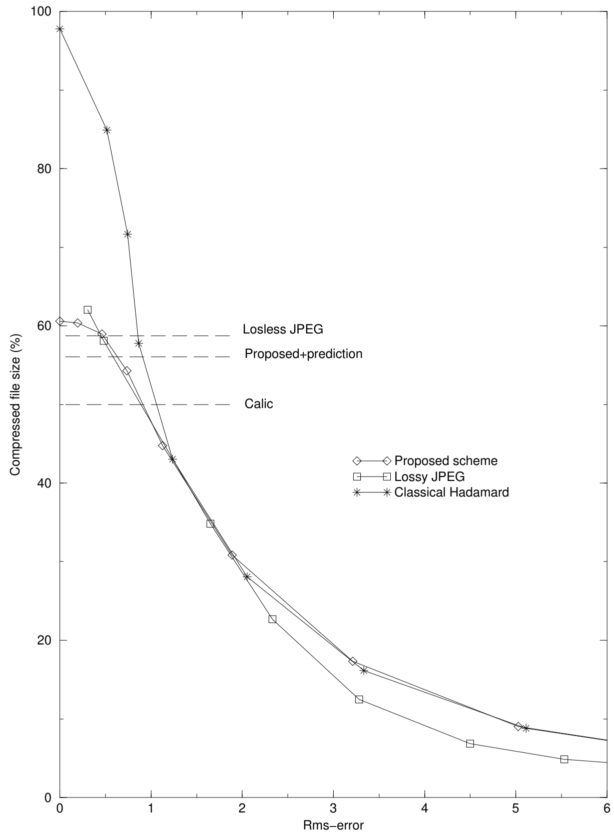

Fig. 1. Rate-distortion curves for three compression methods obtained on the "lena" image. The dashed lines indicate the compressed file size for three lossless compression methods.

Fig. 1 displays the rate-distortion curves for the "lena" image[3] of lossy JPEG, the classical lossy Hadamard scheme, and the proposed scheme (note: the same arithmetic coder was used in both Hadamard schemes). For compression ratios in the range 2 to 4, the Hadamard schemes perform as well as lossy JPEG

[2] In fact, it is easily verified that even in the worst case, k_1 requires no more than 8 bits, while k_2 and k_3 require no more than 9 bits.

[3] There exist different versions of this image; we used the one from `ipl.rpi.edu`, which is most difficult to code by any method.

TABLE I
Lossless Compression ratios for different images

Image	lena	mr_chest	angio_chest
Size	512x512	256x256	512x512
Proposed	1.66	1.52	2.28
Proposed + Pred.	1.78	1.66	2.48
Ljpg	1.70	1.56	2.29
Calic	1.95	1.80	2.60
Btpc	1.80	1.60	2.53
Loco-I	1.88	1.73	2.54

from an root mean squared error (Rmse) point of view. For higher compression ratios, lossless JPEG produces a lower Rmse. However, for compression ratios up to 10, the difference in *subjective* image quality between all of these techniques remains small: at an Rmse of 5, we found that JPEG produced slightly more detailed images than the proposed scheme, but that block distortion was more disturbing in the background regions of the JPEG image. The images are not included as the differences are invisible on paper print outs.

Note that JPEG cannot achieve lossless coding (the Rmse is non-zero, even when the quality factor is set to its highest possible value). The classical Hadamard scheme can achieve lossless operation, but only at the expense of no compression (compression factor 1.02). The proposed scheme achieves lossless operation using about 3 bits fewer per pixel and also performs better in lossy mode. The savings of 3 bits is almost double the theoretically expected 1.5 bits. This can probably be explained by the fact that the coded coefficients are obtained after *two* instead of one transformations. Each quantization step (one in the column transform step and one in the row transform step) reduces the dynamic range of the intermediate coefficients and saves about 1.5 bit per intermediate coefficient.

Fig. 2 shows a comparison of the rate-distortion curves of the proposed technique combined with curves of three state-of-the-art lossy image coders: Jpeg, Loco-I and Btpc. From this figure, we see that the smallest errors occur for the Loco-I technique in the compression ratio range 2 to 4 (but Loco-I does not provide any means for an embedded bitstream). For higher compression ratios, the Btpc yields the smallest root mean square error, while Loco-I completely fails to keep the quality of the image at an acceptable level. In this range Btpc (a combined lossless/lossy multi-resolution technique) also consistently outperforms Jpeg. From an Rmse point of view, the proposed technique still performs worse than Btpc and JPEG. However, as mentioned before, the differences with JPEG are small (and even debatable from a subjective image quality point of view). Furthermore, techniques such as Btpc have been highly optimized, whereas the proposed technique has not.

In principle, Calic (the state-of-the-art technique in lossless image compression) can also be operated in lossy (or rather near-lossless) mode. Unfortunately, the implementation made available by the authors of Calic, has no such possibility so we were not able to test this mode of operation. However, it is to be expected that the image quality of Calic (being a predictive technique) will drop below that of the proposed scheme at high compression ratios. This is because it is generally agreed that transform coders outperform predictive schemes at high compression ratios [11].

IV. Conclusion

The compression scheme presented in this paper clearly demonstrates the feasibility of lossless coding schemes based on non-trivial orthogonal transforms. The proposed Hadamard coder achieves efficient coding in both lossy and lossless operation and may be useful in progressive transmission.

Acknowledgments

This work was financially supported by the Belgian Fund for Scientific Research (FWO) through mandates of "postdoctoral research fellow" and "research assistant" and through the projects 39.0051.93 and 31.5831.95, and by the Flemish Institute for the Advancement of Scientific-Technological Research in Industry (IWT) trough the projects Tele-Visie (IWT 950202) and Samset (IWT 950204).

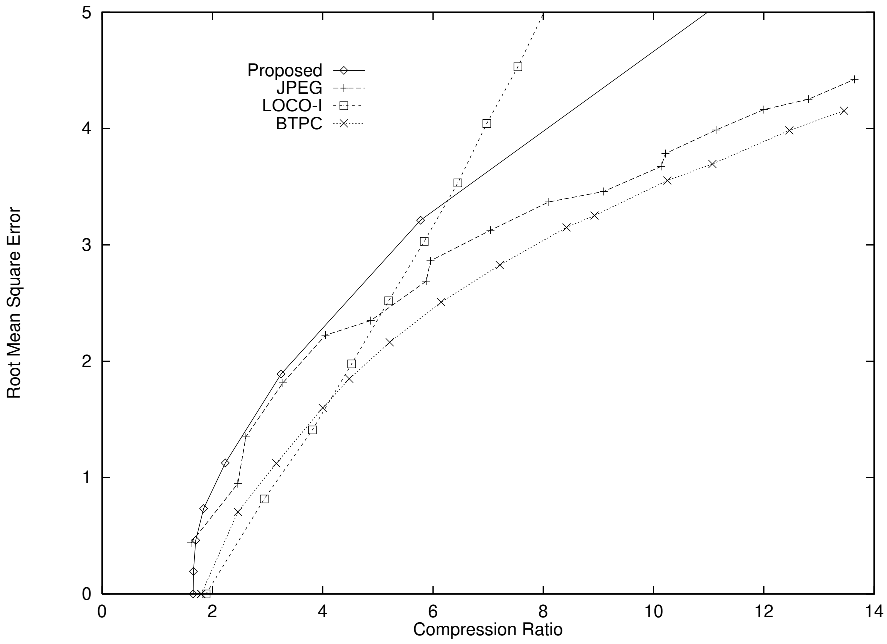

Fig. 2. Comparison of the rate-distortion curve for the proposed technique w.r.t. some other state-of-the-art compression techniques obtained on the "lena" image.

References

[1] A. Said and W. A. Pearlman, "Image compression via multiresolution representation and predictive coding," in *Visual Communications and Image Processing*, Nov. 1993, number 2094 in SPIE, pp. 664–674.

[2] S. Dewitte and J. Cornelis, "Lossless integer wavelet transform," *IEEE Signal Processing Letters*, vol. 4, no. 6, pp. 158–160, May 1997.

[3] P. Piscaglia and B. Macq, "Multiresolution lossless compression scheme," in *Proceedings of the IEEE International Conference on Image Processing (ICIP96)*, Lausanne, Sept. 1996, vol. 1, pp. 69–72.

[4] B. Aiazzi, P. Alba, and L. Alparone, "Three-dimensional lossless compression based on a separable generalized recursive interpolation," in *Proceedings of the IEEE International Conference on Image Processing (ICIP96)*, Lausanne (CH), Sept. 1996, vol. 1, pp. 85–88.

[5] G. Deng, "A new interpolative subband coding algorithm for lossless image compression," in *Proceedings of the IEEE International Conference on Image Processing (ICIP96)*, Lausanne, Sept. 1996, vol. 1, pp. 93–96.

[6] X. Wu, "Lossless compression of continuous-tone images via context selection, quantization, and modeling," *IEEE Transactions on Image Processing*, vol. 6, no. 5, pp. 656–664, May 1997.

[7] M. J. Weinberger, G. Seroussi, and G. Sapiro, "LOCO-I: A low complexity, context-based, lossless image compression algorithm," in *Proceedings of the IEEE Data Compression Conference*, J. Storer and M. Cohn, Eds. Mar. 1996, pp. 140–149, IEEE Computer Society Press.

[8] J. A. Robinson, "Efficient general-purpose image compression with binary tree predictive coding," *IEEE Transactions on Image Processing*, vol. 6, no. 4, pp. 601–608, Apr. 1997.

[9] International Telegraph and Telephone Consultative Committee (CCITT), *Digital Compression and Coding of Continuous-Tone Still Images.* Recommendation T.81, 1992.

[10] H.-Y. Jung, T.-Y. Choi, and R. Prost, "Rounding transform for lossless image coding," in *Proceedings of the IEEE International Conference on Image Processing (ICIP96)*, Lausanne, Sept. 1996, vol. 1, pp. 65–68.

[11] R. J. Clarke, *Transform Coding of Images*, Academic Press, London, 1985.

[12] S. Van Assche, W. Philips, and I. Lemahieu, "Lossless compression of pre-press images using a novel colour decorrelation technique," in *Image Analysis and Processing*, A. Del Bimbo, Ed., Florence, Italy, Sept. 1997, IAPR, vol. 2 of *Lecture Notes in Computer Science (1311)*, pp. 85–92, Springer.

[13] S. Van Assche, W. Philips, and I. Lemahieu, "Lossless compression of pre-press images using a novel colour decorrelation technique," *Pattern Recognition. Special Issue on Colour and Texture Analysis*, 1997, To be published.

[14] K. W. Pratt, *Digital Image Processing*, Wiley, New York, 1991.

[15] Y. Huang, H. Driezen, and N. Galatsanos, "Prioritized DCT for compression and progressive transmission of images," *IEEE Transactions on Image Processing*, vol. 1, pp. 477–487, Oct. 1992.