Merged Arithmetic for Computing Wavelet Transforms

Gwangwoo Choe and Earl E. Swartzlander, Jr.
choe@ece.utexas.edu
Electrical and Computer Engineering Department
The University of Texas at Austin
Austin, TX 78712

Abstract

A variation of merged arithmetic is applied to the implementation of the wavelet transform. This approach offers a simple design trade-off between the computational accuracy and the complexity. Our analysis shows that the trade-off is a function of the input data resolution, the number of filter taps, the arithmetic precision, and the level of the wavelet transform. The design parameter can be also fixed for a given number of taps and used to determine the minimum word size for the wavelet coefficients of the transform. The key element of this approach is to introduce a "truncation" within the merged arithmetic reduction process which provides equivalent throughput with a substantially less complexity. An experiment has been conducted to verify the analysis, which suggests that 24-bit merged arithmetic is required for the EZW algorithm to handle up to a level 6-wavelet transform.

1 Problem Statement

Compactly supported wavelets use a computational model that is a k-tap FIR filter operation. The theory of the wavelet transform is that a pair of FIR filters is applied recursively to the designated quarter of the wavelet coefficients, the result of the previous level of the filter operations. Merged arithmetic [1] produces filter results of size $\Omega_{L_v} = N + (M + \lceil \log_2 k \rceil)^{L_v}$ when applied to N-bit signal coefficients, M-bit filter coefficients, k-tap filter, and L_v levels. The problem is to minimize the size of the wavelet coefficients, without destroying the image quality.

2 Introduction

The wavelet transform is an alternative to Fourier based analysis that has found many practical applications. Image compression is an application where the wavelet transform offers an attractive quality-complexity tradeoff. For many years, wavelet transforms have been studied under different titles: multiresolution analysis [2], compactly supported wavelets [3], symmetric quadrature mirror filters [4], and the Embedded Zerotree Wavelet Algorithm [5]. These researches showed that the quality of wavelet-based image compression is better than conventional approaches such as JPEG. An effective and fast computation of the wavelet transform is necessary to create a practical application.

Merged arithmetic was introduced [1] as a way of achieving an effective implementation of compound arithmetic functions comprised of multiple arithmetic operations (such as add, subtract, and multiply). The approach is to merge one arithmetic operation into another, by dissolving the boundaries separating the discrete arithmetic operations. As a result, the desired algorithm is realized through a process that is arithmetically identical, but more efficient in performance and cost of implementation. The concept originates from the context of parallel multiplier and adder designs to compute inner products which is a core computation of digital signal processing systems as well as the wavelet transform.

The wavelet transform poses a different problem to the selection of the filter word size than the conventional digital signal processing applications. The compactly supported wavelet applies a pair of Finite Impulse Response (FIR) filters to an input image to produce the first outputs of the wavelet transform, or wavelet coefficients. The same wavelet transform is applied over and over again on the designated quarter of wavelet coefficient calculated from the prior wavelet transform until it reaches to the final level. This recursive process introduces an amplification of the word size of the wavelet coefficient. The results of the FIR filter operations have to be truncated; otherwise the word size of the coefficients increases dramatically.

An approximation to the exact solution of the wavelet transform is desired when it is considered as an image compression algorithm. The compression ratio and the error between the original and the restored image measure the quality of image compression. Therefore, a trade-off can be made between the image quality and the complexity by examining the compression ratio and the image error. It is also attractive to use a single arithmetic unit for computing the wavelet transform regardless of the level and the filtering coefficients.

0-8186-8409-7/98 $10.00

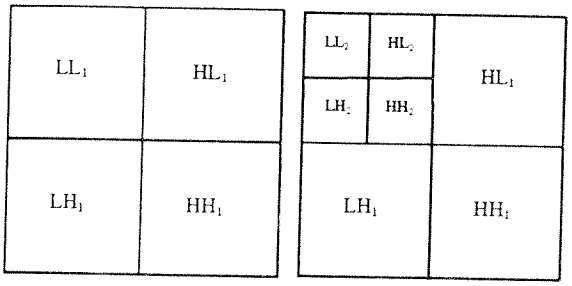

Figure 1. Wavelet decomposition: the first and second level wavelet coefficients.

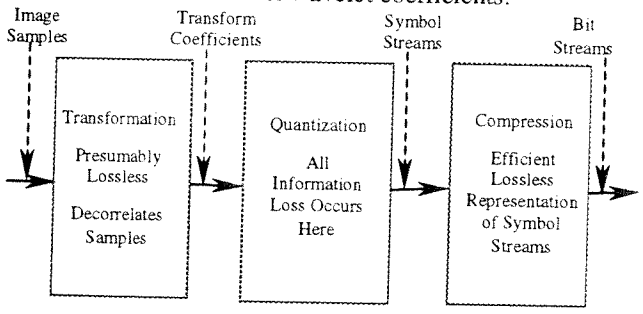

Figure 2 – Transform-based image coder.

This paper introduces a merged arithmetic implementation for the wavelet transform specifically to perform image compression algorithms such as the EZW. First, the characteristics of the compactly supported wavelets are reviewed to identify the word size requirement of the wavelet coefficient. Then, we develop an analytical solution of the design parameters. Finally, an experimental result is presented to confirm our analysis.

3 Discrete Wavelet Transform

The wavelet-based image compression used in this paper is adapted from Shapiro's Embedded Zerotree Wavelet algorithm [5]. It is based on a hierarchical subband system, where the subbands are logarithmically spaced in frequency and represent octave-band decompositions. The overall process of the decomposition is to begin with the original image and apply the four different filters, HH, HL, LH, and LL. H and L represent high-pass filter and low-pass filter, respectively. The results of the four different filter operations are referred as wavelet coefficients and they are decimated after each operation so that the four wavelet transforms have the same size as the original image. The process is recursively applied to the LL_1 wavelet coefficients as shown in Figure 1.

The EZW algorithm is utilized as a transform-based image coder throughout this paper as shown in Figure 2. The quantization of the coder is based on a zerotree data structure, and the compression is from this quantization technique alone. It is common to combine this technique with a lossless compression technique such as Huffman

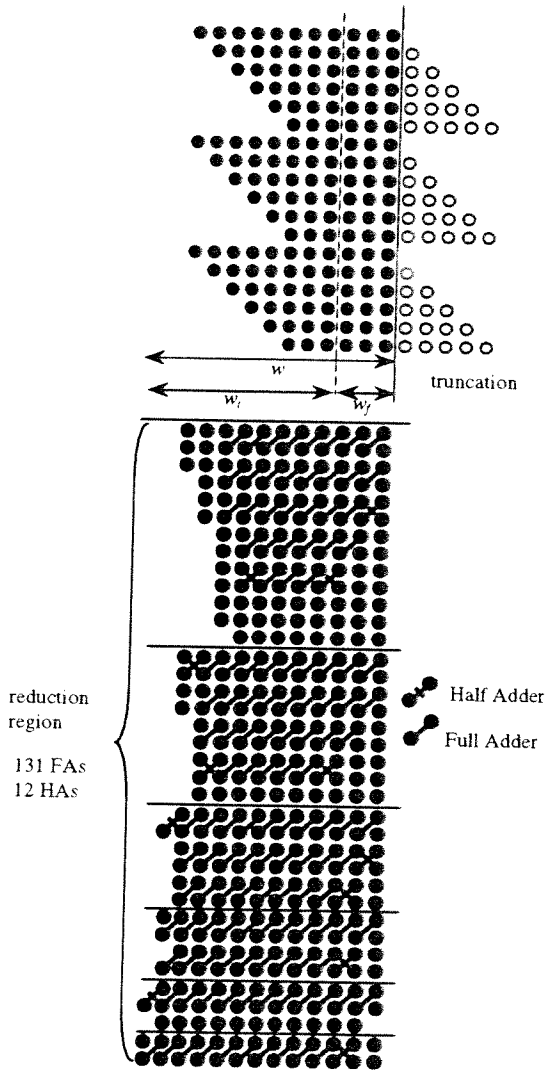

Figure 3 – Truncated reduction of a 14-bit 3-tap filter with 12-bit signals and 6-bit coefficients.

Coding to increase the compression ratio [5], [6]. This additional compression is not included in this paper because our interest is to study the effect of the approximate transform on the image quality.

Since our objective is maximum efficiency of the wavelet transform, we will examine the characteristics of the wavelet. The compactly supported wavelets provide a theoretical explanation that is especially useful for image compression. The compactly supported wavelet is represented by a set of FIR filter coefficients, $h(n)$ and $g(n)$ with the following characteristics:

$$\sum h(n) = \sqrt{2}$$

$$\sum g(n) = 0$$

$$-1 < h(n), g(n) < 1$$

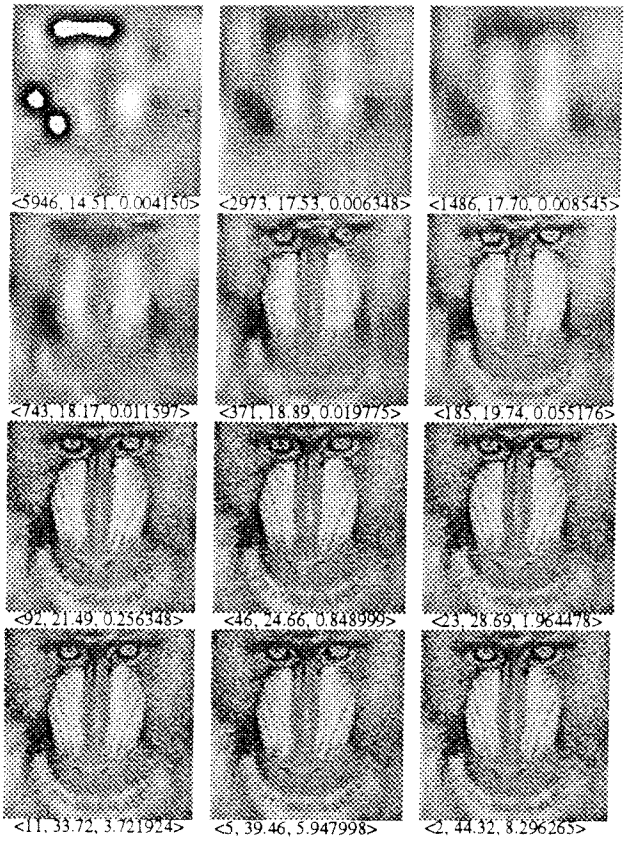

Figure 4 - Restored image sequence of floating-point EZW algorithm. The triple <T, PSNR, bpp> means that T is the EZW quantization threshold, PSNR = $10 \cdot \log_{10}(255^2/\text{MSE})$, and bpp is the compression ratio. The original image is 8 bpp, of size 512×512.

The wavelet transform applies the filters to each dimension of the image, i.e., two passes are required to complete each level of the transform. For example, the LL coefficients are obtained by applying the filter $h(n)$ to the rows of the original image followed by applying $h(n)$ to the columns of the row transforms, LH is obtained by the filter $h(n)$ followed by $g(n)$, etc.

The maximum level of the wavelet transform depends on the size of the original image and the size of the filter bank. Due to decimation, each level of the wavelet transform reduces the number of the coefficients produced by the LL filter by a factor of 4. The next level of wavelet transform is applied to this reduced region. The level of the wavelet transform is L_w, which is given by $1 + \lfloor \log_2 N/k \rfloor$ for an original image of size $N \times N$ and k coefficient filters.

4 Approach

Our design of the wavelet transform utilizes "truncated reduction" to provide a simple trade-off between the computational accuracy and complexity. It offers more flexibility than conventional merged arithmetic [1]. Truncation error is adjusted to minimize the complexity while achieving adequate image quality. Furthermore, this design maintains a constant word size throughout all the stages of the wavelet transform.

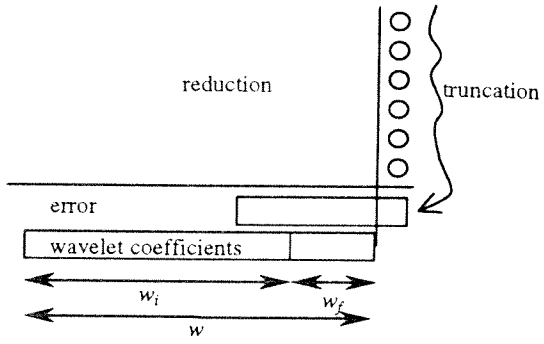

Figure 5 – Truncation process.

Merged arithmetic unifies the parallel multiplier and fast adder to achieve a reduced implementation cost. Regular merged arithmetic applies a Dadda-reduction scheme [7] to the composite bit-product matrix that is the collection of all the bit-products from the multiplication [1]. All the bits are reduced in a uniform fashion. However, truncating bits offers an opportunity to reduce the implementation cost because it eliminates a portion of the composite bit-product terms as well as the reduction counters. Figure 3 shows merged arithmetic with truncation to implement a 12-bit filter operation with fractional coefficients. In this scheme, all the bits right of the truncation line, less significant bits than the upper w bits have been deleted. The white dots represent the truncated bit-products and the black represent the active bit-products that take a part in the reduction process. The reduction is identical to the regular merged arithmetic and it is constructed with counters such as full adders and half adders as introduced in Dadda's method.

Computational error is undesirable for most applications and so is the truncation of the merged arithmetic. However, certain applications tolerate the error depending on the nature of computational error. The EZW algorithm is such an application since it employs lossy compression, as shown on Figure 2. Many experiments conducted for the wavelet application show that the wavelet is relatively insensitive to the quantization error that occurs in the less significant bits of the wavelet coefficients. The EZW or any other wavelet-based image compression algorithm produces a large amount of image information by using the most significant bits of the filter coefficients. This suggests that maintaining the integrity of the most significant part of the wavelet coefficient is critical to the EZW algorithm. Figure 4 explains such characteristics of the EZW algorithm. The Mandrill image is processed with the floating-point implementation of the EZW and the restored images are collected along with the

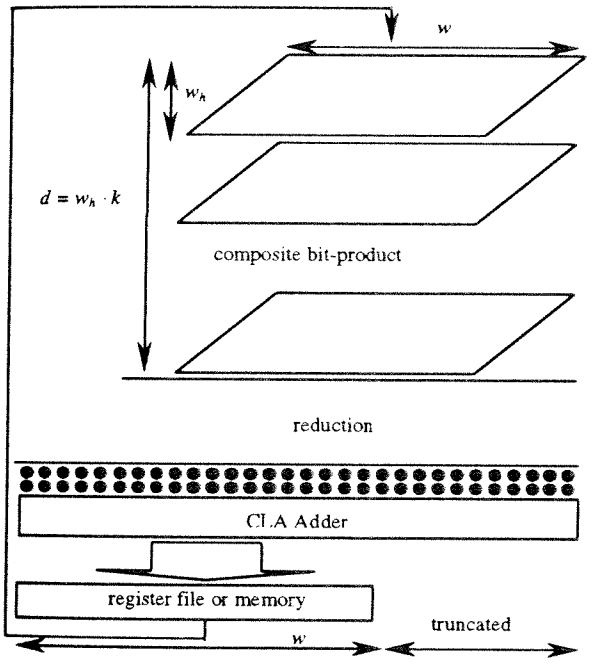

Figure 6 – Conventional w-bit merged arithmetic wavelet transform.

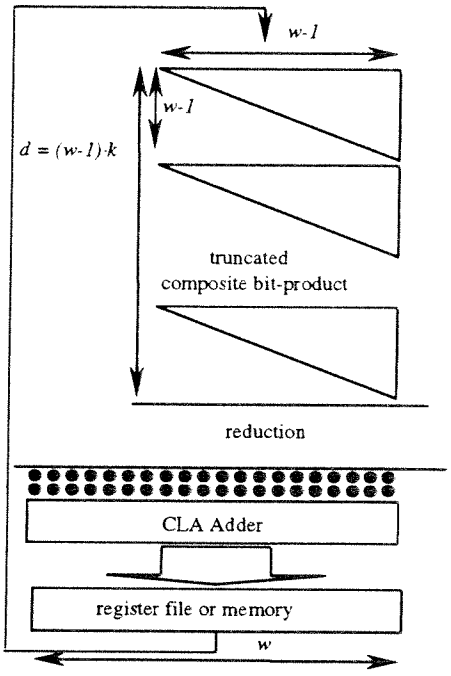

Figure 7 – Truncated w-bit merged arithmetic wavelet transform.

quantization threshold T, the signal-to-noise ratio, PSNR, and the compression ratio. The quantization threshold is used by the EZW algorithm to recover the wavelet coefficients and larger values indicate greater quantization errors.

Since the filter coefficients of the compactly supported wavelet are fractional, the merged arithmetic implementation of such a filter is shown in Figure 3. As prior discussion suggested, a truncation line is drawn vertically across the composite bit-product matrix to keep only the word size of w greater than the integer of the wavelet coefficients. Let w_f and w_i be the size of fractional and integer wavelet coefficient as shown in Figure 3. With this configuration, the w_i most significant bits of the wavelet coefficient provide the bit precision for the required image restoration quality. The w_f least significant bits compensate the error occurred in the truncation. As a result, the merged arithmetic implementation of the wavelet transform can use registers of size $w = (w_i + w_f)$ for the wavelet coefficients.

Our design goal is to position the truncation line so that the wavelet transform produces image quality that is comparable to the floating-point implementation. This is done by estimating w_f, the size of the fractional wavelet coefficient. In order to find this value, let us examine the truncation process as shown in Figure 5. The truncated bits are equivalent to a single number that we refer to an error. Due to the big size of the bit-product matrix, the error value is a multiple bit number. As indicated in Figure 5, the error occurs at the integer wavelet coefficient when the bit size of the error value is greater than w_f. This is *integer-error condition* of merged arithmetic described as follows:

$$2^{w_f} < (w_i + w_f - 1) \cdot k$$

for a k-tap merged arithmetic filter bank with infinite precision of the filter.

Compactly supported wavelet states that each filter has fractional coefficients that sum to less than or equal to $\sqrt{2}$ for each filter. Since the filter is applied in two passes to both dimensions of the image, an additional bit is required for each level of the wavelet transform. Therefore, the integer size of the wavelet coefficients should be

$$w_i = 1 + \Omega_p + L_w$$

if the pixel resolution is Ω_p and wavelet transform is performed up to the L_w level.

With w_i obtained above, the *integer-error condition* estimates the lower boundary of the parameter w_f that does not introduce an error to the integer wavelet coefficient:

$$2^{w_f} - w_f k \geq (\Omega_p + L_w) \cdot k$$

The minimum w_f satisfying this condition is approximately equivalent to $\lceil 1+\log_2(\Omega_p + L_w) \cdot k \rceil$. Therefore the word size of the wavelet coefficient is

$$w = w_i + w_f = 2 + \Omega_p + L_w + \lceil \log_2(\Omega_p + L_w) \cdot k \rceil$$

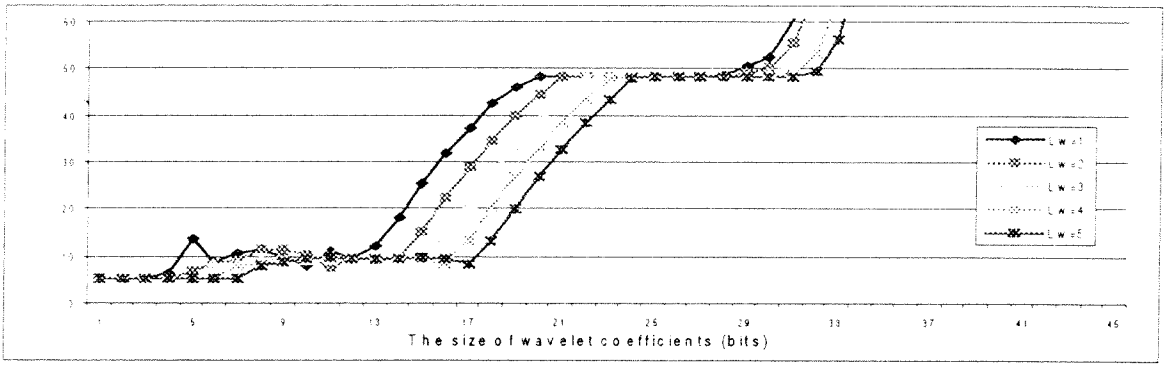

Figure 8 – Image quality of wavelet transform.

for the wavelet transform of an image with a pixel size, Ω_p, k-tap filter bank, and up to L_w level.

5 Results

For a typical application such as EZW with an image of size 512 × 512 by 8-bits, and 9-tap filters [5], the word size should be at least 23 bits (15-bit integer and 8-bit fractional.) The regular merged arithmetic implementation of the EZW is shown in Figure 6; the wavelet coefficient of the merged arithmetic is truncated into w-bits. This provides the wavelet transform with a fixed word size. It is also possible to implement the wavelet transform by utilizing truncated reduction as shown in Figure 7. These two implementations provide image quality that is comparable to the full precision floating-point implementation.

An experiment has been conducted to validate the analytical solution for the design parameters of the truncated reduction. The wavelet transform is performed to the Mandrill image by using merged arithmetic with various truncations. The result wavelet coefficients are applied to the inverse wavelet transform to restore the image by using the same arithmetic. The restored images and the original image are compared to compute the PSNR by using the following formula:

$$\text{PSNR (dB)} = 10 \cdot \log_{10} \frac{255^2}{MSE}$$

$$MSE = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \frac{(x_{ij} - \hat{x}_{ij})^2}{N^2}$$

where x_{ij} is the original image and $\hat{x}_{ij}$ is the restored image of size $N \times N$. The signal-to-noise ratio, PSNR, is measured for each truncation and plotted along with the size of the wavelet coefficient as shown in Figure 8. The experiment shows that the image quality has reached to PSNR of 45 dB when the word size is around 23 at $L_w = 5$. This is equivalent to the maximum image quality of the integer wavelet coefficient of the floating-point implementation (see Figure 4).

Each level differs approximately by one bit as predicted in our analysis. It is interesting that the 5-level image quality is gradually improved as the word size increases from 17 to 24 bits. But it stays flat from 24 through 32 bits. It increases again, but rapidly after 33 bits. This indicates that the fractional portion of the composite bit-product matrix makes a major contribution to the image quality of the wavelet transform. This contribution is incremental until there is no more error to the integer wavelet coefficients (*integer-error condition*). It also indicates that the perfect image restoration requires the 6-level wavelet coefficients larger than 34 bit.

		NOT-MERGED with adder tree		REGULAR MERGED 20-bit filter coefficients		TRUNCATED MERGED	
FUNCTION	GATES	USAGE	GATES	USAGE	GATES	USAGE	GATES
AND GATE	1	4320	4320	4320	4320	2484	2484
HALF ADDER	4	333	1332	60	240	51	204
FULL ADDER	9	3573	32157	4312	38808	2691	24219
27 Bit CLA	351					1	351
44 Bit CLA	620	13	8060				
45 Bit CLA	620	2	1240				
46 Bit CLA	641	1	641				
47 Bit CLA	641	1	641	1	641		
TOTAL GATE COUNT		(110%) 48391		44009		(62%) 27258	

Table – 24-bit Wavelet transform (9-tap filter bank) implementation comparison: the size assessment of CLA is made by utilizing gates with 2 to 4 inputs, each counted as one gate.

The second experiment is to build two different circuits to realize the merged wavelet arithmetic; one is the conventional merged arithmetic and the other is 24-bit merged arithmetic implementation with truncated reduction. Both implementations offer a similar speed performance, but the 24-bit truncated reduction offers a substantial reduction in the cost of implementation. This result is shown in Table along with a comparison to the adder tree implementation [1]; it shows a substantial saving in the gate count of the truncated reduction (40%). The quality of the restored image of the 24-bit EZW by utilizing the truncated reduction is provided in Figure 9 and it reveals performance that is very close to the floating-point implementation.

6 Conclusion

Truncated reduction is investigated as an alternative to the regular merged arithmetic that provides a very effective implementation of the wavelet transform. The truncated reduction offers a simplified assessment of the trade-off between the image quality and the computation accuracy that normally becomes very complicated for a recursive process such as wavelet transform. Our design achieved an optimal operating condition with a single design parameter that is consistently applied to the entire process of the wavelet transform. As a result, the wavelet coefficient was reduced to a fixed word size with a substantial reduction in the implementation cost (on the order of 40%).

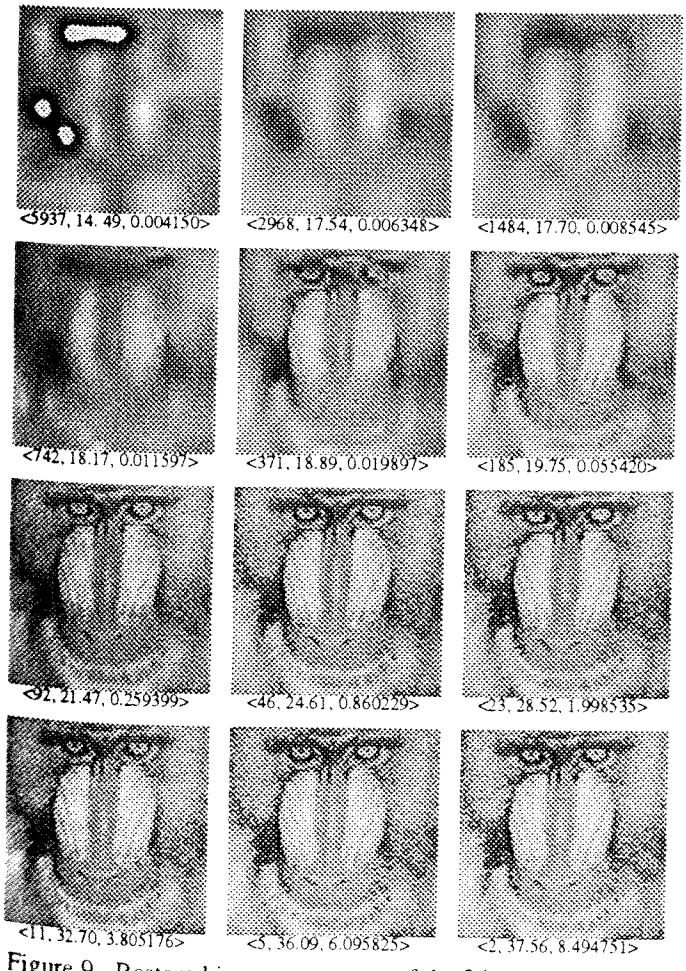

Figure 9 - Restored image sequence of the 24-bit merged arithmetic with a truncation.

Our study shows that the design parameter of the truncated reduction is

$$w = 2 + \Omega_p + L_w + \lceil \log_2(\Omega_p + L_w) \cdot k \rceil$$

for the wavelet transform of an image with a pixel size, Ω_p, k-tap filter bank, and up to L_w level. The word size w is fixed for given L_w, k and Ω_p. The result wavelet arithmetic unit is capable to handle an arbitrary size of image up to L_w. Our research concludes that 24-bit merged arithmetic of the wavelet transform is appropriate for image compression algorithms (such as EZW) to handle up to size of 512 × 512 of 8-bit image with 9-tap filter bank.

7 Reference

1. E. E. Swartzlander, Jr., "Merged Arithmetic," IEEE Trans. Computers, vol. C-29, 1980, pp. 946-950.
2. S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, 1989, pp. 674-693.
3. I. Daubechies, "Orthonormal Bases of Compactly Supported Wavelets," *Commun. Pure and App. Math.*, vol. XLI, 1988, pp. 909-996.
4. E. H. Adelson, E. Simoncelli, and R. Hingorani, "Orthogonal pyramid transforms for image coding," *Proc. SPIE*, vol. 845, 1987, pp. 50-58.
5. J. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients," *IEEE Trans. Signal Processing*, vol. 41, 1993, pp. 3445-3462.
6. I. H. Witten, R. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Comm. ACM*, vol. 30, 1987, pp. 520-540.
7. L. Dadda, "Some Schemes for Parallel Multipliers," *Alta Frequenza*, vol. XXXIV, 1965, pp. 349-356.