January, 1996

Subband absolute moment block truncation coding

Kai-Kuang Ma, *Nanyang Technological University*
Sarah A. Rajala, *North Carolina State University*

Subband absolute moment block truncation coding

Kai-Kuang Ma
Nanyang Technological University
School of Electrical and Electronic
Engineering
Nanyang Avenue
Singapore 639798, Republic of Singapore

Sarah A. Rajala
North Carolina State University
Department of Electrical and Computer
Engineering
Raleigh, North Carolina 27695-7911

Abstract. In this paper, a new subband coding system called subband absolute moment block truncation coding (SAMBTC) is introduced to compress monochrome images and color images recorded in *YIQ* and *L*u*v** uniform color spaces. The SAMBTC incorporates full-band absolute moment block truncation coding (AMBTC) into subbands along with a new subband dynamic bit allocation algorithm which is derived from the Shannon rate-distortion bound. Simulation results show that the proposed Shannon-bound-based bit allocation algorithm outperforms the commonly used standard-deviation-based bit allocation scheme. Compared with AMBTC, SAMBTC achieves superior imaging without blocking artifacts at low bit rates.

Subject terms: visual communication and image processing; image coding; subband coding; dynamic bit allocation; absolute moment block truncation coding; moment-preserving quantizer; uniform color space.

Optical Engineering 35(1), 213–231 (January 1996).

1 Introduction

The principle of subband coding[1–3] is based on the decomposition of an input signal into narrow bands, called subbands, by a set of analysis filters at the encoder. Each subband is then maximally decimated and separately encoded according to a certain bit allocation criterion. At the decoder, the decimated and encoded subbands are decoded, interpolated, and filtered by another set of synthesis filters before being added together to reconstruct the original signal. Multidimensional subband analysis/synthesis theory was first introduced by Vetterli.[4] Later, the first subband image coding results were presented by Woods and O'Neil.[5] Since then, subband coding has been considered to be one of the major coding techniques for compressing images and video.

Subband coding is essentially a frequency-domain approach which consists of two subsystems: (1) subband analysis/synthesis (i.e., subband representation) and (2) encoder/decoder (i.e., codec) along with a subband dynamic bit allocation scheme. Notice that the subband representation provides a special way to represent the original source without any compression. The subband bit allocation algorithm dynamically decides how many bits to assign to each subband; it does not provide any compression. Only the codec performs compression and decompression functions, based on the bit allocation results. For example, Woods and O'Neil[5] exploit differential pulse code modulation (DPCM) and its adaptive version to encode the individual subbands. Gharavi and Tabatabai[6] adopt a hybrid approach using DPCM for the baseband and pulse code modulation (PCM) for the higher subbands. Westerink et al.[7] apply vector quantization (VQ) to code the subbands. In contrast to these works,[5–7] we exploit a special class of 1-bit (or two-level) moment preserving quantizers, called the *absolute moment block truncation coding* (AMBTC),[8] as the codec for subbands in this paper.

Compared with many well-known coding schemes (e.g., discrete cosine transformation, DPCM, VQ),[9–13] AMBTC provides fairly competitive image quality at medium to high bit rates[8] and has a much simpler computational structure. However, we have shown that AMBTC has a bit rate lower bound of 1 bit per pixel and presents severe blocking artifacts on both monochrome and color images at low bit rates. To overcome these drawbacks while fully benefiting from the merits of AMBTC, a new subband coding system, called *subband absolute moment block truncation coding* (SAMBTC) is introduced in this paper. The SAMBTC system combines full-band AMBTC, subband analysis/synthesis, and a new subband dynamic bit allocation algorithm. For simulation purposes, two monochrome and two color digital images recorded in various color spaces are considered.

Section 2 describes how we decompose the source into multiple subbands. Section 3 summarizes the AMBTC algorithm that is used as the codec of the subband coding system. In Sec. 4, a new dynamic bit allocation algorithm is derived (called the direct form) subject to the Shannon rate-distortion relationship using a Lagrangian multiplier optimization technique. A sequential implementation form is then developed from the direct form in order to precisely match the bit rates of the available AMBTC windows. The new algorithm and the commonly used standard deviation bit allocation algorithm are both converted into the sequential implementation form for comparing their performance. In Sec. 5, the coding performance using the SAMBTC system is presented and compared with the one using AMBTC. Starting from Sec. 6, the development is extended to color images

Paper VIS-12 received May 20, 1995; revised manuscript received Aug. 31, 1995; accepted for publication Sep. 3, 1995.

recorded in conventional color space *YIQ* and uniform color space $L^*u^*v^*$. Section 7 presents some new observations regarding the bit plane and the blocking artifacts while using AMBTC on color images. In Sec. 8, the developed new subband dynamic bit allocation algorithm for monochrome images is extended to color images. Some simulation results are presented in Sec. 9. Section 10 presents the conclusions.

2 Subband Filtering for Images

The theory and development of subband filtering can be found in numerous sources (e.g., Refs. 1, 2, 14, 15). In this paper, 2-D subband filtering for digital images is implemented by sequentially applying 1-D two-channel quadrature mirror filtering (QMF)[5,15–17] to the source image in both horizontal and vertical directions (called *2-D separable filtering*). There are many ways to split a source image into multiple subbands with equal or unequal bandwidths. As in Woods and O'Neil,[5] the filter coefficients of the 1-D Johnston QMF filter (32D in Ref. 17) have been used to generate 16 equal-bandwidth subbands for each image. Through extensive simulations, the difference in coding performance using 16-tap and 32-tap QMF filters in the SAMBTC is fairly small and unnoticeable.

The boundary effects are ringing artifacts introduced in the finite impulse response (FIR) filtering of a 2-D digital image due to its finite region. In order to reduce the artifacts, both the symmetric extension method proposed by Smith and Eddins[18] and the boundary replication method suggested by Vetterli[19] have been evaluated. Vetterli's boundary replication method[19] is subjectively preferred and is used in this paper.

3 Absolute Moment Block Truncation Coding

3.1 *Background*

Consider an image divided into nonoverlapping squared blocks. Delp and Mitchell[20] developed the block truncation coding (BTC) algorithm in which the local mean and variance of each block are preserved.* The BTC has been applied to compress images[20–22] and video.[23] Halverson et al.[24] generalized the BTC to preserve a family of moments, called the *generalized BTC* (or GBTC). They derived a closed-form solution of preserving $(n,2n)$ for the two-moment GBTC and $(n,2n,3n)$ for the three-moment GBTC (where n is an integer and small n means low-order moment). Lema and Mitchell[8] modified the constraints of BTC and developed a new algorithm called the AMBTC. For an extensive study of BTC, refer to a recent publication by Dasarathy.[25]

Assume each block has a size of $n\times n$. Let $m=n^2$ be even and let $\mathbf{x}=[x_1,x_2,\ldots,x_m]^T$ be the vector of the gray level of the pixels in each block. In AMBTC, the block mean η and the first absolute central moment $\alpha=(1/m)\Sigma_{i=1}^{m}|x_i-\eta|$ $=(1/m)\Sigma_{i=1}^{m}|y_i-\eta|$ (rather than the sample variance σ^2 $=\overline{\mathbf{x}^2}-\eta^2$ in BTC) are preserved; where x_i and y_i are the gray level of input and output pixels, respectively. Like BTC and GBTC, AMBTC is also a 1-bit (or two-level) quantizer since only two quantization levels (denoted as **a** and **b**) are generated in each block. They all can be categorized into a class called the 1-bit *moment-preserving quantizer* (MPQ).[26]

It is worth mentioning that there is another special type of two-level quantizer, called the *1-bit minimum mean-square error quantizer* (MMSEQ). The 1-bit MMSEQ has the same algorithmic structure as the AMBTC but uses a different truncation threshold as follows. AMBTC uses the block mean while 1-bit MMSEQ requires an iterated search for the optimum [in the mean-square error (MSE) sense] threshold on each block. The 1-bit MPQ and the 1-bit MMSEQ can be categorized as 1-bit quantizers. Note that the MPQ preserves certain moments but may not achieve the minimum MSE. On the other hand, the 1-bit MMSEQ has a minimum MSE but may not preserve the moments. The reader is referred to Ref. 27 for a more detailed discussion.

The AMBTC is chosen among the 1-bit quantizers as the codec for subbands based on the following reasons. First, AMBTC has a smaller computational complexity than BTC. This can be easily justified from their algorithmic structures.[8,20] Second, we have mathematically proved that AMBTC achieves the least MSE among the 1-bit quantizers subject to preserving the block moments.[28] Third, we have shown that AMBTC is more practical to use than the 1-bit MMSEQ.[28] This is because the images resulting from using these two algorithms are fairly close while the 1-bit MMSEQ's computations are at least two times higher than that of the AMBTC's. Fourth, Halverson et al.[24] demonstrated that whether the GBTC outperforms the BTC (in terms of peak-to-peak signal-to-noise ratio, PSNR) depends on the image and the chosen model, two-moment or three-moment. They also pointed out that the computation of very high moments frequently leads to overflow and inferior performance. Finally, AMBTC is suitable for real-time application and is feasible to implement. A real-time implementation of the AMBTC using hardware only for the compression of the National Television Systems Committee (NTSC) television pictures has been reported by Ko and Lee.[29] In addition, the VLSI implementation of the AMBTC algorithm for video signal processing has been recently documented by Chen et al.[30]

3.2 *The AMBTC Algorithm*

An image is first divided into nonoverlapping blocks with size $n\times n$ each (usually, $n=2^i$, where i is a positive integer). The AMBTC encoding process is independently applied to individual blocks as follows:

Step 1: For each block of pixels $\mathbf{x}=[x_1,x_2,\ldots,x_m]^T$, the mean value $\eta=(\Sigma_{i=1}^{m}x_i)/m$ is calculated and used as the block (or local) threshold.

Step 2: An $n\times n$ bit plane consisting only of 1's and 0's is generated from the original block by independently applying a 1-bit quantizer to each pixel in the block such that the quantizer output is 0 (if $x_i\leqslant\eta$) or 1 (if $x_i>\eta$), where $1\leqslant i\leqslant m$. This results in $(m-q)$ 0's and (q) 1's in the bit plane.

Step 3: Two quantization levels **a** and **b** (generally assumed to be 8 bits each) are calculated using the following formulas:

$$\mathbf{a}=\eta-\frac{m\alpha}{2(m-q)}=\frac{1}{m-q}\left(\sum_{\text{for } x_i\leqslant\eta}x_i\right), \tag{1}$$

*Note that the 1-bit quantizer can be used to maximally preserve three moments, called *three-moment BTC*.[20] However, the BTC referred to in this paper implicitly means two-moment BTC.

$$\mathbf{b}=\eta+\frac{m\alpha}{2q}=\frac{1}{q}\left(\sum_{\text{for } x_i>\eta} x_i\right) . \tag{2}$$

That is, the combination of the two quantization levels **a** and **b** and the bit plane provides an approximate representation for each block. Thus, only the two quantization levels and the bit plane are transmitted and/or stored. For the decoding process, the 0's and 1's of the bit plane on each block are simply substituted by the quantization levels **a** and **b** of that block, respectively. Thus, the AMBTC codec is unsymmetrical due to its greatly reduced computational load at the decoder.

3.3 *Fundamental Insights*

In the following paragraphs, we provide several important insights on the AMBTC algorithm. The arguments are also applicable to the 1-bit MPQ. More properties of the AMBTC can be found in Refs. 27 and 28. Recently, Delp and Mitchell[26] provided fundamental insights on the MPQ through the Gauss–Jacobi mechanical quadrature.

3.3.1 *Bit-rate lower bound–1 bit/pixel*

It can be shown that all the 1-bit moment-preserving quantizers are bounded below by 1 bit/pixel. Let the window size be $n\times n$ and let the number of bits representing levels **a** and **b** be N_a and N_b, respectively. The average bit rate for this $n\times n$ block of pixels can be calculated as

$$R_{\text{MPQ}}=\frac{n^2+N_a+N_b}{n^2} . \tag{3}$$

Theoretically, when the window size approaches infinity, the bit rate R_{MPQ} is 1 bit/pixel, which is the bit-rate lower bound of the MPQ.

For those applications which require less than 1 bit/pixel, all 1-bit MPQs are not applicable unless some modifications are applied, and/or additional compression techniques are cascaded (e.g., entropy coding). However, the SAMBTC approach proposed in this paper is not constrained by this lower bound. For example, the source image can be decomposed into subbands through several levels of octave band splitting. The higher subbands can be unconditionally discarded, leading to total bit rates for the reconstructed image that are less than 1 bit/pixel.

Table 1 summarizes the bit rate contributions from the bit plane and the quantization levels in the AMBTC for various window sizes. It is important to realize that the bit plane always contributes 1 bit/pixel and is independent of the window size. Owing to the lower bound, the number of effective AMBTC windows is small. For the simulations conducted in this paper, we chose a set of windows as listed in Table 1 plus 0×0 and 1×1 to denote "discard" and "no compression," respectively.

Table 1 Bit rate contributions from the bit plane and the quantization levels in the AMBTC for various window sizes.

Window Size	*Bit Plane Rate (bpp)*	*Quant. Levels Rate (bpp)*	*Total Rate (bpp)*	*Bit Plane Percentage*
2 × 2	1	4.0	5.0	20%
4 × 4	1	1.0	2.0	50%
8 × 8	1	0.250	1.250	80%
16 × 16	1	0.06250	1.06250	94.1%
32 × 32	1	0.0156250	1.0156250	98.5%
64 × 64	1	0.0039063	1.0039063	99.6%

3.3.2 *Major artifacts*

Compared with other coding algorithms, AMBTC provides fairly competitive compressed imagery quality at 2 bits/pixel (equivalent to using a 4×4 window) or higher rates[20] (i.e., using smaller windows). However, the AMBTC coding performance quickly degrades at low bit rates (e.g., using an 8×8 or larger window). In particular, two major artifacts occur: contouring artifacts and blocking artifacts.

With only two reconstruction levels, AMBTC introduces jaggedness. For the flat areas in an image, the pixel values vary slowly. As a result, the jaggedness would cause contouring artifacts due to abrupt changes in the reconstruction values. Generally speaking, contouring artifacts are less noticeable in the busy areas. Second, note that no matter what the AMBTC window size is, there are only two quantization levels. Therefore, for the large window sizes, another type of distortion occurs, called *blocking artifacts.* This is because an abrupt change in the reconstruction value occurs at block boundaries. The larger the AMBTC window, the more distinct the blocking artifacts are. In summary, the contouring artifacts are an intrablock distortion, and the blocking artifacts are an interblock distortion.

To demonstrate these artifacts, two monochrome images, "Lena" and "House" (see Fig. 1), are used for the simulations. Each image has a size of 256×256 pixels with 8 bits/pixel. Four compressed images using the AMBTC 8×8 and 32×32 for "Lena" and "House" are presented in Fig. 2. These images will be compared with those (in Fig. 8) using SAMBTC at the same bit rates. The MSE and PSNR measurements can be found in Table 4 of Sec. 5.

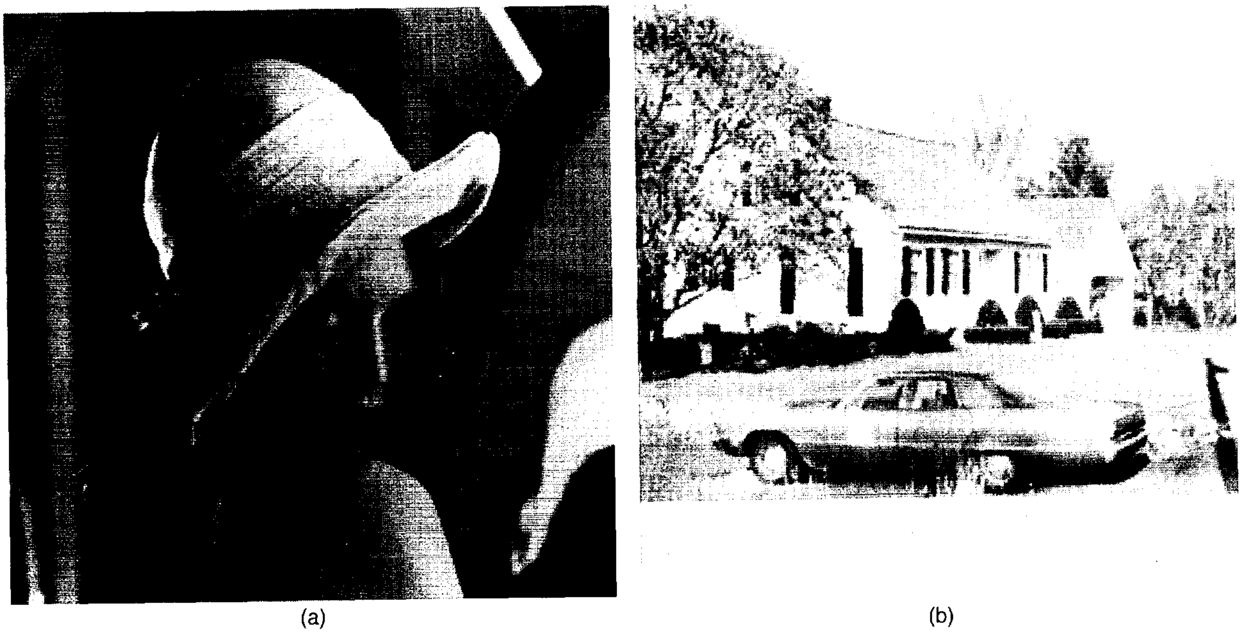
4 Subband Dynamic Bit Allocation

4.1 *Overview*

Dynamic bit allocation is a major concern in coding systems where a given number of bits must be efficiently distributed among a number of sources. For subband dynamic bit allocation, these sources are the decomposed subbands.

For vector source coding, Huang and Schultheiss[31] first recognized the issue of optimum bit allocation for transform coding and derived an analytical expression for a number of Gaussian sources. The standard-deviation-based dynamic bit allocation algorithm described in Eq. (15) is a direct result of their work and probably the most frequently used bit allocation algorithm both in image (e.g., Ref. 5) and in speech (e.g., Ref. 32) processing. However, the derivations were based on an approximate relationship between the quantizer distortion and the corresponding bit rate, which is only accurate for high bit rates. Thus, the solution is approximate and suboptimum. In addition, if the standard deviation of a subband is small enough, the number of bits allocated for a subband could be negative. In such a case, Huang et al.[31] used a trial-and-error procedure for practical bit allocation. Improvements to this method were proposed by Segall[33] and Ramstad.[34]

Segall[33] assumed that the rate-distortion function does not have to be exponential and developed an optimal solution for non-negative real-valued bit allocation. Ramstad[34] de-



(a)

(b)

Fig. 1 Two uncompressed monochrome images: (a) "Lena" and (b) "House." Each image has a size of 256×256 pixels with 8 bits/pixel.

signed a practical bit allocation process which is an iterative procedure and avoids negative bit assignment. Both Huang et al.[31] and Segall[33] assumed that all the quantizers were identical. Fox,[35] on the other hand, removed this assumption and proposed a new algorithm based on marginal return analysis but assumed that the rate-distortion function is strictly convex.

Without assuming the nature of the quantizers, Shoham and Gersho[36] and Trushkin[37,38] developed bit allocation algorithms using a generalized Lagrangian multiplier method[39] and dynamic programming, respectively. Instead of assuming any mathematical relationship between rate and distortion, they assumed that all the quantizers have already been designed and consequently that all possible pairs of rate and distortion are known. That is, for N subbands and M quantizers, M^N pairs must be computed in advance. The algorithm then searches for the best combination of existing quantizers to achieve the minimum distortion subject to a given total bit rate. Westerink[7,40] generalized Trushkin's algorithm to allow for real-valued bit rates. Instead of using dynamic programming, Westerink applied a convex hull searching technique[41] to improve searching efficiency. However, these methods tend to be computationally intensive and may not be practical for real-time implementations.

4.2 *Our Approach*

In this paper, a new bit allocation algorithm is developed from the foundation of Goodman[42] and is compared with the one presented by Huang and Schultheiss[31] [Eq. (15)].

Goodman[42] first considered the problem of optimally allocating the total available bit rate for encoding analog messages. Goodman began with a formula given by Shannon[43] which relates the rate and the mean-square error. He presented closed forms for the optimum rate allocation and the minimum total distortion but did not show the derivations or prove whether the stationary point was a minimum or maximum. In this paper, a Lagrangian multiplier optimization technique is used to derive an optimum bit allocation algorithm subject to the Shannon rate-distortion bound.[43] Compared with Goodman's work, our additional contributions are as follows. Complete mathematical proofs have been provided in Ref. 44 and show that the derived bit allocation algorithm achieves the minimum distortion. Second, the whole derivation is then extended for the development of equal-bandwidth subband bit allocation. Third, the sequential form is developed to effectively solve the practical bit allocation concerns.

4.3 *Shannon Bound Dynamic Bit Allocation—The Direct Form*

Shannon proved that the required bit rate $\mathcal{R}$ for any source with zero mean and bandwidth $\mathcal{W}$ is bounded by[43]

$$\mathcal{R} \leq \mathcal{W} \log_2 \left(\frac{\mathcal{P}_{\text{ave}}}{\mathcal{N}} \right) , \tag{4}$$

where $\mathcal{P}_{\text{ave}}$ is the average signal power of the source, and $\mathcal{N}$ is the allowable mean-square error or noise power between the original and recovered sources. The Shannon bound is used to specify a relationship between the bit rate and the distortion for each subband. A subband bit allocation algorithm is then derived based on this bound using a Lagrangian multiplier optimization technique. Given subband i with bandwidth w_i (bits/pixel) and average signal energy $\mathcal{P}_i$, the bit rate b_i (bits/pixel) and the resulting mean-square distortion $\mathcal{D}_i$ is given by

$$\mathcal{D}_i = \mathcal{P}_i [2^{-(b_i/w_i)}] . \tag{5}$$

The quantity $\mathcal{P}_i$ for subband i (with the size of $M \times N$) is computed by:

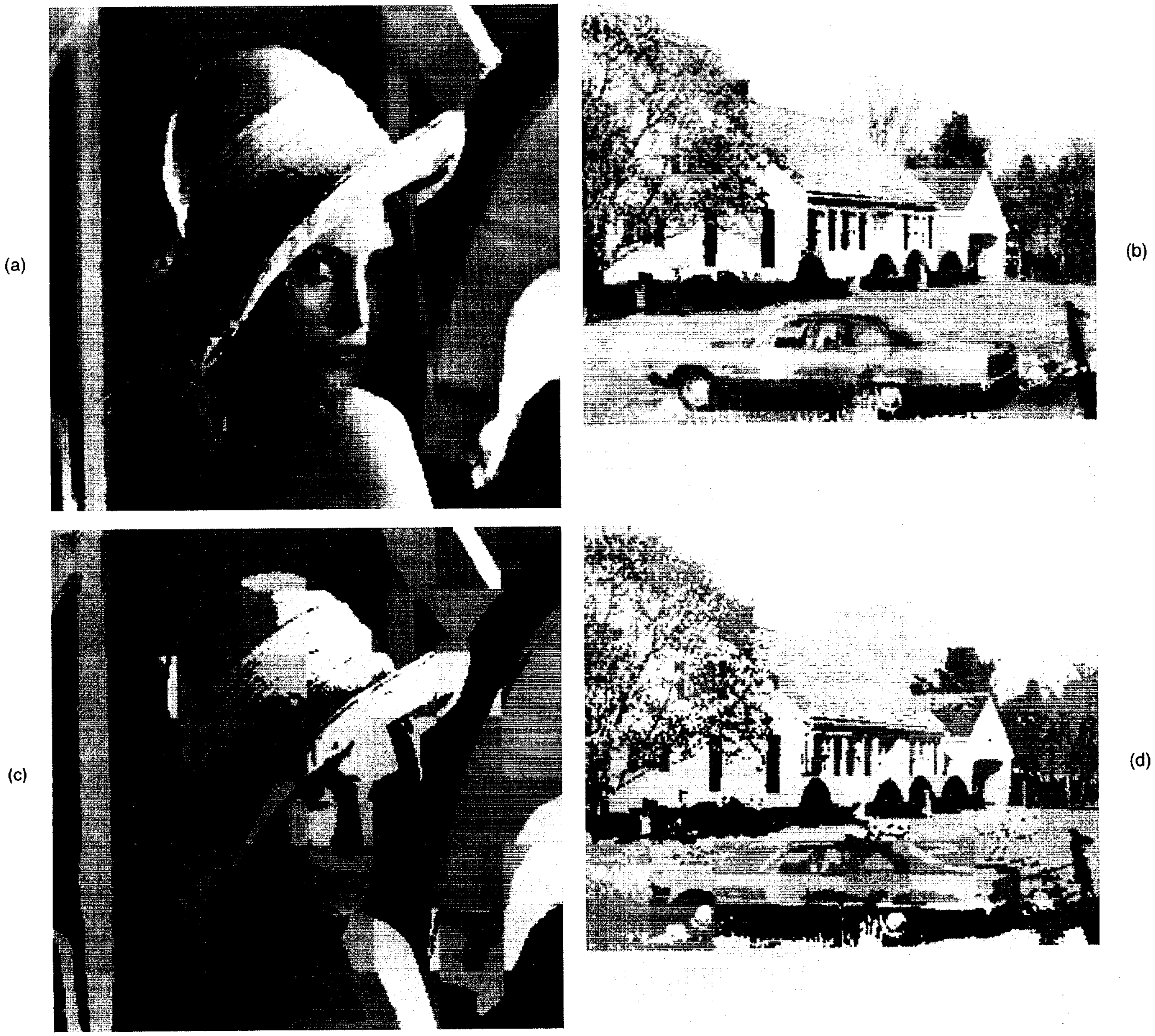


Fig. 2 The compressed 256×256 "Lena" and "House" images using full-band AMBTC at (a) and (b), an 8×8 window (i.e., 1.250 bits/pixel), and (c) and (d), a 32×32 window (i.e., 1.015625 bits/pixel).

$$\mathcal{P}_i=\frac{1}{M\times N}\sum_{m=1}^{M}\sum_{n=1}^{N} I^2(m,n) \ , \tag{6}$$

where $I(m,n)$ is the gray-level intensity of the pixel (m,n).

Let the total distortion of a compressed image be $\mathcal{D}$. Since the subband synthesis is a linear process, it is assumed that the total distortion $\mathcal{D}$ is a sum of the individual subband's mean-square distortion $\mathcal{D}_i$. That is,

$$\mathcal{D}=\sum_{i=1}^{K}\mathcal{D}_i=\sum_{i=1}^{K}\mathcal{P}_i\, 2^{-(b_i/w_i)} \tag{7}$$

subject to the constraints

$$\mathcal{B}=\sum_{i=1}^{K} b_i \, , \quad \mathcal{W}=\sum_{i=1}^{K} w_i \, , \qquad i=1,...,K \ , \tag{8}$$

where $\mathcal{B}$ and $\mathcal{W}$ are the given total bit rate and the total bandwidth, respectively. Both have the dimension of bits/pixel for K subbands. It is important to note that the subband bit allocation algorithm essentially allocates the total number of available bits among the subbands. However, the bit rate (bits per pixel) is used in the discussion of dynamic bit allocation. That is, if subband j is assigned a bit rate r, it means that subband j is allocated a fixed number of bits so that the resulting bit rate is equal to r. Thus, $\mathcal{B}$ is the total bit rate (i.e., bits per pixel per K subbands) distributed among K subbands, and b is the average bit rate $(\mathcal{B}/K)$.

Let the bit allocation vector be denoted as $\mathbf{b}\triangleq[b_1,b_2,\ldots,b_K]^T$. That is, subbands 1,2, through K have been separately allocated bit rates equal to b_1, b_2, $\ldots$, and b_K, respectively. Let λ be the Lagrangian multiplier, with the Lagrangian (or Hamiltonian) defined as

$$\mathcal{H}(\mathbf{b},\lambda)=\sum_{i=1}^{K}\mathcal{P}_i 2^{-(b_i/w_i)}+\lambda\left[\mathcal{B}-\sum_{i=1}^{K}b_i\right] . \qquad (9)$$

To find the stationary point $\mathbf{b}^*$ and to prove that it is a minimum point, the first and second derivatives were computed.[44] It has been shown that the Hessian (or curvature) matrix is positive definite. As a result, the optimum bit allocation for any subband j with bandwidth w_j and average signal energy $\mathcal{P}_j$ is derived as (arbitrary bandwidth case):

$$b_j^*=\left(\frac{w_j}{\mathcal{W}}\right)\mathcal{B}+\sum_{i=1}^{K}\left(\frac{w_i w_j}{\mathcal{W}}\right)\log_2\left(\frac{\mathcal{P}_j w_i}{\mathcal{P}_i w_j}\right) \qquad (10)$$

for $j=1,2, \ldots, K$. The minimum total distortion $\mathcal{D}^*$ can be obtained as:

$$\mathcal{D}^*=2^{-(\mathcal{B}/\mathcal{W})}\sum_{j=1}^{K}\mathcal{P}_j\left[\Pi_{i=1}^{K}\left(\frac{\mathcal{P}_i w_j}{\mathcal{P}_j w_i}\right)^{(w_i/\mathcal{W})}\right] . \qquad (11)$$

When the subband bandwidths are equal [i.e., $w_i=(\mathcal{W}/K)$ for all i], Eqs. (10) and (11) can be further simplified as follows. Define the geometric mean of the average signal energy of K subbands as

$$\mathcal{P}_g\triangleq[\Pi_{i=1}^{K}\mathcal{P}_i]^{(1/K)} . \qquad (12)$$

Then, Eq. (10) becomes

$$b_j^*=\frac{\mathcal{B}}{K}+\left(\frac{\mathcal{W}}{K}\right)\log_2\left(\frac{\mathcal{P}_j}{\mathcal{P}_g}\right) \quad \text{(equal bandwidth case).} \qquad (13)$$

Note that b_j^* is the optimally allocated bit rate for the subband j. From Eq. (11), the average minimum distortion per subband (for the equal bandwidth case) is

$$\mathcal{D}_{\text{ave}}^*=\frac{\mathcal{D}^*}{K}=\frac{1}{K}\left\{[2^{-(\mathcal{B}/\mathcal{W})}]\sum_{j=1}^{K}\mathcal{P}_j\,\Pi_{i=1}^{K}\left(\frac{\mathcal{P}_i}{\mathcal{P}_j}\right)^{(1/K)}\right\}$$

$$=\mathcal{P}_g[2^{-(\mathcal{B}/\mathcal{W})}] , \qquad (14)$$

which is the same formula as the Shannon rate-distortion bound expressed in Eq. (5) except that Eq. (14) uses the total information ($\mathcal{P}_g$ for $\mathcal{P}_i$, $\mathcal{B}$ for b_i, and $\mathcal{W}$for w_i).

Note that Eq. (13) has a similar form as the commonly used standard deviation[†] dynamic bit allocation algorithm,[31] except for the factor of ($\mathcal{W}/K$):

$$\tilde{b}_j=\frac{\mathcal{B}}{K}+\log_2\left(\frac{\sigma_j}{\sigma_g}\right) , \qquad (15)$$

where σ_g is the geometric mean of the standard deviations of the K subbands,

$$\sigma_g\triangleq(\Pi_{i=1}^{K}\sigma_i)^{(1/K)} . \qquad (16)$$

It is important to bear in mind that Eq. (15) is *not* optimum since an approximation has been used in the development. Although Eq. (13) is derived without any approximation, it has an undesired phenomenon in bit allocation as described later.

Refer to Eqs. (13) and (15); note that the first term ($\mathcal{B}/K$) is the average bit rate, which is a positive constant for a given $\mathcal{B}$ and K. Only the logarithmic terms may have negative values and could cause b_j^* to be negative as a result. The multiplicative factor ($\mathcal{W}/K$) of Eq. (13) is a positive number and equal to the pixel depth (usually 8 bits/pixel for monochrome images and 10 bits/pixel for medical images). Therefore, if the logarithmic term is negative, the factor ($\mathcal{W}/K$) further amplifies the negative values and increases the possibility of b_j^* being negative. [Note that given a set of decomposed subbands, the terms $\log_2(\mathcal{P}_j/\mathcal{P}_g)$ or $\log_2(\sigma_j/\sigma_g)$ are fixed regardless of $\mathcal{B}$.] As expected, through simulations, about half of the subbands (on average) have negative bit allocations due to the existence of this factor. In addition, when the AMBTC quantizers are exploited in subbands, the resulting so-called total unassigned bit rate (discussed in Sec. 4.4) is extremely large and unacceptable. Hence, the factor ($\mathcal{W}/K$) causes a higher degree of bit-rate fluctuations and should be discarded for a more stable and robust bit assignment. It is highly important to note that discarding the factor ($\mathcal{W}/K$) from Eq. (13) still meets the fundamental constraint specified by Eq. (8). Indeed, it can be easily shown that $\Sigma_{j=1}^{K} b_j^*=\mathcal{B}$. Note that only the summation of logarithmic terms turns out to be zero.

Based on these justifications, the factor ($\mathcal{W}/K$) of Eq. (13) is discarded and Eq. (13) becomes

$$b_j^*=\frac{\mathcal{B}}{K}+\log_2\left(\frac{\mathcal{P}_j}{\mathcal{P}_g}\right) . \qquad (17)$$

Now Eqs. (17) and (15) have the same direct form except that the Shannon-bound-based algorithm uses the subband mean energy while the standard-deviation-based algorithm uses the subband standard deviation.[‡] It would be quite meaningful and interesting to see how each subband signal characteristic contributes to the bit allocation process and compare their resulting coding performance.

The Shannon-bound and standard-deviation bit allocation algorithms, that is, Eqs. (15) and (17), are referred to as direct bit allocation algorithms for discussion in the following paragraphs.

4.4 *The Sequential Form*

It is important to note that the dynamic bit allocation process does not perform any compression but only serves as a mechanism for assigning the proper number of bits to each subband. Based on the bit allocation results, an appropriate quantizer is then independently applied to an individual subband. Hence, a dynamic bit allocation process must be incorporated with quantizers which may have a fixed bit rate. If the two are mismatched, then poor coding performance may occur. The "mismatch" here means that the results calculated using

[†] Equation (15) should be precisely called a standard-deviation-based bit allqcation. For the variance-based algorithm, Eq. (15) should be rewritten as $b_j=(\mathcal{B}/K)+(1/2)\log_2(\sigma_j^2/\sigma_g^2)$; where $\sigma_g^2\triangleq(\Pi_{i=1}^{K}\sigma_i^2)^{(1/K)}$ is the geometric mean of the variances of the K subbands.

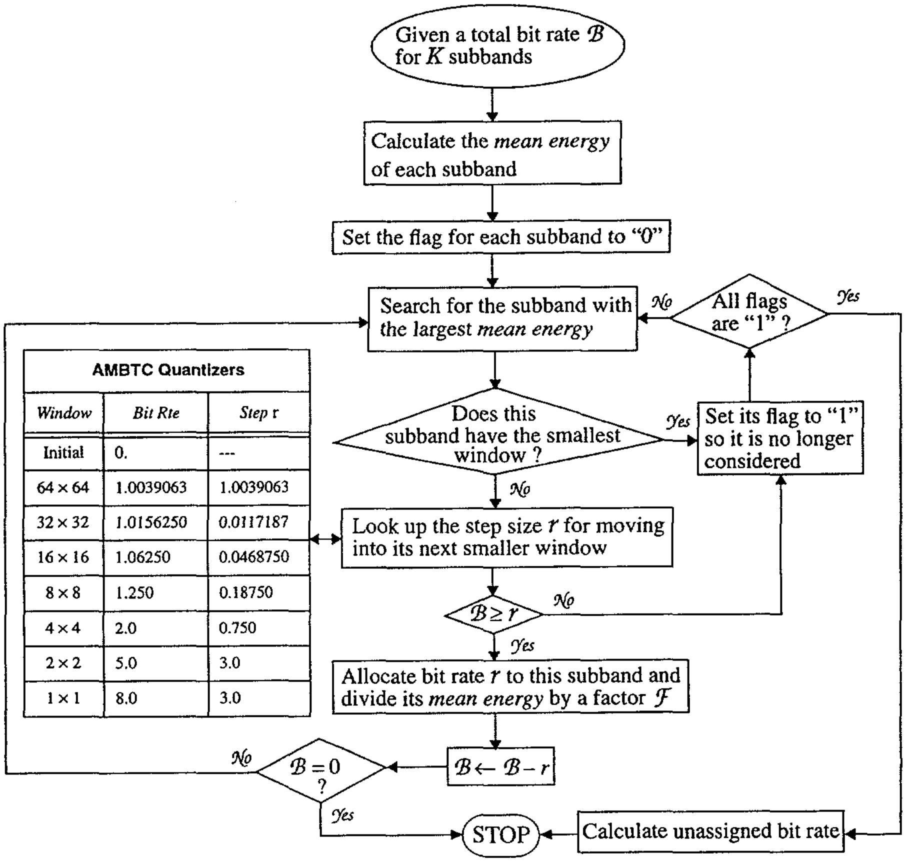
[‡] For a zero-mean source, these two quantities have the same value. However, the mean intensity of each subband image is usually not zero.

Eqs. (15) and (17) (which could be any real numbers) are different from the quantizers' bit rates (which are always positive and possibly constant). In practice, the quantizer with a bit rate closest to but less than the allocated bit rate for the subband is used as its quantizer. Thus, the difference between the two is the unassigned bit rate. The sum of the unassigned bit rates from all the subbands (called the *total unassigned bit rate*) can be quite large, and the resulting image quality is unacceptable. Hence, a new implementation algorithm called *sequential dynamic bit allocation* is developed from the direct form, which is similar to Ramstad's work.[34]

The sequential form is an iterative implementation of the direct form (see Fig. 3). For the sequential form of the standard-deviation-based algorithm, only the subband mean energy in Fig. 3 needs to be changed to the subband standard deviation. The sequential algorithm works as follows. Assume a total bit rate of $\mathcal{B}$ and that each subband initially starts with 0 bits. The total bit rate $\mathcal{B}$ is gradually distributed to the subbands through the iterations. At each iteration, only the subband with the largest signal energy can be assigned a portion of the bit rate. The amount assigned to this subband depends on three factors: (1) its current window size, (2) the required incrementals for changing to the next smaller window, and (3) the remaining total bit rate. If the subband starts with 0 bits, then it will be assigned a number of bits which has a bit rate equivalent to using the largest available window, i.e., window 64 × 64 (see the table of AMBTC quantizers in Fig. 3). For a successful bit rate assignment at each iteration, the window size changes to the next smaller one for the assigned subband; the energy of this subband is divided by a factor $\mathcal{F} = 2^r$ (this condition will be derived later), and the total bit rate $\mathcal{B}$ is reduced by the amount of bit rate assigned at that iteration. Owing to the energy reduction factor $\mathcal{F}$, a subband has multiple chances for bit rate allocations (as described earlier) and is gradually moved from a larger window to a smaller one (i.e., from a coarser quantization to a finer quantization). The iterative bit allocation process ends when either the total bit rate $\mathcal{B}$ is entirely assigned or the remaining unassigned bit rate is too small to let any subband change its current window size to the next smaller one (thus the corresponding subband flag is set for each subband successively).

In Fig. 3, the key parameter which assures the bit allocation is optimum in each sequential step is the reduction factor $\mathcal{F}$, which is derived from the direct form as follows. Assume that subband i has the highest average energy among all the available subbands in any given iteration. If the subband i is assigned an additional bit rate r, the total bit rate is reduced to $(\mathcal{B} - r)$. Thus, the average energy of subband i is divided by a factor $\mathcal{F}$. For the bit allocation in the next iteration, Eq. (17) becomes

$$\hat{b}_j = \frac{\mathcal{B} - r}{K} + \log_2\left(\frac{\mathcal{P}_j}{\hat{\mathcal{P}}_g}\right) \qquad j = 1, \ldots, K , \tag{18}$$


Given a total bit rate $\mathcal{B}$ for K subbands
Calculate the *mean energy* of each subband
Set the flag for each subband to "0"
Search for the subband with the largest *mean energy*
No
All flags are "1" ?
Yes
Does this subband have the smallest window ?
Yes
Set its flag to "1" so it is no longer considered
No
Look up the step size r for moving into its next smaller window
$\mathcal{B} \geq r$
No
Yes
Allocate bit rate r to this subband and divide its *mean energy* by a factor $\mathcal{F}$
$\mathcal{B} \leftarrow \mathcal{B} - r$
No
$\mathcal{B} = 0$?
Yes
STOP
Calculate unassigned bit rate

AMBTC Quantizers		
Window	*Bit Rte*	*Step* r
Initial	0.	---
64 × 64	1.0039063	1.0039063
32 × 32	1.0156250	0.0117187
16 × 16	1.06250	0.0468750
8 × 8	1.250	0.18750
4 × 4	2.0	0.750
2 × 2	5.0	3.0
1 × 1	8.0	3.0

Fig. 3 The flowchart for the Shannon-bound sequential dynamic bit allocation algorithm, where $\mathcal{F} = 2^r$. For the standard-deviation-based algorithm, the term *mean energy* is changed to *standard deviation.*

where

$$\hat{\mathcal{P}}_g = \left[\left(\frac{1}{\mathcal{F}} \right) \Pi_{i=1}^{K} \mathcal{P}_i \right]^{(1/K)} = [\mathcal{F}^{-(1/K)}] \mathcal{P}_g \ .$$

Hence,

$$\hat{b}_j = \frac{\mathcal{B} - r}{K} + \log_2 \left\{ [\mathcal{F}^{(1/K)}] \frac{\mathcal{P}_j}{\mathcal{P}_g} \right\}$$

$$= \frac{\mathcal{B}}{K} - \left\{ \frac{r}{K} - \log_2 [\mathcal{F}^{(1/K)}] \right\} + \log_2 \left(\frac{\mathcal{P}_j}{\mathcal{P}_g} \right) \ . \qquad (19)$$

To maintain the optimum bit allocation in each iteration, the term in braces in Eq. (19) is required to be zero; thus, $\hat{b}_j = b_j^*$ and

$$\mathcal{F} = 2^r \quad \text{(equal bandwidth case)} \ . \qquad (20)$$

To compare the total unassigned bit rates using both direct and sequential forms, simulations (using the "Lena" image) are performed for both Shannon-bound-based and standard deviation-based bit allocation algorithms at various $\mathcal{B}$. The results are documented in Fig. 4. Note that in the sequential form, only in the last iteration might a small amount of unassigned bit rate occur for the total K subbands, and this is much less than that produced in the direct algorithm (see the vertical scales of Fig. 4). Thus the sequential form effectively reduces the total unassigned bit rate when the quantizers have a fixed bit rate.

4.5 *Shannon Bound Versus Standard Deviation*

The coding performance of SAMBTC with the Shannon bound or standard deviation sequential dynamic bit allocation algorithms are compared at various bit rates. Images "Lena" and "House" are used as real data for simulations. Both images are decomposed into 16 equal-bandwidth subbands. The total bit rate $\mathcal{B}$ is varied with a unit integer increment, from 12 to 32 bits/pixel, for 16 subbands (i.e., the average bit rate varies from 0.75 to 2 bits/pixel). The sequential versions of the Shannon-bound and the standard-deviation bit allocation algorithms are applied to both images at each given $\mathcal{B}$. To compare the distribution of windows among the subbands and their sizes, it has been noted that the standard deviation sequential bit allocation tends to use larger windows (i.e., coarser quantization) and more subbands. On the contrary, the Shannon-bound sequential bit allocation tends to select a smaller number of subbands with finer quantization on these bands using smaller windows.

After the bit allocation process is applied and the window size for each subband is determined, quantization using AMBTC is independently performed on each subband. The final compressed image is obtained by combining the compressed subbands through subband reconstruction. Each compressed image $\hat{x}(m,n)$ is then compared against the original source image $x(m,n)$. The PSNR curves of compressed "Lena" and "House" are plotted in Fig. 5. The PSNR is defined as:

$$\text{PSNR} \triangleq 10 \log_{10} \frac{(255)^2}{\text{MSE}} \text{dB} \ , \qquad (21)$$

where $\text{MSE} \triangleq (1/MN) \sum_{m=1}^{M} \sum_{n=1}^{N} [x(m,n) - \hat{x}(m,n)]^2$ per pixel, and M and N are the number of rows and columns in a digital image, respectively. The Shannon-bound sequential dynamic bit allocation consistently outperforms the standard-deviation sequential algorithm. In addition, the reconstructed image quality has been compared at each bit rate. The Shannon bound algorithm is clearly superior.

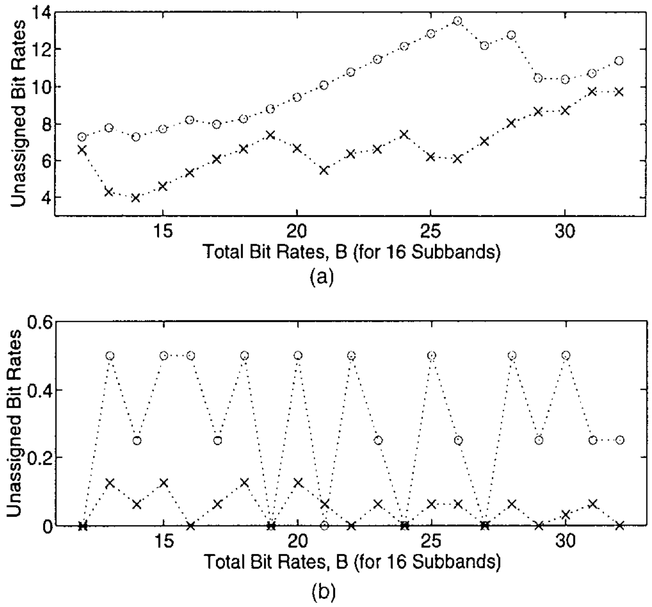


Fig. 4 Comparison of the total unassigned bit rates in (a) the direct form and (b) the sequential form. Each subplot has two curves for Shannon-bound-based (denoted by "○") and standard-deviation-based (denoted by "×") algorithms.

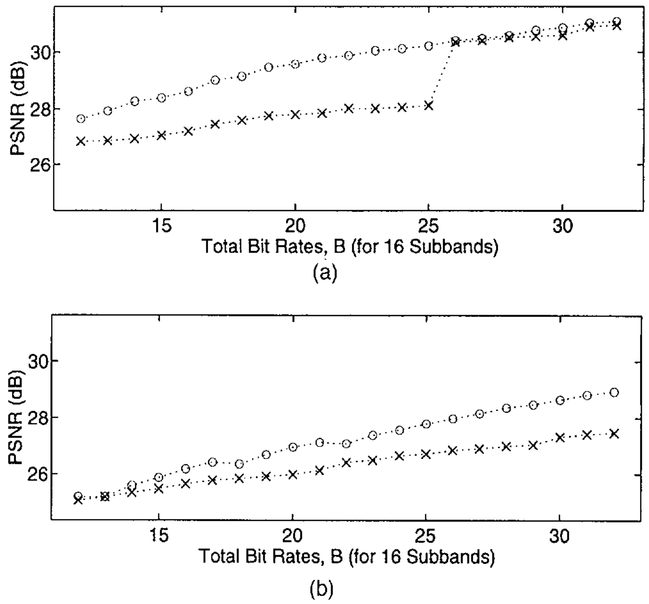


Fig. 5 The peak-to-peak signal-to-noise ratio comparison between the Shannon-bound (denoted by "○") and standard-deviation (denoted by "×") sequential dynamic bit allocation algorithms using images (a) "Lena" and (b) "House".

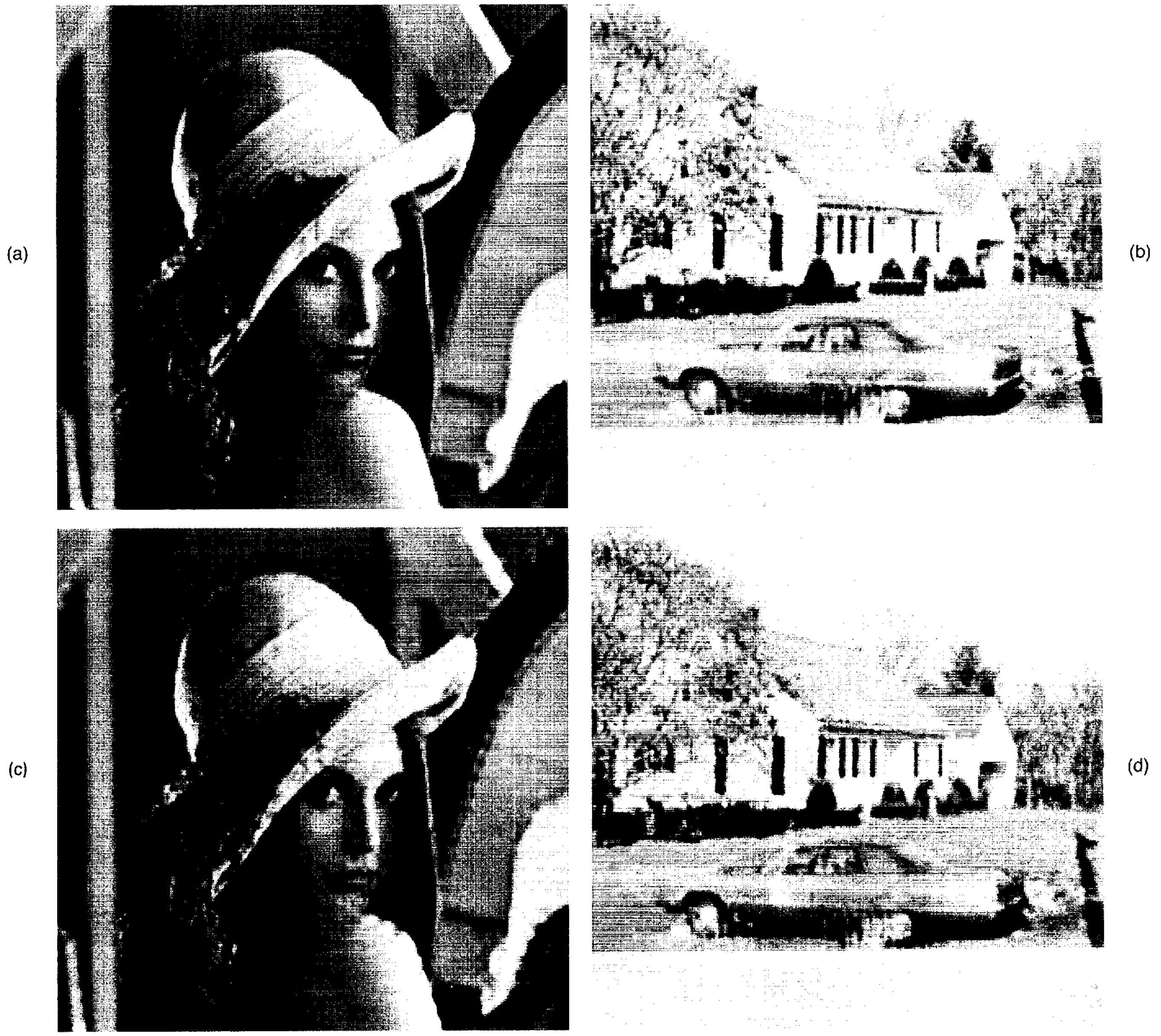


Fig. 6 Comparison between the Shannon bound [(a) and (b)] and the standard deviation [(c) and (d)] dynamic bit allocation algorithms using "Lena" and "House." Each compressed 256×256 "Lena" image has 1.5625 bits/pixel ($=\mathcal{B}/b=25/16$).

To illustrate, a set of compressed "Lena" and "House" images are presented in Fig. 6, for both algorithms at 1.5625 bits/pixel (i.e., $\mathcal{B}=25$). Their bit allocation results in terms of the AMBTC window assignment are listed in Table 2 where the numbers (1 through 16) in the table heading are the subband labels. For the subband labeling, see Fig. 7 where subband 1 represents the baseband. The number in the entry of Table 2 is the chosen window size for that band. For example, an 8×8 AMBTC window was used to compress subband 5 in image (a). For the chosen AMBTC windows and their corresponding bit rate see the table in Fig. 3. In addition, we used 0×0 and 1×1 to denote "discard" and "no compression," respectively.

Based on the simulation results shown in Fig. 4, it is interesting to note that on average the standard deviation algorithm has a lower unassigned bit rate than that using the Shannon bound algorithm. From this standpoint, the standard deviation algorithm is more efficient than the Shannon bound algorithm in allocating bits among subbands and is expected to perform better. On the contrary, the signal characteristic used in these two algorithms makes the bit allocation results

6	8	14	16
5	7	13	15
2	4	10	12
1	3	9	11

Fig. 7 A notation for 16 equal-bandwidth subbands.

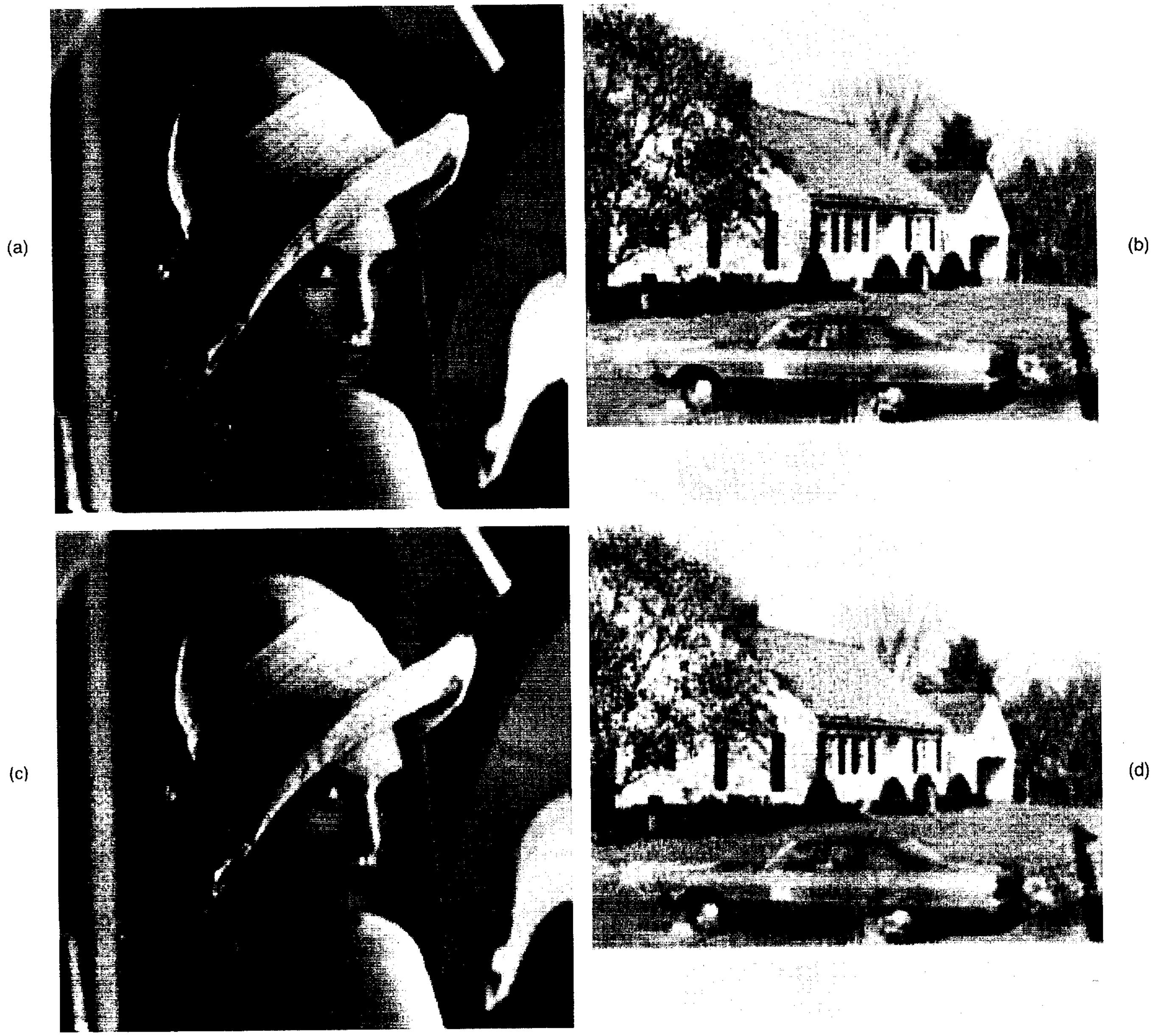


Fig. 8 The compressed 256×256 "Lena" and "House" images using SAMBTC with the Shannon-bound sequential dynamic bit allocation algorithm at 1.250 bits/pixel in (a) and (b); and at 1.015625 bits/pixel in (c) and (d).

quite different and causes the Shannon bound algorithm to have superior imagery at low bit rates. This reveals that the subband mean energy is more effective for use in the bit allocation process than the subband standard deviation (or variance) in SAMBTC. This might also be true for other subband coding algorithms and/or signal sources (e.g., speech signal).

Table 2 Subband bit allocation results in terms of the AMBTC window assignment for the images in Fig. 6. For each case, from (a) to (d), the total unassigned bit rate is 0.0313, 0.0039, 0.0039, and 0.0 bits/pixel respectively.

Fig.6	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
(a)	1	2	4	2	8	4	0	8	0	0	0	0	0	0	0	0
(b)	1	2	2	8	16	8	0	16	16	0	8	0	0	0	0	0
(c)	2	2	4	4	8	4	16	4	0	0	16	8	0	0	16	8
(d)	2	2	4	8	8	4	32	16	16	32	8	16	0	0	32	32

5 Coding Performance

5.1 *SAMBTC Versus AMBTC*

Reconstructed "Lena" and "House" images using the SAMBTC approach with the Shannon-bound-based sequential dynamic bit allocation are presented in Fig. 8. Comparing Figs. 8 and 2, SAMBTC clearly outperforms the AMBTC in every case by effectively reducing the blocking artifact and significantly improving the image quality.

The corresponding window assignments for the subbands are listed in Table 3. Performance comparisons in terms of the MSE and PSNR are shown in Table 4 for both AMBTC

Table 3 Window assignment for the 16 equal-bandwidth subbands of (a) "Lena" and (b) "House" using the Shannon-bound sequential dynamic bit allocation algorithm. For the total bit rate $\mathcal{B}$ at 32, 20, 17, and 16.25 bits/pixel, the corresponding total unassigned bit rate is 0.0156, 0.0313, 0.0156, and 0.0625 bit/pixel for "Lena" and 0.0078, 0.0, 0.0, and 0.0 bit/pixel for "House," respectively.

(a). *LENA*		Shannon-Bound Sequential Bit Allocation for the 16 Equal-Bandwidth Subbands										
Bit Rate, (bpp)												
$\mathcal{B}$	$b = \mathcal{B}/16$	1	2	3	4	5	6	7	8	9↦11	12	13↦16
32	2.0	1	2	2	2	8	2	0	8	0	8	0
20	1.25	1	2	4	4	0	8	0	8	0	0	0
17	1.0625	1	2	8	8	0	8	0	0	0	0	0
16.25	1.015625	1	2	16	16	0	16	0	0	0	0	0

(b). *HOUSE*		Shannon-Bound Sequential Bit Allocation for the 16 Equal-Bandwidth Subbands										
Bit Rate, (bpp)												
$\mathcal{B}$	$b = \mathcal{B}/16$	1	2	3	4↦6	7	8,9	10	11	12	13↦15	16
32	2.0	1	2	2	4	16	8	0	4	8	0	16
20	1.25	1	2	4	8	0	0	0	8	0	0	0
17	1.0625	1	4	4	8	0	0	0	8	0	0	0
16.25	1.015625	1	4	4	16	0	0	0	16	0	0	0

and SAMBTC. Notice that the lower the bit rates, the greater the gain in PSNR by using SAMBTC.

5.2 *SMMSEQ Versus 1-bit MMSEQ*

In Section 3.1, we provided justifications about why AMBTC is the best among the 1-bit quantizers to exploit as the codec. To further demonstrate this, we replaced the AMBTC of the SAMBTC algorithm by the 1-bit MMSEQ and called this *subband MMSEQ* (SMMSEQ). Simulations with results documented in Table 4 for the AMBTC and the SAMBTC are independently conducted for the MMSEQ and the SMMSEQ at the same bit rates.

Comparisons are independently made at each bit rate for (1) full-band case—AMBTC versus MMSEQ and (2) subband case—SAMBTC versus SMMSEQ. The results show that the images are perceptually (almost) unnoticeable in the full-band case and even smaller in the subband counterpart. In terms of the PSNR, the 1-bit MMSEQ only outperforms (on average) the AMBTC by 0.66 dB for "Lena" and 0.42 dB for "House" in the full-band case and by 0.22 dB for "Lena" and 0.26 dB for "House" in the subband case.[28] However, the computational complexity of the 1-bit MMSEQ is about 2 to 7 times that using the AMBTC in both full-band and subband cases.[27,44] In the rest of the paper, the SAMBTC system is extended to compress color images recorded in various color spaces.

6 Color Images and Spaces

In general, an uncompressed color image is commonly represented with 24 bits/pixel—8 bits/pixel for each of the three uncompressed color component images. Two 256×256 color images with 24 bits/pixel, "Girl" and "Doll" [see Figs. 9(a) and 9(b) in Color Plate 1, respectively], are used for the simulations in this paper. The "Girl" includes a facial image with flesh tones, highly important for television and visual communication. The "Doll" contains highly saturated areas. Together, these two images cover a wide range of chrominance information and have been recorded in the Commission Internationale de l'Eclairage (CIE) standard observer, *XYZ* color space.[45–47] Usually, a color space transformation from the *XYZ* to another space is performed before compression begins.

Table 4 Comparison between the SAMBTC and the full-band AMBTC at various bit rates using (a) "Lena" and (b) "House." In the full-band AMBTC case, the bit rate *b* corresponds to use of 4×4, 8×8, 16×16, and 32×32 windows, respectively.

(a). *LENA*		AMBTC		SAMBTC	
Bit Rate, (bpp)					
$\mathcal{B}$	$b = \mathcal{B}/16$	*MSE (/pel)*	*PSNR (dB)*	*MSE (/pel)*	*PSNR (dB)*
32	2.0	63.941	30.073	50.582	31.091
20	1.25	134.931	26.830	71.742	29.573
17	1.0625	229.466	24.524	81.768	29.005
16.25	1.015625	374.059	22.401	87.720	28.700

(b). *HOUSE*		AMBTC		SAMBTC	
Bit Rate, (bpp)					
$\mathcal{B}$	$b = \mathcal{B}/16$	*MSE (/pel)*	*PSNR (dB)*	*MSE (/pel)*	*PSNR (dB)*
32	2.0	90.528	28.563	82.994	28.940
20	1.25	168.017	25.877	130.873	26.962
17	1.0625	251.807	24.120	148.456	26.415
16.25	1.015625	409.908	22.004	155.042	26.226

The impact of applying SAMBTC in two different color spaces is investigated. Two important categories of color spaces are considered: (1) television color spaces (i.e., *YIQ* and *YUV*), through linear transformation from *XYZ*, and (2) perceptually uniform color spaces (i.e., $L^*u^*v^*$ and $L^*a^*b^*$), through nonlinear transformation from *XYZ*.[45–47] The main objective of the transformation is to reduce the covariances among the three component images, and the energy and variance distributions are more compact (e.g., see Table 5 for "Girl").

Note that the *Y* components in *YIQ*, *YUV*, and *XYZ* are all identical where *Y* is the luminance. The *I* and *Q* in the *YIQ* space and the *U* and *V* in the *YUV* space are the chrominance signals. The coordinates of (I,Q) and (U,V) have a 33-deg phase difference. As a result, the coding performance conducted on these spaces is fairly close judging from the perspectives of imaging quality and compressibility. The major difference is their transmission format. Therefore, to demonstrate the coding performance of the SAMBTC in category (1), *YIQ* is chosen for simulations. For category (2), $L^*u^*v^*$ is used to represent color images because the $L^*u^*v^*$ space has been used in television while the $L^*a^*b^*$ space is used by the colorant industries (Ref. 47, p. 69 and Ref. 48, p. 19).

To view a color image recorded in any color space, the image must be transformed into *XYZ* space first, followed by linear transformation from *XYZ* to *RGB* space along with proper gamma corrections. Denote this space as display RGB in order to distinguish it from the transmission *RGB* space.

7 Observations on Applying AMBTC to Color Images

7.1 *AMBTC Bit Planes for Color Images*

Recall that any compressed monochrome image using the AMBTC method generates at most two quantization levels for each block in the image. However, this is often not true for color images. When AMBTC is applied to a block in a color image, each color component of this block independently generates its own bit plane. Thus, each color block is represented by more than two quantization levels provided that the bit planes from the three component images at the same block position are not identical.

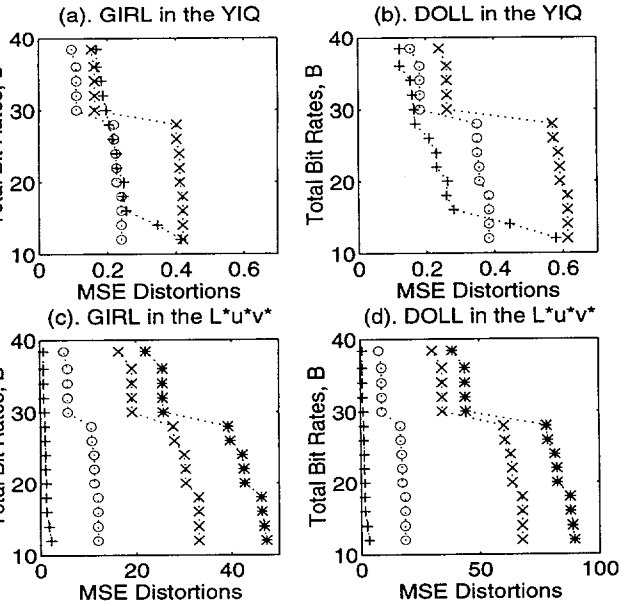


Fig. 10 The rate distortion curves for "Girl" and "Doll" in the *YIQ* [(a) and (b)] and in the *L*u*v** [(c) and (d)], respectively. In (a) and (b), legends "+", "×", and "o" denote the MSE of components *Y*, *I*, and *Q*, individually. In (c) and (d), legends "+", "×", and "o" denote the MSE of components *L**, *u**, and *v**, individually. The additional legend "*" denotes the total MSE, ΔE_{uv}^{*2}.

Consider, for example, a 2×2 AMBTC window applied to a color image represented in $L^*u^*v^*$ space. Assume the three bit planes of components L^*, u^*, and v^* at a particular block position are

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{and} \quad \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix},$$

respectively. The 0's and 1's in each bit plane correspond to the two quantization levels $\mathbf{a}_{L^*}$ and $\mathbf{b}_{L^*}$ for component L^*, $\mathbf{a}_{u^*}$ and $\mathbf{b}_{u^*}$ for component u^*, and $\mathbf{a}_{v^*}$ and $\mathbf{b}_{v^*}$ for component v^*. Thus, the three component blocks of the decoded image at this particular position are:

$$\begin{bmatrix} \mathbf{b}_{L^*} & \mathbf{a}_{L^*} \\ \mathbf{b}_{L^*} & \mathbf{b}_{L^*} \end{bmatrix}, \quad \begin{bmatrix} \mathbf{b}_{u^*} & \mathbf{a}_{u^*} \\ \mathbf{a}_{u^*} & \mathbf{b}_{u^*} \end{bmatrix}, \quad \text{and} \quad \begin{bmatrix} \mathbf{a}_{v^*} & \mathbf{a}_{v^*} \\ \mathbf{b}_{v^*} & \mathbf{b}_{v^*} \end{bmatrix}.$$

In order to display the block on a CRT monitor, these three component blocks must be converted into display *RGB* with gamma correction. Let the combination of the transformations be denoted as $\mathbf{T}(\cdot)$. Then, this block in display *RGB* space is expressed as:

$$\begin{bmatrix} \mathbf{T}(\mathbf{b}_{L^*},\mathbf{b}_{u^*},\mathbf{a}_{v^*}) & \mathbf{T}(\mathbf{a}_{L^*},\mathbf{a}_{u^*},\mathbf{a}_{v^*}) \\ \mathbf{T}(\mathbf{b}_{L^*},\mathbf{a}_{u^*},\mathbf{b}_{v^*}) & \mathbf{T}(\mathbf{b}_{L^*},\mathbf{b}_{u^*},\mathbf{b}_{v^*}) \end{bmatrix}.$$

In this example, each pixel may display a different color since all the arguments of $\mathbf{T}(\cdot)$ are different.

7.2 AMBTC Color Blocking Artifacts

Recall that the full-band AMBTC introduced blocking artifacts in monochrome images at low bit rates. For color images, two aspects regarding the severeness of blocking artifacts have been studied as follows.

Table 5 Covariance matrices, energy distribution, and variance distribution in various color spaces using the color image "Girl."

Space	Covariance Matrix			Average Energy		Variance	
X	1.0000	0.9964	0.9231	349.6890	33.07%	179.0159	30.45%
Y	0.9964	1.0000	0.9347	405.3735	38.33%	220.1113	37.44%
Z	0.9231	0.9347	1.0000	302.5155	28.60%	188.7695	32.11%
R	1.0000	0.9657	0.8513	414.4769	38.34%	200.4518	33.77%
G	0.9657	1.0000	0.9256	459.8803	42.55%	262.6112	44.25%
B	0.8513	0.9256	1.0000	206.6066	19.11%	130.4380	21.98%
Y	1.0000	0.1409	-0.8608	405.5250	94.68%	220.1896	93.14%
I	0.1409	1.0000	-0.2087	13.7166	3.20%	9.4703	4.01%
Q	-0.8608	-0.2087	1.0000	9.0893	2.12%	6.7487	2.85%
Y	1.0000	-0.6999	-0.3210	405.5289	94.67%	220.1923	93.13%
U	-0.6999	1.0000	-0.2387	14.9182	3.48%	9.1111	3.85%
V	-0.3210	-0.2387	1.0000	7.9259	1.85%	7.1310	3.02%
L^*	1.0000	0.0540	0.4278	3899.3662	77.38%	1262.6151	63.10%
u^*	0.0540	1.0000	0.5443	689.6266	13.69%	424.3394	21.20%
v^*	0.4278	0.5443	1.0000	449.9100	8.93%	313.9520	15.70%
L^*	1.0000	-0.1752	0.3834	3899.3662	87.13%	1262.6151	76.38%
a^*	-0.1752	1.0000	0.3070	213.9459	4.78%	148.5258	8.98%
b^*	0.3834	0.3070	1.0000	361.9387	8.09%	241.9921	14.64%

7.2.1 *Blocking artifacts in* YIQ *and* L*u*v*

AMBTC windows with size $2^i \times 2^i$ for $i = 1,2,\ldots,6$ are chosen for the simulations. For each size, the full-band AMBTC is applied to each component image of "Girl" and "Doll" represented in the *YIQ* and $L^*u^*v^*$ color spaces. The compressed component images are then transformed into the display *RGB* space with gamma correction. The simulation results show that the severeness of blocking artifacts in the *YIQ* and $L^*u^*v^*$ color spaces is about the same. In particular, the difference of the color blocking artifacts presented in the *YIQ* and $L^*u^*v^*$ spaces is not noticeable when using a window size smaller than or equal to 16×16. For window sizes greater than 16×16, the resulting image difference between these two spaces is quite small.

To demonstrate the blocking artifacts exploiting the AMBTC algorithm at low bit rates, the compressed images using a 32×32 AMBTC window to "Girl" and "Doll" in the $L^*u^*v^*$ space are presented in Figs. 9(c) and 9(d) in Color Plate 1, respectively.

7.2.2 *Blocking artifacts within the color components*

To investigate the severeness of the blocking artifacts present in each component image, a set of five compressed images in the $L^*u^*v^*$ color space is generated for each AMBTC window (from 2×2 to 64×64). For example, using a 32×32 AMBTC window, five images are generated as follows: (1) A 32×32 window is independently applied to all three component images, i.e., L^*, u^*, and v^*; (2) a 32×32 window is applied to the luminance L^* only; (3) a 32×32 window is separately applied to the two chrominance signals u^* and v^*; (4) a 32×32 window is applied to the chrominance u^* only; and (5) a 32×32 window is applied to the chrominance v^* only. The compressed images have a bit rate of 3.046875 bits/pixel for experiment (1), 17.015625 bits/pixel for experiments (2), (4), and (5), and 10.03125 bits/pixel for experiment (3). Although the resulting bit rates are not identical, this is not a concern in this particular experiment. The blocking artifacts due to experiment (1) are shown in Figs. 9(c) and 9(d). Comparing the compressed images of experiments

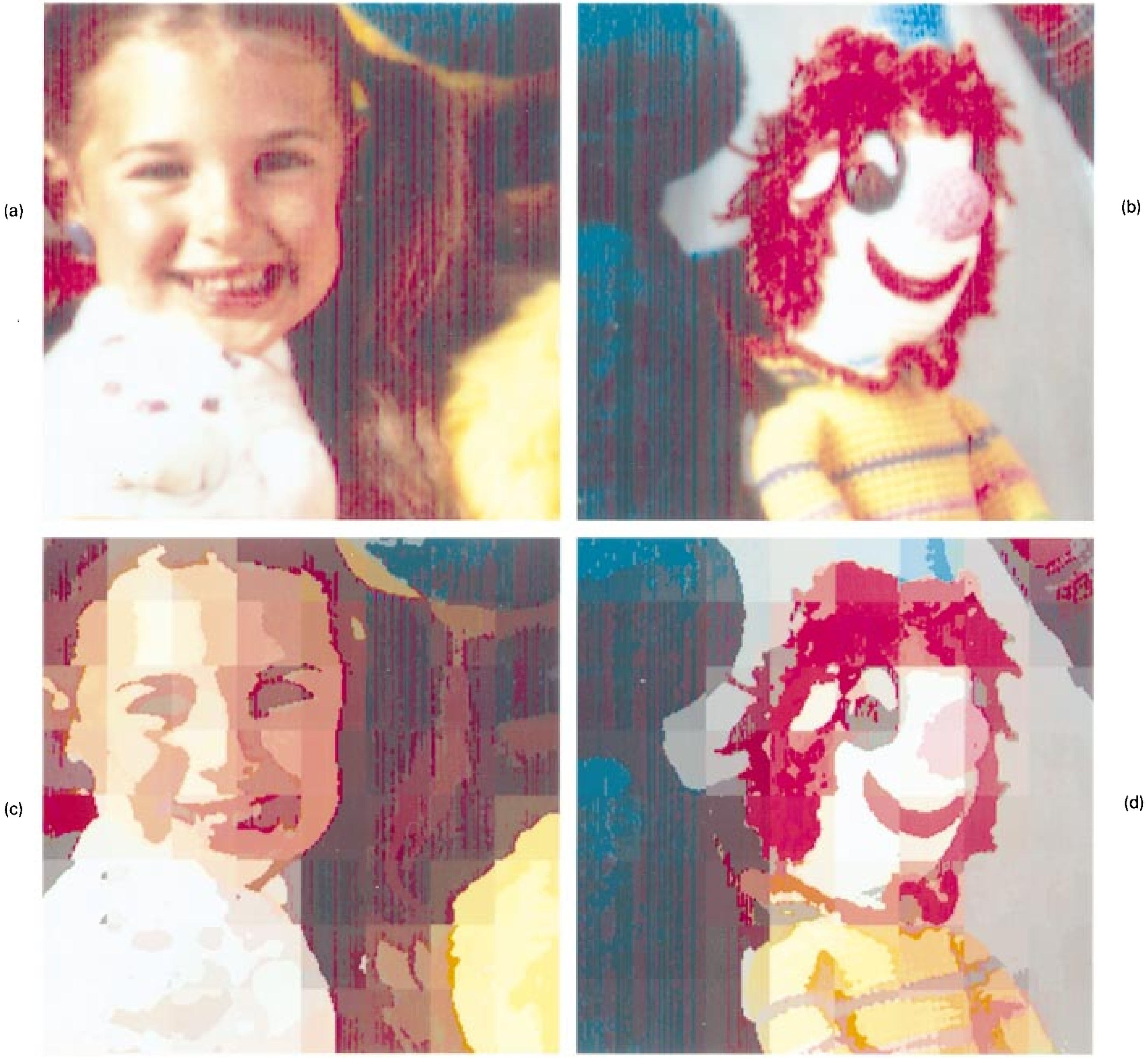


Fig. 9 Each color image is 256×256 with 24 bits/pixel, i.e., 8 for each color component: (a) original "Girl," (b) original "Doll," and (c) and (d) blocking artifacts of "Girl" and "Doll," respectively, in the $L^*u^*v^*$ color space using a 32×32 AMBTC window on each color component plane.

Color Plate 1

(1) and (2), they are perceptually identical. On the other hand, the compressed images from experiments (3), (4), and (5) are almost identical to the original images in Figs. 9(a) and 9(b).

The identical experiments described above were also applied to color images recorded in the *YIQ* space. The same conclusion is also drawn. This indicates that the luminance has much more severe blocking artifacts than the chrominance at the same bit rate.

8 Dynamic Bit Allocation for Color Images

Unlike the monochrome image, each color image has three color component images. Since each component image is decomposed into 16 equal-bandwidth subbands in this paper, a given total bit rate $\mathcal{B}$ must be distributed among the 48 subbands. Thus, the Shannon-bound sequential dynamic bit allocation algorithm developed for monochrome images earlier is exploited with two additional algorithms: (1) quota-ratio division and (2) intercomponent unassigned bit rate reduction.

8.1 *Quota-Ratio Division*

Let a quota ratio be denoted as $(q_1:q_2:q_3)$. The numbers q_1, q_2, and q_3 correspond to the proportion of the total bit rate $\mathcal{B}$ allocated to components L^*, u^* and v^* in the $L^*u^*v^*$ space (or for components *Y, I,* and *Q* in the *YIQ* space), respectively. The coding performance of the Shannon bound sequential dynamic bit allocation algorithm is evaluated with or without a quota ratio, individually. Without using a quota, simulation results show that the image quality is poor, even at medium to high bit rates.

With a quota $(q_1:q_2:q_3)$, the total bit rate $\mathcal{B}$ is partitioned into three unequal parts. This can be justified from the perceptual and numerical viewpoints. From the perceptual viewpoint, it is well known that the human visual system (HVS) perceives luminance and chrominance differently, and the HVS is much less sensitive to the errors that occur in chrominance than in luminance.[45–47] From a numerical viewpoint, the luminance and chrominance of a color space may have different numerical ranges depending on the chosen color space. For example, component L^* has a fixed range (between 0 and 100) of value while components u^* and v^* can have negative values. Therefore, it is not appropriate to treat the three components in a color space equally.

The HVS-based (4:1:1) ratio[47] [also used by the source input format (SIF) of the Motion Picture Experts Group (MPEG)[49]] is chosen. That is, the total bit rate $\mathcal{B}$ is partitioned into three parts: $\mathcal{B}_{L^*}=(2/3)\mathcal{B}$ and $\mathcal{B}_{u^*}=\mathcal{B}_{v^*}=(1/6)\mathcal{B}$ for the $L^*u^*v^*$ space and $\mathcal{B}_Y=(2/3)\mathcal{B}$ and $\mathcal{B}_I=\mathcal{B}_Q=(1/6)\mathcal{B}$ for the *YIQ* space. Using a (4:1:1) quota, simulation results obtained at various bit rates show that the images are greatly superior to those without quota division.

8.2 *Intercomponent Unassigned Bit Rate Reduction*

For ease of discussion, let us consider the bit allocation in the $L^*u^*v^*$ space. After independently applying the Shannon bound sequential dynamic bit allocation algorithm to each color component plane based on $\mathcal{B}_{L^*}$, $\mathcal{B}_{u^*}$, and $\mathcal{B}_{v^*}$, a small number of total unassigned bit rates (possibly zero) may result at the end of the process on each component image—denoted as ε_{L^*}, ε_{u^*}, and ε_{v^*} for components L^*, u^*, and v^*, respectively. Hence, an additional algorithm called intercomponent unassigned bit rate reduction is developed as follows.

Based on our simulation studies, the severeness of the color blocking artifacts in each color component at the same bit rates is ranked as $v^*<u^*<L^*$ in the $L^*u^*v^*$ space. Thus, the Shannon bound sequential dynamic bit allocation algorithm is initially performed for the component v^* based on $\mathcal{B}_{v^*}$. Any unassigned bit rate ε_{v^*} from component v^* will be unconditionally added to $\mathcal{B}_{u^*}$. The Shannon bound sequential dynamic bit allocation algorithm is then performed on component u^* based on $(\mathcal{B}_{u^*}+\varepsilon_{v^*})$. Similarly, any unassigned bit rate ε_{u^*} from component u^* will be unconditionally allotted to $\mathcal{B}_{L^*}$. Thus, the bit allocation algorithm for component L^* is based on $(\mathcal{B}_{L^*}+\varepsilon_{u^*})$. As a result, the total unassigned bit rate $\varepsilon_{L^*u^*v^*}\,(=\varepsilon_{L^*}+\varepsilon_{u^*}+\varepsilon_{v^*})$ might be reduced further provided that not both ε_{v^*} and ε_{u^*} are zero.

All the simulations have shown that the intercomponent unassigned bit rate algorithm consistently reduces the total unassigned bit rate $\varepsilon_{L^*u^*v^*}$ and thus improves the image quality. For the *YIQ* space, the Shannon bound sequential dynamic bit allocation is performed in the following sequence: *Q, I,* and *Y*. The same arguments and simulations can be applied to the *YIQ* space. The intercomponent unassigned bit rate algorithm is also consistently effective in the *YIQ* space.

Finally, it is important to observe that the quota ratio (4:1:1) may be changed after adding the ε_{u^*} (to $\mathcal{B}_{L^*}$) or ε_{v^*} (to $\mathcal{B}_{u^*}$), assuming that ε_{u^*} or ε_{v^*} is nonzero. However, since the quantities ε_{u^*} and ε_{v^*} are quite small, the actual ratio is still close to (4:1:1).

9 Color Rate-Distortion Performance

9.1 *Simulation*

To conduct extensive experiments, a set of 14 bit rates ranging from 2.4 to 0.75 bits/pixel was chosen for each of the four cases ("Girl" and "Doll" in the *YIQ* and $L^*u^*v^*$ spaces, combined) to evaluate the color rate-distortion performance of the SAMBTC. Given a total bit rate $\mathcal{B}$, a proper AMBTC window is decided upon for each subband based on the bit allocation result. The chosen AMBTC window is independently applied to the subband as its quantizer. After AMBTC decoding, the subband synthesis is performed independently for each color component based on its quantized subbands. The synthesized $\widehat{L^*}$, $\widehat{u^*}$, and $\widehat{v^*}$ are the reconstructed components (with distortions) of the original L^*, u^*, and v^*, respectively.

To display the compressed color image on a CRT monitor, the compressed $\widehat{L^*}$, $\widehat{u^*}$, and $\widehat{v^*}$ are then transformed into the CIE *XYZ* space followed by the conversion from the *XYZ* space to the display *RGB* space with gamma correction. Similarly, the same subband synthesis procedures are also applicable to the compressed subband images for each component in the *YIQ* space.

9.2 *Rate-Distortion Measurement*

The distortion in a compressed monochrome image can be measured in terms of mean-square error and peak-to-peak signal-to-noise ratio. Since a compressed color image involves three compressed component images, the total distortion is thus a function of the distortion in each component. For the *YIQ* space, this function is unknown. Therefore, the total MSE and PSNR measurements cannot be computed. Instead, the resulting rate-distortion curves are presented for each component space in Figs. 10(a) and 10(b) for "Girl" and "Doll," respectively.

On the other hand, it is well known that the color difference between two color images recorded in the CIE perceptually uniform color space can be expressed in terms of their component differences.[45–47] That is, for the $L^*u^*v^*$ space,

$$\begin{aligned}\Delta E^*_{uv} &= [(\widehat{L^*} - L^*)^2 + (\widehat{u^*} - u^*)^2 + (\widehat{v^*} - v^*)^2]^{(1/2)} \\ &= [(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2]^{(1/2)} \ , \qquad (22)\end{aligned}$$

where ΔL^*, Δu^*, and Δv^* are color component differences between two images and independently computed for each component on a pixel-by-pixel basis. We can utilize this formula and take a square on both sides to obtain the MSE measurement, $\Delta E^{*2}_{uv} = (\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2$, in the $L^*u^*v^*$ space. Hence, an additional rate-distortion curve based on the ΔE^{*2}_{uv} is presented in Figs. 10(c) and 10(d) for "Girl" and "Doll," respectively.

Compared with the full-band AMBTC, the simulation results show that the SAMBTC has constantly achieved greatly superior imaging quality and is free from blocking artifacts at low bit rates. In the appendix, AMBTC window assignments for luminance and chrominance subbands are shown for certain bit rates to see the overall trend of the bit allocation results in each case. To illustrate the compressed image quality, two bit rates are chosen. They are 2.4 (i.e., 10:1) and 0.75 bits/pixel (i.e., 32:1). These eight color images are shown in Fig. 11 (see Color Plate 2) for "Girl" and Fig. 12 (see Color Plate 3) for "Doll," in both *YIQ* and $L^*u^*v^*$ spaces. Note that the luminance and chrominance information in these images and all other compressed images between these two bit rates is very well preserved and consistent when viewed on the monitor.

10 Conclusions

In this paper, a new subband coding system called subband absolute moment block truncation coding is presented for digital image compression. The SAMBTC combines full-band AMBTC and subband coding with a new subband dynamic bit allocation scheme that is derived from the Shannon rate-distortion bound.[43] In summary, the proposed SAMBTC has the following system merits: (1) unsymmetrical codec with nonparametric encoding and fast decoding, (2) unbounded bit rate lower bound (while AMBTC is bounded by 1 bit/pixel), (3) robust 1-bit moment-preserving quantizer, (4) unnoticeable blocking artifacts at low bit rates, (5) consistent and stable coding performance, and (6) suitable for progressive transmission and real-time implementation.

The derived Shannon-bound-based bit allocation algorithm indicates that subband mean energy could be more appropriate to use for allocating bits than the subband standard deviation (as used in the standard-deviation-based algorithm.[31] Thus, other subband coding algorithms, transform coding schemes, and/or signal sources might also benefit from the use of the Shannon-bound-based algorithm for bit allocation.

For future work, the SAMBTC system can be incorporated with other lossy and lossless coding schemes to achieve a lower bit rate. In addition, the proposed SAMBTC can be applied to video compression. Recently, Webb and Munson[50] applied an error diffusion technique of digital halftoning to effectively smooth out blocking artifacts. Their findings can be exploited in SAMBTC presented here to further improve imaging quality.

Finally, a VLSI realization of AMBTC codec for video signal processing has been reported in Ref. 30. With attractive performance and system merits, a VLSI implementation of the SAMBTC system for multimedia applications is suggested.

11 Appendix

The following tables show some subband bit allocation results in terms of AMBTC window assignment for the subbands of color images "Girl" and "Doll" (see Fig. 7 for subband labeling). The total unassigned bit rates for each case are fairly small and insignificant (on average, 0.0067 bit/pixel per subband). Therefore, they are not listed here, but are available in Ref. 44.

Bit Rate (*bpp*)		4:1:1	(a). The *Luminance* of GIRL in YIQ								
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_Y =$	Component Y								
48 bands	(⇔ *Ratio*)	$\mathcal{B} \times \left(\frac{2}{3}\right)$	1	2	3	4	5	6	7↦10	11	12↦16
12	0.75 (*32:1*)	8.000	1	0	0	0	0	0	0	0	0
16	1.00 (*24:1*)	10.667	1	4	4	0	0	0	0	0	0
24	1.50 (*16:1*)	16.000	1	4	2	0	0	0	0	0	0
32	2.00 (*12:1*)	21.333	1	2	2	4	0	0	0	0	0
38.4	2.40 (*10:1*)	25.600	1	2	1	4	0	8	0	8	0

Bit Rate (*bpp*)		4:1:1	(b). The *Chrominance* of GIRL in YIQ						
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_I = \mathcal{B}_Q$	Component I				Component Q		
48 bands	(⇔ *Ratio*)	$= \mathcal{B} \times \left(\frac{1}{6}\right)$	1	2	3	4↦16	1	2	3↦16
12	0.75 (*32:1*)	2.000	4	0	0	0	4	0	0
16	1.00 (*24:1*)	2.667	4	0	0	0	4	0	0
24	1.50 (*16:1*)	4.000	4	0	4	0	4	4	0
32	2.00 (*12:1*)	5.333	2	0	0	0	2	0	0
38.4	2.40 (*10:1*)	6.400	2	0	8	0	2	8	0

Bit Rate (*bpp*)		4:1:1	(c). The *Luminance* of DOLL in YIQ								
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_Y =$	Component Y								
48 bands	(⇔ *Ratio*)	$\mathcal{B} \times \left(\frac{2}{3}\right)$	1	2	3	4	5	6	7↦10	11	12↦16
12	0.75 (*32:1*)	8.000	1	0	0	0	0	0	0	0	0
16	1.00 (*24:1*)	10.667	1	4	4	0	0	0	0	0	0
24	1.50 (*16:1*)	16.000	1	4	2	0	0	0	0	0	0
32	2.00 (*12:1*)	21.333	1	2	2	4	0	4	0	0	0
38.4	2.40 (*10:1*)	25.600	1	2	1	4	0	8	0	8	0

Bit Rate (*bpp*)		4:1:1	(d). The *Chrominance* of DOLL in YIQ						
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_I = \mathcal{B}_Q$	Component I				Component Q		
48 bands	(⇔ *Ratio*)	$= \mathcal{B} \times \left(\frac{1}{6}\right)$	1	2	3	4↦16	1	2	3↦16
12	0.75 (*32:1*)	2.000	4	0	0	0	4	0	0
16	1.00 (*24:1*)	2.667	4	0	0	0	4	0	0
24	1.50 (*16:1*)	4.000	4	4	0	0	4	4	0
32	2.00 (*12:1*)	5.333	2	0	0	0	2	0	0
38.4	2.40 (*10:1*)	6.400	2	0	8	0	2	8	0

Bit Rate (*bpp*)		4:1:1	(e). The *Luminance* of GIRL in $L^*u^*v^*$								
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_{L^*} =$	Component L^*								
48 bands	(⇔ *Ratio*)	$\mathcal{B} \times \left(\frac{2}{3}\right)$	1	2	3	4	5	6	7↦10	11	12↦16
12	0.75 (*32:1*)	8.000	1	0	0	0	0	0	0	0	0
16	1.00 (*24:1*)	10.667	1	4	4	0	0	0	0	0	0
24	1.50 (*16:1*)	16.000	1	4	4	0	4	4	0	0	0
32	2.00 (*12:1*)	21.333	1	2	2	0	4	4	0	0	0
38.4	2.40 (*10:1*)	25.600	1	2	2	4	4	4	0	8	0

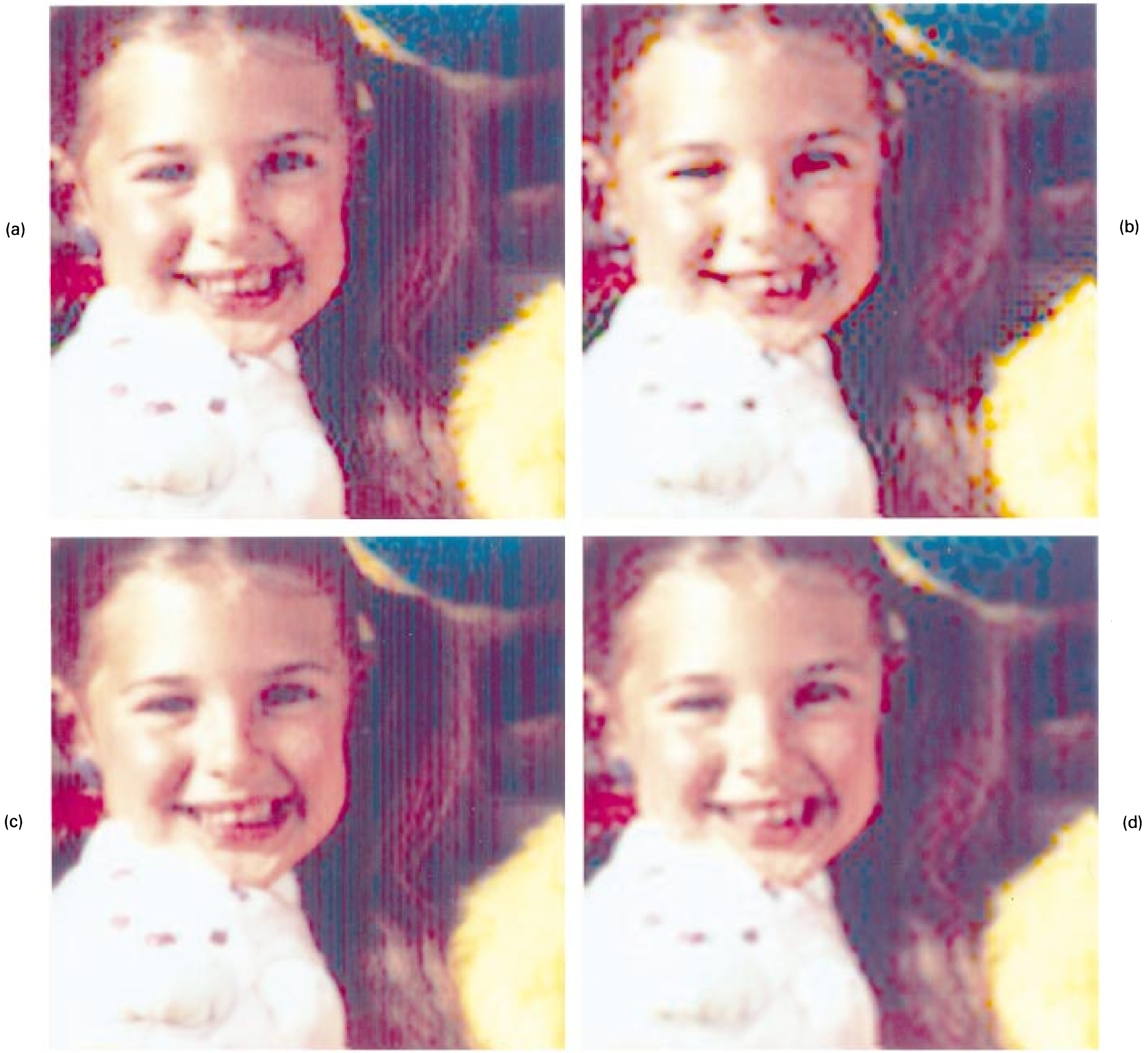


Fig. 11 Compressed "Girl" images: (a) in *YIQ* at 2.4 bits/pixel—10:1, (b) in *YIQ* at 0.75 bits/pixel—32:1, (c) in *L*u*v** at 2.4 bits/pixel—10:1, and (d) in *L*u*v** at 0.75 bits/pixel—32:1. For presentation, all these images were transformed from the *YIQ* or *L*u*v** space into the display RGB space with gamma correction.

Color Plate 2

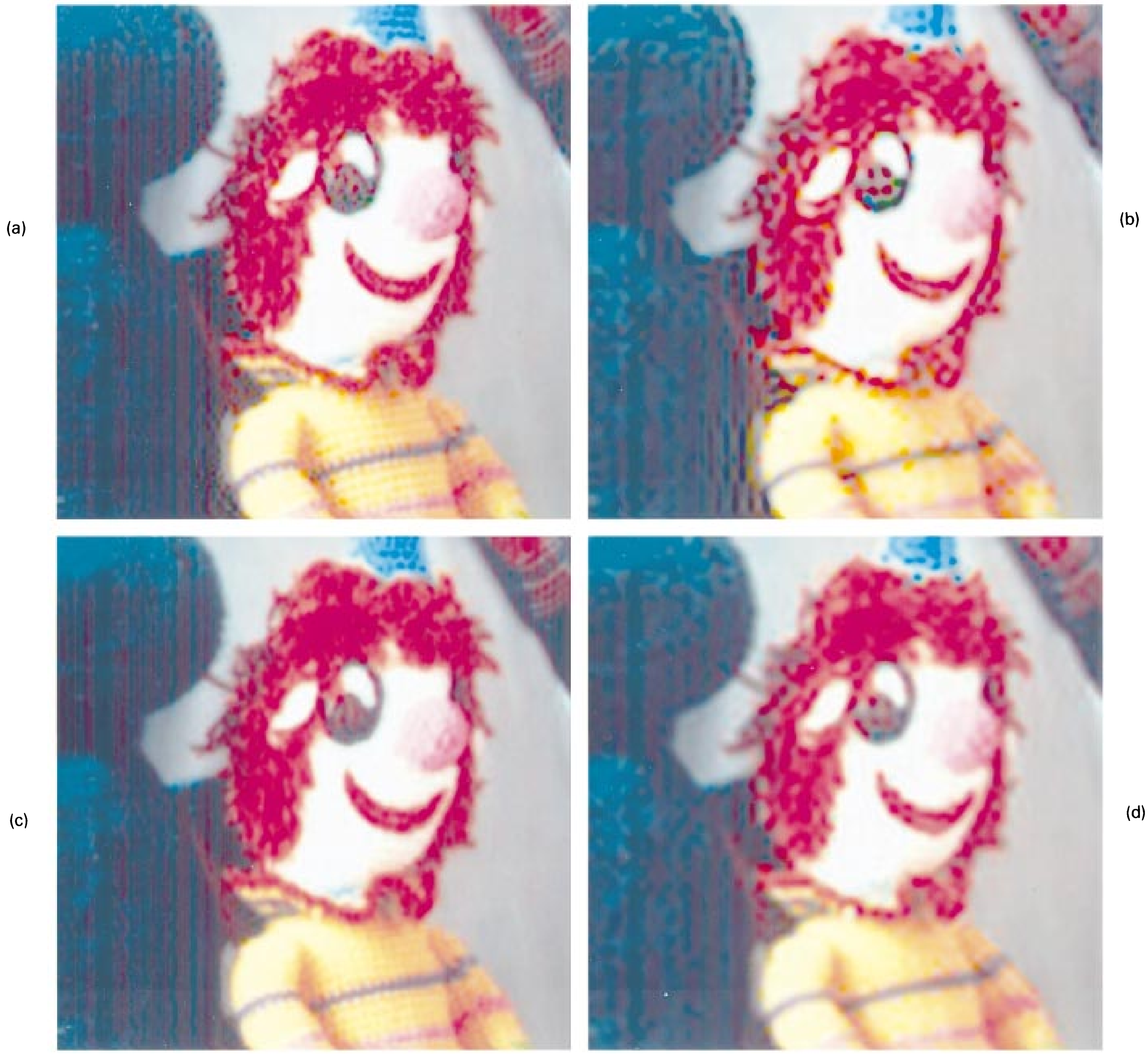


Fig. 12 Compressed "Doll" images: (a) in *YIQ* at 2.4 bits/pixel—10:1, (b) in *YIQ* at 0.75 bits/pixel—32:1, (c) in *L*u*v** at 2.4 bits/pixel—10:1, and (d) in *L*u*v** at 0.75 bits/pixel—32:1. For presentation, all these images were transformed from the *YIQ* or *L*u*v** space into the display RGB space with gamma correction.

Color Plate 3

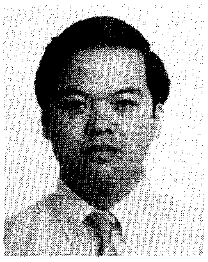
Bit Rate (*bpp*)		4:1:1	(f). The *Chrominance* of GIRL in $L^*u^*v^*$					
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_{u^*} = \mathcal{B}_{v^*}$	Component u^*			Component v^*		
48 bands	(⇔ *Ratio*)	$= \mathcal{B} \times \left(\frac{1}{6}\right)$	1	2	3↦16	1	2	3↦16
12	0.75 (*32:1*)	2.000	4	0	0	4	0	0
16	1.00 (*24:1*)	2.667	4	0	0	4	0	0
24	1.50 (*16:1*)	4.000	4	4	0	4	4	0
32	2.00 (*12:1*)	5.333	2	0	0	2	0	0
38.4	2.40 (*10:1*)	6.400	2	8	0	2	8	0

Bit Rate (*bpp*)		4:1:1	(g). The *Luminance* of DOLL in $L^*u^*v^*$									
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_{L^*} =$	Component L^*									
48 bands	(⇔ *Ratio*)	$\mathcal{B} \times \left(\frac{2}{3}\right)$	1	2	3	4	5	6	7	8↦10	11	12↦16
12	0.75 (*32:1*)	8.000	1	0	0	0	0	0	0	0	0	0
16	1.00 (*24:1*)	10.667	1	4	4	0	0	0	0	0	0	0
24	1.50 (*16:1*)	16.000	1	2	4	0	0	0	0	0	0	0
32	2.00 (*12:1*)	21.333	1	2	2	2	8	8	8	0	0	0
38.4	2.40 (*10:1*)	25.600	1	2	2	4	4	4	0	0	8	0

Bit Rate (*bpp*)		4:1:1	(h). The *Chrominance* of DOLL in $L^*u^*v^*$					
$\mathcal{B}$ for	$b = \left(\frac{\mathcal{B}}{48}\right) \times 3$	$\mathcal{B}_{u^*} = \mathcal{B}_{v^*}$	Component u^*			Component v^*		
48 bands	(⇔ *Ratio*)	$= \mathcal{B} \times \left(\frac{1}{6}\right)$	1	2	3↦16	1	2	3↦16
12	0.75 (*32:1*)	2.000	4	0	0	4	0	0
16	1.00 (*24:1*)	2.667	4	0	0	4	0	0
24	1.50 (*16:1*)	4.000	4	4	0	4	4	0
32	2.00 (*12:1*)	5.333	2	0	0	2	0	0
38.4	2.40 (*10:1*)	6.400	2	8	0	2	8	0

References

1. M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Prentice-Hall, Englewood Cliffs, NJ (1995).
2. P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, NJ (1993).
3. J. W. Woods (ed.), *Subband Image Coding*, Kluwer Academic, Norwell, MA (1991).
4. M. Vetterli, "Multi-dimensional sub-band coding: some theory and algorithms," *Signal Processing* **6,** 97–112, Apr. (1984).
5. J. W. Woods and S. D. O'Neil, "Sub-band coding of images," *IEEE Trans. Acoustics, Speech, and Signal Proc.* **ASSP-34,** 1278–1288, Oct. (1986).
6. H. Gharavi and A. Tabatabai, "Sub-band coding of monochrome and color images," *IEEE Trans. Circuits and Systems* **35,** 207–214, Feb. (1988).
7. P. H. Westerink, J. Biemond, D. E. Boekee, and J. W. Woods, "Subband coding of images using vector quantization," *IEEE Trans. Commun.* **COM-36,** 713–719, June (1988).
8. M. D. Lema and O. R. Mitchell, "Absolute moment block truncation coding and its application to color images," *IEEE Trans. Commun.* **32,** Oct. (1984).
9. N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, Englewood Cliffs, NJ (1984).
10. A. N. Netravali and B. G. Haskell, *Digital Pictures: Representation and Compression*, Plenum Press, New York (1989).
11. M. Rabbani and P. W. Jones, *Digital Image Compression Techniques*, SPIE (The International Society for Optical Engineering) Tutorial Texts, volume TT 7, Bellingham, WA (1991).
12. K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic Press, San Diego, CA (1990).
13. A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic, Norwell, MA (1992).
14. R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ (1983).
15. P. P. Vaidyanathan, "Quadrature mirror filter banks, M-band extensions and perfect-reconstruction techniques," *IEEE ASSP Mag.* **4,** 4–20, July (1987).
16. A. Croisier, D. Esteban, and C. Galand, "Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques," *IEEE Int. Conf. Information Science and Systems*, pp. 443–446 (1976).
17. J. D. Johnston, "A filter family designed for use in quadrature mirror filter banks," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Proc.*, pp. 291–294 (1980).
18. M. J. T. Smith and S. L. Eddins, "Subband coding of images with octave band tree structures," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Proc.*, pp. 1382–1385 (1987).
19. M. Vetterli, "Extension of finite length signals for subband coding," *Signal Processing* **17,** 161–168, June (1989).
20. E. J. Delp and O. R. Mitchell, "Image compression using block truncation coding," *IEEE Trans. Commun.* **COM-27,** 1335–1342, Sep. (1979).
21. O. R. Mitchell, S. C. Bass, E. J. Delp, T. W. Goeddel, and T. S. Huang, "Image Coding for Photo Analysis," *Proc. Society for Information Display* **21/3,** 279–292 (1980).
22. O. R. Mitchell and E. J. Delp, "Multilevel graphics representation using block truncation coding," *Proc. IEEE* **68,** 868–873, July (1980).
23. D. J. Healy and O. R. Mitchell, "Digital video bandwidth compression using block truncation coding," *IEEE Trans. Commun.* **COM-29,** 1809–1817, Dec. (1981).
24. D. R. Halverson, N. C. Griswold, and G. L. Wise, "A generalized block truncation coding algorithm for image compression," *IEEE Trans. Acoustics, Speech, Signal Proc.* **ASSP-32,** 664–668, June (1984).
25. B. V. Dasarathy, *Image Data Compression Block Truncation Coding*, IEEE Computer Society Press, Los Alamitos, CA (1995).
26. E. J. Delp and O. R. Mitchell, "Moment preserving quantization," *IEEE Trans. Commun.* **COM-39,** 1549–1558, Nov. (1991).
27. K.-K. Ma and S. A. Rajala, "New properties of AMBTC," *IEEE Signal Proc. Lett.* **2**(2), 34–37, Feb. (1995).
28. K.-K. Ma and S. A. Rajala, "A comparison of absolute moment block truncation coding and the minimum mean square error quantizer," in *IEEE 1991 Int. Symp. Circuits and Systems*, pp. 296–299.
29. H. H. Ko and C. W. Lee, "Real time implementation of block truncation coding for picture data compression," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Proc.*, pp. 1067–1070 (1987).
30. L.-G. Chen, Y.-C. Liu, T.-D. Chiueh, and Y.-P. Lee, "A real-time video signal processing chip," *IEEE Trans. Consumer Electronics* **39**(2), 82–92, May (1993).
31. J. J. Y. Huang and P. M. Schultheiss, "Block quantization of correlated Gaussian random variables," *IEEE Trans. Commun.* **CS-11,** 289–296, Sep. (1963).
32. J. Makhoul, S. Roucos, and H. Gish, "Vector quantization in speech coding," *Proc. IEEE* **73,** 1551–1588, Nov. (1985).
33. A. Segall, "Bit allocation and encoding for vector sources," *IEEE Trans. Information Theory* **IT-22**(2), 162–169, Mar. (1976).
34. T. Ramstad, "Considerations on quantization and dynamic bit-allocation in subband coders," *Proc. IEEE Int. Conf. on Acoust., Speech and Signal Proc.* pp. 841–844 (1986).
35. B. Fox, "Discrete optimization via marginal analysis," *Management Sci.* **13**(3), 210–216, Nov. (1966).
36. Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, and Signal Proc.*, **36**(9), 1445–1453, Sep. (1988).
37. A. V. Trushkin, "Bit number distribution upon quantization of a multivariate random variable," *Problems of Inform. Transmission* **16,** 76–79 (1980). Translated from Russian.
38. A. V. Trushkin, "Optimal bit allocation algorithm for quantizing a random vector," *Problems of Inform. Transmission,* **17,** 156–161 (1981). Translated from Russian.
39. H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operations Research* **11,** 399–417 (1963).
40. P. H. Westerink, J. Biemond and D. E. Boekee, "An optimal bit allocation algorithm for subband coding," *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Proc.*, pp. 757–760 (1988).
41. G. T. Toussaint, "Pattern recognition and geometrical complexity," in *Proc. IEEE 5th Int. Conf. on Pattern Recognition,* pp. 1324–1347 (1980).
42. L. M. Goodman, "Optimum rate allocation for encoding sets of analog messages," *Proc. IEEE,* **53,** 1776–1777, Nov. (1965).
43. C. E. Shannon, "A mathematical theory of communication," *Bell System Tech. J.* **27,** 379–423, July (1948).
44. K.-K. Ma, "Subband image coding using absolute moment block truncation with a Shannon-bound sequential dynamic bit allocation algorithm," Ph.D. diss., North Carolina State University, Raleigh, NC (1992).
45. G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae,* 2nd ed., Wiley, New York (1982).
46. W. K. Pratt, *Digital Image Processing,* 2nd ed., Wiley, New York (1991).
47. R. W. G. Hunt, *Measuring Colour,* 2nd ed., E. Horwood, New York (1991).
48. W. N. Sproson, *Colour Science in Television and Display Systems,* Adam Hilger, Bristol, England (1983).
49. *Generic Coding of Moving Pictures and Associated Audio,* Recommendation H.262, ISO/IEC 13818-2, IS, Singapore (Nov. 1995).
50. J. L. H. Webb and D. C. Munson, Jr., "Reduced-rate block truncation coding of images using error diffusion," *IEEE Signal Proc. Lett.* **2**(4) 68–69, Apr. (1995).



Kai-Kuang Ma received his BE in electronic engineering from Chung Yuan Christian University (Taiwan) and MS and PhD degrees in electrical engineering from Duke University and North Carolina State University, respectively. Currently, he is a lecturer at the School of Electrical and Electronic Engineering, Nanyang Technological University, Republic of Singapore. From 1992 to early 1995, he was a member of the technical staff at the Institute of Microelectronics (IME), National University of Singapore. At IME, he was responsible for R&D in advanced imaging technologies. From 1984 to 1992, he was with IBM in the United States and worked on various advanced product developments in the areas of PC-based multimedia communications using the IBM digital signal processor MWave, telecommunications, and VLSI. His research interests include subband/wavelet image coding, very low bit rate coding for wireless visual communications, ATM packet video, and neural networks. Dr. Ma is a senior member of IEEE and a member of Sigma Xi and Eta Kappa Nu.

Sarah A. Rajala received her BS in electrical engineering from Michigan Technological University, Houghton, in 1974, and MS and PhD degrees in electrical engineering from Rice University, Houston, Texas, in 1977 and 1979, respectively. In 1979 she joined the Electrical and Computer Engineering Department at North Carolina State University, Raleigh, where she is a professor and director of the Industry/University Center for Advanced Computing and Communication. She is also an adjunct professor in the Radiology Department at the Bowman Gray School of Medicine, Wake Forest University, Winston-Salem, NC. From July 1987 to July 1988, she held a visiting appointment in the School of Electrical Engineering at Purdue University. Her research interests include color image processing and time-varying image analysis with application to image coding/compression, motion estimation, and target acquisition and tracking.