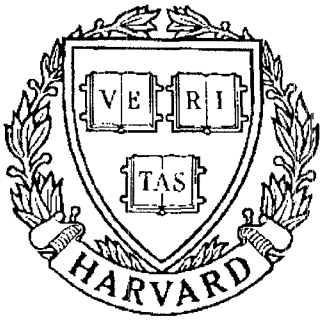
TECHNICAL RESEARCH REPORT

SYSTEMS RESEARCH CENTER



Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

Low Bit-Rate Image Coding Using a Three-Component Image Model

by X. Ran and N. Farvardin

TR 92-75

Low Bit-Rate Image Coding Using a Three-Component Image Model †

Xiaonong Ran and Nariman Farvardin
Electrical Engineering Department
and
Systems Research Center
University of Maryland
College Park, Maryland 20742

Abstract

In this paper the use of a perceptually-motivated image model in the context of image compression is investigated. The model consists of a so-called primary component which contains the strong edge information of the image, a smooth component which represents the background slow-intensity variations and a texture component which contains the textures. The primary component, which is known to be perceptually important, is encoded separately by encoding the intensity and geometric information of the strong edge brim contours. Two alternatives for coding the smooth and texture components are studied: Entropy-coded adaptive DCT and entropy-coded subband coding. It is shown via extensive simulations that the proposed schemes, which can be thought of as a hybrid of waveform coding and feature-based coding techniques, result in both subjective and objective performance improvements over several other image coding schemes and, in particular, over the JPEG continuous-tone image compression standard. These improvements are especially noticeable at low bit rates. Furthermore, it is shown that a perceptual tuning based on the contrast-sensitivity of the human visual system can be used in the DCT-based scheme, which in conjunction with the 3-component model, leads to additional subjective performance improvements.

†This work was supported in part by National Science Foundation grants NSFD MIP-86-57311 and NSFD CDR-85-00108.

I Introduction

The growing interest and need to store and/or transmit digital imagery and video and the practical limitations on the storage and transmission capacity, have led to a significant amount of research activity in image compression (also called image coding) over the past two decades. In any image compression system the goal is that of reproducing a good replica of the original image with as small a number of bits as possible.

In this paper, we limit ourselves to compression of continuous-tone still images. The major thrust in still image compression can be divided into the following two categories: (i) *waveform coding* techniques and (ii) *feature-based* or *second-generation coding* techniques.

Waveform coding techniques revolve around information-theoretic principles in which upon selecting a distortion criterion (in most cases squared-error) and certain probabilistic assumptions on the image data, the goal becomes that of minimizing the average distortion between the original image and its reconstruction under an average bit rate constraint. In these techniques, the human visual system (HVS) and its sensitivity to the coding error generally do not play a central role in system design. Waveform coding techniques, which, by and large constitute the bulk of the research in image coding, have led to a plethora of different coding techniques. Among these, the most noteworthy are: (i) various adaptive forms of two-dimensional (2-D) discrete cosine transform (DCT) coding techniques, (ii) 2-D subband and wavelet image coding techniques and (iii) various forms of vector quantization of images. Several versions of transform-based methods have been proposed in the past two decades, [1] - [4]. The work in DCT coding has recently led to an international still image compression standard known by the acronym JPEG (Joint Photographic Experts Group) [5]. Subband image coding was first introduced by Woods and O'Neil [6] in 1986 and since then has been the subject of much research [7] - [10]. More recently, similar multi-resolution methods using wavelet transforms have received some attention in the context of image coding [11]. Since the pioneering work of Linde, Buzo and Gray [12], vector quantization has been widely studied for image coding both in the spatial domain and in the frequency domain in conjunction with transform or subband coding. A survey of vector quantization methods in image coding can be found in [13]. Additional details on these waveform coding

techniques and various combinations thereof can be found in [14], [15].

Generally speaking, waveform coding techniques provide good-to-excellent quality results at compression ratios of 20:1 - 10:1 (bit rates of 0.4 - 0.8 bits/pixel, assuming 8 bits/pixel for the original). With such compression ratios, the average distortion is typically small and hence the corresponding reconstructed image is of high perceptual quality. At lower bit rates however, waveform coding produces specific types of artifacts such as blockiness, ringing and blurring around the edges, etc. In contrast with waveform coding, feature-based image coding methods are closely tied to the HVS and its separate sensitivities to the strong edge and texture information. Instead of considering the image as a waveform, these methods attempt to describe the image by a collection of physically significant entities such as regions or contours; this leads to a more compact representation of the image and hence significantly higher compression ratios (e.g., 100:1), albeit at the cost of a different type of distortion [16], [17]. The sketch-based image coding of [18], [19] is motivated by similar ideas: Upon extracting the *contours*, their geometric and intensity information are encoded resulting in what is called the *sketch image*; the difference between the original and the sketch image is called the texture and is encoded using waveform coding. This approach can be thought of as an extension of an earlier work by Yan and Sakrison [20]. In the sketch-based scheme [18], the contours are extracted using the Laplacian-Gaussian operator (LGO) [21] followed by a gradient operator. This procedure has two drawbacks: (i) It results in location errors of the estimated edges which, in turn, lead to providing incorrect intensity variations on the two sides of the edges and (ii) it makes the unrealistic assumption that all edges are one pixel wide. These problems can lead to large errors in constructing the sketch image [22].

In this paper, following the spirit of Carlsson's sketch-based method, we adopt an approach based on an amalgamation of feature-based and waveform coding techniques. Specifically, to take into account the properties of the HVS, the image signal is first decomposed into components of distinctive significance to the human perception. This is accomplished by adopting an approach based on a model in which the image is decomposed into the *strong edge*, *texture* and *smooth* components [22]. This approach uses a different methodology for edge extraction in which the above mentioned drawbacks of the LGO are circumvented. The

geometric and intensity information of the strong edges (also called the primary component) is encoded separately. The texture and smooth components are encoded using waveform coding methods. Both entropy-coded adaptive DCT and entropy-coded subband coding are considered. Within this framework, we have demonstrated the advantages of the proposed approach in terms of objective and subjective performance improvements. Furthermore, this study has enabled us to make meaningful comparisons between two rival schemes, namely, adaptive DCT and subband coding, under similar modeling assumptions. In addition, we have developed a modification of our DCT-based scheme in which the well-known contrast sensitivity of the HVS [23] - [25], is used for perceptual tuning of the parameters of the coder. It is shown that the combination of this perceptual tuning and the 3-component based approach further improves the subjective performance of the coder, especially at low bit rates.

The rest of the paper is organized as follows. In Section II, a brief description of the 3-component image model along with some decomposition examples are provided. The encoding of the primary component is described in Section III followed by the description of two encoding schemes for the smooth and texture components in Section IV. Section V includes the description of the overall encoding schemes as well as extensive simulation results and comparisons. Section VI is devoted to the contrast sensitivity of the HVS and how it is used for perceptual tuning of the encoding system. Finally, Section VII contains a summary and conclusions.

II Three-Component Image Model

In this section, we will briefly describe the 3-component image model used in our image coding schemes; the reader is referred to [22] for details.

The block diagram of the system used for generating the three components, namely, the *primary, smooth* and *texture components*, from the original image is illustrated in Fig. 1. The original image is first processed by a space-variant filter whose output is what we call the stressed image. The stressed image is essentially a low-pass version of the original image except near the "strong edges" where the filter acts as an all-pass filter to preserve

the sharpness of these edges. The stressed image, therefore, contains the smooth intensity variations and the strong edge information of the original image [22]. The space-variant filter is implemented using an iterative procedure described by a flow chart in Fig. 2; $\{x_{i,j}\}$ is the original $M \times M$ image, ν is the iteration index and N_ν is the number of iterations before updating of the weighting factors.

The strong edge information is subsequently extracted from the stressed image by identifying pixels of local maximum curvature energy, and is used to generate the primary component of the original image [22] following quantization of the intensity values (and possibly the geometric information) of the strong edge brim contours. The primary component is similar, in concept, to the sketch image of [18]. The quantization and encoding procedure of the strong edge information is described in the next section.

Two examples of the 3-component image model, referred to as the A- and B-configuration, are provided in the following for the 512×512 Lenna. The parameters for generating these two examples (see [22] for the definition of parameters) are $N_\nu = 20$, $T = 64$, $T_c = 32$, $T_\ell = 8$ and $d = 3$ for the A-configuration and $N_\nu = 10$, $T = 64$, $T_c = 32$, $T_\ell = 4$ and $d = 3$ for the B-configuration. The original image Lenna, the stressed images, the strong edge brim contours and the resulting three components for the two configurations are shown in Figs. 3 and 4.[1] In both cases, the primary component is obtained from the quantized strong edge information (see Section III), and the corresponding quantization distortion is included in the smooth component. These distortions can be noticed near the strong edges in Figs. 3 (e) and 4 (e). Note that the A-configuration results in more strong edges than the B-configuration due to its larger N_ν. Generally, larger values of N_ν lead to extracting a larger number of strong edges [26].

III Coding of Primary Component

The primary component of the image is specified by the collection of strong edge brim contours. These contours are defined by their pixel locations and intensity values. More

[1]Since the smooth and texture components can have negative values, for display purposes, we have added a constant to these components to render all pixel values non-negative.

specifically, a strong edge brim contour, denoted by $\bar{b}$, is defined as a sequence of triples:

$$\bar{b} \equiv \{(i^k, j^k, x_{i^k,j^k})\}_{k=0}^{m-1}, \tag{1}$$

where m is the length of the contour, (i^k, j^k) specifies the location of the kth pixel on the contour and x_{i^k,j^k} is the intensity value at (i^k, j^k); the consecutive pixels on the contour are connected and distinct, i.e., $|i^{k-1} - i^k| \leq 1$, $|j^{k-1} - j^k| \leq 1$ and $(i^{k-1}, j^{k-1}) \neq (i^k, j^k)$, $k = 1, \ldots, m-1$, and the variation of the intensity values on the contour is constrained by

$$\max_{k=0,1,\ldots,m-1} |x_{i^k,j^k} - \frac{1}{m} \sum_{\ell=0}^{m-1} x_{i^\ell,j^\ell}| \leq T_c, \tag{2}$$

where $T_c > 0$.

Our objective is to code the contours (location and intensity) with little or no perceptual distortion in reconstructing the primary component. To this end, the sequence of contour pixel locations, $\bar{c} \equiv \{(i^0, j^0), (i^1, j^1), \ldots, (i^{m-1}, j^{m-1})\}$, is encoded using chain coding [27] - [29]. When the well-known Freeman code [27] (also known as the 1-ring code [29]) is used, the sequence is losslessly encoded. To achieve lower bit rates, the N-ring code with $N > 1$ can be used. This procedure results in lossy coding of the contour locations. An example of the 2-ring code is given below.

Consider the sequence $\bar{c}$ of length $m = 7$ shown in Fig. 5, where the pixel locations are marked with solid dots. The 2-ring coding starts at (i^0, j^0). The first intersection of the 2-ring centered at (i^0, j^0) (boundary of the shaded region) with $\bar{c}$ is identified. The intersection point, (i^3, j^3), is encoded with the corresponding index s^1 on the ring (in this case $s^1 = 2$, as indicated in Fig. 5) and the center of the 2-ring is moved to (i^3, j^3). Following the same procedure, a new intersection is identified at (i^5, j^5) which is encoded with index $s^2 = 15$. Then the 2-ring is shifted to pixel (i^5, j^5), but no intersection point is identified; this terminates the encoding operation. The resulting encoded message is $\{(i^0, j^0), 2, 15\}$. The decoding is straightforward. Those points on $\bar{c}$ that correspond to intersection points are correctly reconstructed; the intermediate points are reconstructed by interpolation. For this example, the reconstructed sequence can be either $\{(i^0, j^0), (i^2, j^2), (i^3, j^3), (i^4, j^4), (i^5, j^5)\}$ or $\{(i^0, j^0), (i^2, j^2), (i^3, j^3), (i^4 + 1, j^4), (i^5, j^5)\}$. One of the two possibilities is selected according to a prescribed decoding rule.

Let $\{s^1, s^2, s^3, \ldots\}$ denote the sequence of indices on the N-ring obtained in successive steps of the chain coding operation. Because the contours are generally smooth, it is more efficient to code the differences between the neighboring indices than to code them directly. Therefore, upon defining $r^i = (s^i - s^{i-1}) \bmod 8N$, $i = 2, 3, \ldots$, a code sequence $\{(i^0, j^0), s^1, r^2, r^3, \ldots\}$, is generated where (i^0, j^0) is called the starting location, s^1 the first direction, and $r^2, r^3, \ldots$, the relative directions.

To encode the contour intensity values, the sequence $\{x_{i^0,j^0}, x_{i^1,j^1}, \ldots, x_{i^{m-1},j^{m-1}}\}$ is quantized to a constant sequence $\{\overline{x}, \overline{x}, \ldots, \overline{x}\}$, where $\overline{x} = [m^{-1} \sum_{\ell=0}^{m-1} x_{i^\ell,j^\ell}]$ and $[x]$ denotes the integer closest to x. Simulation results show that this quantization gives rise to little perceptual degradation on edges (for the choices of T_c considered in this work) as compared with the case where the intensity values are losslessly coded.

The sequence $\{\overline{x}, m, (i^0, j^0), s^1, r^2, r^3, \ldots\}$ is used to represent each strong edge brim contour. While $\overline{x}, m$ and (i^0, j^0) are encoded by a fixed number of bits, s^1 and $\{r^i\}$ are encoded separately by means of arithmetic coding [30], [31].

The average rate for encoding the primary component, denoted by r_p bits/pixel, is tabulated in Table 1 for the 512×512 Lenna using the two configurations considered in Section II. The corresponding primary components are shown in Figs. 3 (d) and 4 (d) for the 1-ring code; the results for 2- and 3-ring codes are perceptually indistinguishable and are therefore omitted.

Remark

It is worth noting that the constraint (2) may be changed to a more general one:

$$|x_{i^k,j^k} - \frac{1}{n} \sum_{\ell=0}^{n-1} x_{i^{k-\ell},j^{k-\ell}}| \leq T_c', \tag{3}$$

for $k = 0, 1, \ldots, m-1$, and $T_c' > 0$, where $n = \min\{k+1, n_1\}$ and $n_1 \geq 1$ is a finite window size. With this new constraint, some of the contours obtained from (2) will be connected and thus the coding rate may be decreased due to the reduced number of starting locations to be encoded. However, we need to specify more than one intensity value for some contours with this new constraint. The trade-off between the above two factors is controlled by the threshold T_c' and the window size n_1. After extensive studies of these trade-offs, we

have concluded that the reduction in the bit rate is negligibly small. Hence, the simpler constraint (2) is used for the rest of the work.

IV Coding of Smooth and Texture Components

In this section, we consider the coding of the smooth and texture components. The basic approach is to encode the *sum* of these components using waveform coding techniques. Two different entropy-coded image coding techniques, adaptive (2-D) DCT coding and 2-D subband coding, are used in this work.

A. Adaptive DCT Coding

Let the smooth and texture components of the original image $X = \{x_{i,j}\}$ be denoted by $S = \{s_{i,j}\}$ and $T = \{t_{i,j}\}$, respectively, where $i, j = 0, 1, \ldots, M-1$. Let S and T be segmented into $L \times L$ blocks $\mathbf{s}_{m,n}$ and $\mathbf{t}_{m,n}$, $m, n = 0, 1, \ldots, (M/L) - 1$, respectively, and denote the block of 2-D DCT coefficients of $\mathbf{s}_{m,n}$ and $\mathbf{t}_{m,n}$ by $\bar{\mathbf{s}}_{m,n}$ and $\bar{\mathbf{t}}_{m,n}$, respectively.

The structure of the adaptive DCT coding scheme used here is similar to that in [2]. After computing the 2-D DCT of $\mathbf{u}_{m,n} \equiv (\mathbf{s}_{m,n} + \mathbf{t}_{m,n})$, the blocks $\bar{\mathbf{u}}_{m,n} \equiv (\bar{\mathbf{s}}_{m,n} + \bar{\mathbf{t}}_{m,n})$ are classified into one of a finite number of classes. In [2], the classification is done based on the ac energies of the blocks. As indicated in [32], the classification scheme in [2] neglects the frequency distribution of the ac energy which should be considered for a more efficient coding. In our case, due to the three component model, the low and high frequency energies of a block are already separated and are contained in $\bar{\mathbf{s}}_{m,n}$ and $\bar{\mathbf{t}}_{m,n}$, respectively. Thus, a more efficient classification is possible by using the ac energies of $\bar{\mathbf{s}}_{m,n}$ and $\bar{\mathbf{t}}_{m,n}$. The classification procedure used here consists of two stages. In the first stage, the transform blocks are classified into one of K_1 classes by comparing the ac energies of $\bar{\mathbf{s}}_{m,n}$ against T_k^1, $k = 0, 1, \ldots, K_1 - 2$; the thresholds $\{T_k^1\}$ are chosen such that each of the resulting K_1 classes contains the same number of blocks. In the second stage, each resulting class of the first stage, say k, is further divided into K_2 classes by comparing the ac energies of $\bar{\mathbf{t}}_{m,n}$ against another set of thresholds, $T_{k,\ell}^2$, $\ell = 0, 1, \ldots, K_2 - 2$. These thresholds are also chosen

such that the resulting $K = K_1 K_2$ classes have the same number of blocks in them.[2]

We have conducted a study, similar to that of [33], to determine the approximate distribution of the 2-D DCT coefficients of the different classes by comparing their empirical distributions against the so-called generalized Gaussian distribution [34] using the Kolmogorov-Smirnov test [33]. This is done using a database consisting of a large number of different images. Based on this study, we have concluded that the $(0,0)$th, $(0,1)$th and $(1,0)$th coefficients have an almost Gaussian distribution; all other coefficients (except those that have very small variances and hence are subsequently assigned very small bit rates) are best approximated by a Laplacian distribution. This observation holds for the coefficients in all classes in the case where $K = 4$. From now on, we will use these assumptions for the design of the overall system.

The 2-D DCT coefficients are quantized with uniform-threshold quantizers (UTQs) [34] and subsequently encoded using Huffman codes (HCs). This leads to performance close to that of optimal entropy-constrained scalar quantization [9]. We denote the variance-normalized mean squared error (MSE) associated with the UTQ-HC pair operating at rate r bits/sample for the Gaussian and Laplacian distributions by $d_G(r)$ and $d_L(r)$, respectively. If the variance and the coding rate associated with the (u,v)th coefficient in the kth class are denoted, respectively, by $\sigma_k^2(u,v)$ and $r_{u,v}^k$, the overall MSE is given by [3]

$$D = \frac{1}{L^2}\{\sigma^2(0,0)d_G(r_{0,0}) + \frac{1}{K}\sum_{k=0}^{K-1}[\sigma_k^2(0,1)d_G(r_{0,1}^k) + \sigma_k^2(1,0)d_G(r_{1,0}^k) + \sum_{(u,v)\neq(0,0),(0,1),(1,0)} \sigma_k^2(u,v)d_L(r_{u,v}^k)]\}, \quad (4)$$

where $\sigma^2(0,0)$ and $r_{0,0}$ are the variance and the coding rate, respectively, for the dc coefficient which is encoded in the same manner regardless of the class it belongs to. The average bit rate for encoding the smooth and texture components, r_{st}, is given by

$$r_{st} = r_o + \frac{1}{L^2}\{r_{0,0} + \frac{1}{K}\sum_{k=0}^{K-1}\sum_{u,v\neq(0,0)} r_{u,v}^k\}, \text{ bits/pixel}, \quad (5)$$

[2]The constraint that each class should contain the same number of blocks is not a requirement; it just makes the system simpler to implement. Better performance results can be expected if this constraint is removed.

[3]Throughout this paper, the 2-D DCT used is as defined in [14]. This transformation is unitary and hence the MSE in the spatial domain is the same as that in the transform domain.

where r_o is the bit rate for coding the overhead information. We will elaborate on r_o later.

At this point, it remains to determine the optimal allocation of bit rates among the 2-D DCT coefficients of the different classes. This bit allocation which is the solution of the following constrained minimization problem,

$$\min_{\{r_o, r_{0,0}, r^k_{u,v}\}:\, r_{st} \le r_d} D, \tag{6}$$

where r_d is the design average bit rate, can be obtained efficiently using the steepest descent algorithm described in [35]. To reduce the overhead information for the bit allocation maps and to have a simple system, we have limited $r_{0,0}$ and $\{r^k_{u,v}\}$ to be an integer multiple of 0.1 bits; they are also constrained to be less than 5.0 bits for the Laplacian distribution and 8.0 bits for the Gaussian distribution.

To reconstruct the DCT coefficients at the receiver side, estimates of $\{\sigma^2_k(u,v)\}$ are required ($\sigma^2(0,0)$ is transmitted directly). If the variances of quantization errors of the DCT coefficients, denoted by $\epsilon^2_k(u,v)$, are known, then $\{\sigma^2_k(u,v)\}$ can be estimated from the allocated bit rates $\{r^k_{u,v}\}$. This is because, for $(u,v) \neq (0,0)$, we have

$$\epsilon^2_k(u,v) = \begin{cases} \sigma^2_k(u,v) d_G(r^k_{u,v}), & (u,v) = (0,1) \text{ or } (1,0), \\ \sigma^2_k(u,v) d_L(r^k_{u,v}), & \text{otherwise.} \end{cases} \tag{7}$$

As mentioned before, the performance of the UTQ-HC pairs is very close to that of optimal entropy-constrained scalar quantizers. These quantizers, in turn, have a performance which for high bit rates is about 0.255 bits/sample above the rate-distortion function [34], [36]. Approximating the rate-distortion function by the Shannon Lower Bound [37] and using this observation, we have

$$d_G(r) \approx \frac{\pi e}{6} 2^{-2r}, \qquad d_L(r) \approx \frac{e^2}{6} 2^{-2r}. \tag{8}$$

Substituting (8) into (4) to solve (6), one can easily show that when the bit rates are optimally chosen, the resulting quantization error of the DCT coefficients have the same variances. In the ideal case if the distribution of the coefficients is exactly what we have assumed, the knowledge of this *common* quantization error variance, called the normalization factor and denoted by c, at the receiver, can be used to compute the variances $\{\sigma^2_k(u,v)\}$

using the bit maps (see eq. (7)). In the system actually implemented, the average of those $\epsilon_k^2(u,v)$ corresponding to coefficients with $r_{u,v}^k > 0$, $(u,v) \neq (0,0)$ is used as the value of c.

Before closing this subsection, we should mention that the overhead information that needs to be sent consists of the classification information ($\log_2 K$ bits per block), the mean and variance of the dc coefficient (32 bits for each), the normalization factor c (32 bits) and the bit maps [4] $(2K\lceil\log_2 L\rceil + \lceil\log_2 B\rceil(1 + \sum_{k=0}^{K-1}(u_k v_k - 1)))$. For an $M \times M$ image, this amounts to $r_o = (\log_2 K)/L^2 + (96 + 2K\lceil\log_2 L\rceil + \lceil\log_2 B\rceil(1 + \sum_{k=0}^{K-1}(u_k v_k - 1)))/M^2$, bits/pixel. For example, for encoding the 512×512 Lenna at 0.5 bits/pixel with $L = 16$, $r_o = 0.017$ bits/pixel, $r_p = 0.048$ bits/pixel with 1-ring coding and the A-configuration leaving 0.435 bits/pixel for r_{st}.

B. *Subband Coding*

The entropy-coded 2-D subband coding (SBC) system used here for encoding the sum of the smooth and texture components (hereafter called the smooth-texture component) is identical to that of [9]. The smooth-texture component is analyzed into several narrow-band (or subband) images which are then encoded separately. In the decoder, a replica of the smooth-texture component is synthesized from the decoded subband images. The analysis and synthesis are performed using 2-D separable quadrature mirror filter (QMF) banks. The input image is split into 16 subbands. The reader is referred to [6] for details. All subbands except the lowest frequency subband (LFS) are encoded by means of appropriately designed UTQ-HC pairs; the LFS is encoded by means of a non-adaptive 2-D DCT encoding system in which the DCT coefficients are encoded also by UTQ-HC pairs. The overall system design, the bit allocation procedure and the performance results are reported in [9].

Perhaps we should mention that in the context of SBC, it suffices to have a 2-component model which separates the primary component from the smooth-texture component. This

[4] The integers u_k and v_k are defined to be the smallest integers such that $u \geq u_k$ or $v \geq v_k$ implies $r_{u,v}^k = 0$. Thus, for the kth class, we need to transmit $2\lceil\log_2 L\rceil$ bits for u_k and v_k, and $\lceil\log_2 B\rceil(u_k v_k - 1)$ bits to specify $r_{u,v}^k$, $u < u_k$, $v < v_k$, $(u,v) \neq (0,0)$, where $\lceil x\rceil$ denotes the smallest integer larger than x and B is the number of possible different values of $r_{u,v}^k$. Since $r_{0,0}$ is the same for all classes, the overall number of bits for the bitmaps is $2K\lceil\log_2 L\rceil + \lceil\log_2 B\rceil(1 + \sum_{k=0}^{K-1}(u_k v_k - 1))$.

is because in a SBC system, the input image is separated into subband images based on their frequency contents and therefore the system has the ability to extract the smooth and texture components from the smooth-texture input.

V Simulation Results and Comparisons

In this section we describe the overall structure of the different image coding systems developed based on the 3-component model. We also present simulation results and perform appropriate comparisons. The two main encoding schemes are described next.

A. Description of systems

The ADCT-based scheme

The block diagram of the ADCT-based scheme is illustrated in Fig. 6. The primary component is encoded using the contour coding scheme of Section III. The sum of the smooth and texture components is encoded using the ADCT coding scheme of Subsection IV.A. We will refer to this scheme as the 3C-ADCT-HC scheme (3C for the 3-component model and HC for Huffman coding of the quantized DCT coefficients).

The SBC-based scheme

The block diagram of the SBC-based scheme is illustrated in Fig. 7. The only difference with 3C-ADCT-HC is in the coding of the smooth and texture components which in this case are encoded using the SBC coding scheme of Subsection IV.B. This scheme will be denoted by 3C-SBC-HC.

Other related schemes

For comparison purposes, we will also study the performance of an ADCT-based scheme like that of Subsection IV.A which operates on the original image directly and hence does not utilize the 3-component model; in this scheme the classification method is exactly like

that of [2]. This scheme, referred to as 1C-ADCT-HC, is essentially [5] an entropy-coded version of the system in [2]. Also, we consider the SBC-based coding scheme of [9] (called System B in [9]) directly applied to the image (thus ignoring the 3-component model) and refer to it as 1C-SBC-HC. Finally, we refer to the Huffman coded version of JPEG as JPEG-HC. All JPEG simulation results are obtained using a software package described in [38].

B. *Simulation results*

We have simulated the performance of the above mentioned schemes on the 512×512 Lenna. The reconstructed images at different bit rates are illustrated in Figs. 8-10. For these simulations, in the 1C-ADCT-HC and 3C-ADCT-HC schemes the blocksize used is 16×16; JPEG-HC uses an 8×8 blocksize. In 1C-SBC-HC and 3C-SBC-HC, the lowest frequency subband is encoded using a nonadaptive DCT encoder with blocksize 4×4 as in [9]. In the 3C-ADCT-HC schemes, $K_1 = K_2 = 2$; in the 1C-ADCT-HC schemes, $K = 4$. In these figures, for all 3-component based systems, the A-configuration described in Section II is used as the choice of parameters; 1-ring code is used, but 2- and 3- ring codes are also tried. In some cases, 2- or 3- ring codes yielded slightly better PSNRs. However, in no case did we observe a noticeable perceptual difference.

The peak signal-to-noise-ratio (PSNR) associated with these schemes as well as other schemes to be discussed shortly are summarized in Table 2. In this table, the PSNR results associated with the 3-component model are the best ones obtained among the choices of A- and B- configurations and the 1-, 2- and 3-ring codes.

C. *Discussion*

The simulation results of Figs. 8-10 and their corresponding PSNRs tabulated in Table 2 deserve some discussion. Perhaps the most important observation to be made is that the 3C-ADCT-HC and 1C-ADCT-HC schemes provide the best results. The performance difference

[5] The other differences between 1C-ADCT-HC and the scheme in [2] are the bit assignment algorithm, the assumption on the distribution of the transform coefficients, and the estimation of $\{\sigma_k^2(u,v)\}$ at the receiver side.

between these schemes and JPEG-HC is quite dramatic at low bit rates.

In comparing the subjective performances of 1C-ADCT-HC and 3C-ADCT-HC, it becomes evident that 3C-ADCT-HC performs better near the strong edges; the difference is quite visible at low bit rates. An additional very interesting observation is that even though 3C-ADCT-HC is designed to provide good *perceptual quality* and not to maximize PSNR, it, in fact, results in a PSNR larger than that of 1C-ADCT-HC – a system designed to maximize the PSNR. We will provide some explanation for this behavior later.

The important note to be made in comparing 1C-SBC-HC against 3C-SBC-HC is the significant reduction of "ringing" near the strong edges in 3C-SBC-HC; these ringing effects, which are well-known in subband coding systems, are quite visible in the 1C-SBC-HC scheme especially at low bit rates. While this superior perceptual performance is quite noticeable, the PSNR improvement of 3C-SBC-HC over 1C-SBC-HC is almost negligible.

In spite of the fact that 3C-ADCT-HC and JPEG-HC schemes are both DCT-based systems, the former system exhibits a significantly better performance than the latter. This improvement can be attributed to a combination of the following factors: (i) the method of block classification and DCT coefficient quantization used in 3C-ADCT-HC, (ii) the separate encoding of the primary component in 3C-ADCT-HC and (iii) the larger blocksize used in 3C-ADCT-HC.

To study the importance of these factors separately, we have provided, in Fig. 11, a plot of PSNR vs. bit rate for JPEG-HC with blocksize 8×8 as well as 1C-ADCT-HC and 3C-ADCT-HC with blocksizes 8×8 and 16×16. The important conclusions are as follows. The significance of the larger blocksize is noticeable at low bit rates; at higher bit rates the improvement due to the larger blocksize is vanishingly small. The specific method of block classification and DCT coefficient encoding used in 3C-ADCT-HC is responsible for a significant portion of the PSNR improvement. Additional studies have revealed that the PSNR performance improvement of 3C-ADCT-HC over 1C-ADCT-HC (for the same blocksize) are primarily due to the more efficient 2-stage block classification used in 3C-ADCT-HC and not because of the separate encoding of the primary component. We verified this by simulating a modified version of 3C-ADCT-HC in which the 2-stage classification was replaced by a 1-stage classification similar to that of 1C-ADCT-HC; the PSNR performance of this

modified scheme was even inferior to that of 1C-ADCT-HC. Finally, we should mention that the separate encoding of the primary component is responsible for the improved perceptual quality of 3C-ADCT-HC near the strong edges.

As an alternative to Huffman coding, we have also considered arithmetic coding of the quantized DCT coefficients in all DCT-based schemes considered. The arithmetic coded counterpart of JPEG-HC, 1C-ADCT-HC and 3C-ADCT-HC are denoted, respectively, JPEG-AC, 1C-ADCT-AC and 3C-ADCT-AC. For the theory behind arithmetic coding, its operation and advantages the reader is referred to [30], [31]. The PSNR of the arithmetic coded schemes are also tabulated in Table 2. Clearly, in all cases arithmetic coding performs better than Huffman coding in terms of reduced encoding rate.

In addition to comparisons with JPEG, we make comparisons against a number of other entropy-coded schemes which have been reported in the literature recently. Specifically, we consider the constant block distortion ADCT (CBD-ADCT) of [4], the 2-D adaptive entropy coded SBC (2-D ECSBC) of [10] (System C with arithmetic coding in [10]), the embedded wavelet hierarchical image coder (EWHIC) of [39] and the variable-rate residual vector quantizer (VR-RVQ) of [40]. Since the PSNRs of these schemes are not all reported at the bit rates considered in Table 2, we have plotted the PSNR vs. bit rate performance of all these schemes in Fig. 12. We shall leave the comparisons to the reader.

For the sake of completion, we have repeated our simulations for the 256×256 version of Lenna and generated a similar PSNR vs. bit rate plot in Fig. 13. Where available, the PSNR of other schemes is also included. These PSNR results are also tabulated in Table 3. An important observation here is the noticeable gain obtained by arithmetic coding over Huffman coding obtained in the 256×256 case.

To close this section, we provide analytical PSNR results based on the average distortion formula in (4). These analytical results (in terms of PSNR) are computed and plotted in Fig. 14 for both 1C-ADCT-HC and 3C-ADCT-HC schemes. For the 1C-ADCT-HC scheme, the overall rate is r_{st} (see Eq. (5)) while for 3C-ADCT-HC, it is given by $r_{st} + r_p$. For comparison purposes, we have also plotted the actual simulation results in Fig. 14. The closeness of the analytical and simulation results, especially for the 3C-ADCT-HC case, is quite striking.

VI Perceptual Weighting of Distortions

In this section, we will continue the effort to achieve high perceptual quality for the reconstructed images at low bit rates by adopting an approach based on the *contrast sensitivity* [23] - [25] of the human visual system (HVS). We apply the contrast sensitivity approach to the ADCT-based schemes only; similar ideas have been used in a subband coding framework in [41].

We briefly describe the concept of contrast sensitivity and refer the reader to [23], [24] for details. To determine the contrast sensitivity of the HVS, a test stimulus of the form of a grating pattern is presented to an observer. The grating pattern consists of vertical bars with sinusoidal-intensity scanlines [23]. The sine wave has mean $\overline{m}$, amplitude a and frequency f, defined as the reciprocal of the angle, subtended at the observer's eye, of one complete cycle, in cycles/degree (c/deg). The visibility threshold [24] of the pattern is measured for different choices of $\overline{m}$, a and f. It is shown that for a given spatial frequency f, the visibility threshold of the pattern depends primarily on $a/\overline{m}$, and not separately on a and $\overline{m}$. Upon defining $a/\overline{m}$ as the *contrast* of the grating, the *threshold contrast* is defined as the contrast below which the grating pattern is barely detectable. The reciprocal of the threshold contrast at frequency f is called the *contrast sensitivity* at f and denoted by $m(f)$. The contrast sensitivity curve is measured and presented in [23]. We have reproduced their curve (by reading the coordinates off the graph) and presented it in Fig. 15 (a). The important result is that the contrast sensitivity is very much frequency dependent; it peaks around $2 \sim 4$ c/deg and then rapidly falls off with increasing frequency.

To get a closed-form approximation of the contrast sensitivity curve of Fig. 15 (a), we assume that $\log(m(f))$ is quadratic in $\log(f)$, which implies that $m(f)$ is of the form:

$$m(f) = a \times b^{(\log(\frac{f}{c}))^d}, \tag{9}$$

for some constants a, b, c and d. The optimum (in the sense of minimizing the squared

fitting error for $f > 2$ c/deg [6]) values of these constants are determined using CONSOLE, an optimization-based design tool [42]. These values are: $a = 621.31$, $b = 0.14$, $c = 1.73$ and $d = 1.83$. The original data points along with their least-squares fit are shown in Fig. 15 (b). In the sequel, we make a slight abuse of notation and use $m(f)$ to denote the least-squares fit to the empirical contrast sensitivity curve.

Next we describe how the contrast sensitivity of the HVS can be used for perceptual tuning of the ADCT-based image coding scheme. Consider a generic block of 2-D DCT coefficients, $\{d(u,v);\ u,v = 0,1,\ldots,L-1\}$. Let the error in quantizing $d(u,v)$ be denoted by $e(u,v)$. Then the resulting error in the spatial domain is given by [14]

$$\frac{2}{L}\alpha(u)\alpha(v)e(u,v)\cos\frac{\pi u(2i+1)}{2L}\cos\frac{\pi v(2j+1)}{2L},\quad i,j = 0,1,\ldots,L-1, \tag{10}$$

where $\alpha(0) = 1/\sqrt{2}$ and $\alpha(u) = 1,\ u \neq 0$.

Now consider the simplified case where $d(u,v) = 0$, for all $(u,v) \neq (0,0)$, and $e(u,v) = 0$, for all $(u,v) \neq (0,v_1)$, where $v_1 \neq 0$. Then the corresponding block in the spatial domain is given by

$$\frac{1}{L}(d(0,0) + \sqrt{2}e(0,v_1)\cos\frac{\pi v_1(2j+1)}{2L}),\quad i,j = 0,1,\ldots,L-1, \tag{11}$$

which is exactly of the form of the sinusoidal-intensity grating used as the test stimulus for determining the contrast sensitivities in [23]. By the definition of the contrast sensitivity, the error $e(0,v_1)$ will be essentially undetectable by a human observer if [7]

$$e(0,v_1) \leq \frac{d(0,0) + c_1 L}{\sqrt{2}m(f(v_1))}, \tag{12}$$

where $f(v_1)$ is the spatial frequency of the waveform $\cos\frac{\pi v_1(2j+1)}{2L}$ measured in c/deg, and the constant c_1 is the luminance measurement of the block when $d(u,v) = 0$ for all (u,v).

The above scenario represents an over-simplified case. In practice, $d(u,v)$ are not necessarily all zero for $(u,v) \neq (0,0)$, and $e(u,v)$ are generally non-zero. A complete investigation

[6] On the monitor used to view images in this study, a 16 x 16 block is of size 4 mm by 4 mm. Setting the viewing distance to be 570 mm as in [23], the angle subtended at the observer's eye of a 16 x 16 block is 0.4 degree. Thus, for $L = 16$, the $(0,1)$th and $(1,0)$th 2-D DCT coefficients have a spatial frequency of 2.5 c/deg; other coefficients have higher spatial frequencies. Therefore, in studying $m(f)$, the range $f > 2$ c/deg is adequate for this work.

[7] Note that according to the definition of 2-D DCT in [14], $d(0,0)/L$ is the mean intensity of the block.

of the perceptual effects of the quantization errors for such general cases, requires a study of the superposition of sinusoidal-intensity gratings in both horizontal and vertical directions displayed on non-uniform backgrounds.[8] Such an investigation is more of a psychovisual flavor and is beyond the scope of this paper.

However, the above over-simplified case can be used to provide a guideline for incorporating the contrast sensitivity of the HVS into our ADCT-based image coding system. Specifically, we argue that for good perceptual quality the quantization error $e(u,v)$ should be proportional to $(d(0,0)+c_1L)/m(f(u,v))$, for $(u,v)\neq(0,0)$, where $f(u,v)$ is the spatial frequency in c/deg of the waveform associated with the (u,v)th DCT coefficient. Obviously, such a condition necessitates a different quantization rule for each block. To circumvent this problem, we require, instead, that

(i) $e(u,v)$ be proportional to $\hat{d}(0,0)+c_1L$, where $\hat{d}(0,0)$ is the quantized version of $d(0,0)$, and

(ii) the variance of $e(u,v)$ be inversely proportional to $m^2(f(u,v))$.

In the following, we describe a scheme designed to achieve these two requirements.

Denote the (m,n)th block of 2-D DCT [9] coefficients by $\mathbf{d}_{m,n}$, $m,n=0,1,\ldots,(M/L)-1$, where $\mathbf{d}_{m,n}\equiv\{d_{m,n}(u,v);\ u,v=0,1,\ldots,L-1\}$. We assume that the classification is performed as described in Subsection IV.*A*. The dc coefficients $d_{m,n}(0,0)$ are quantized independently of the classification and the perceptual weighting described below; their quantized versions are denoted by $\hat{d}_{m,n}(0,0)$. To account for the perceptual role of the block luminance, the other 2-D DCT coefficients, $d_{m,n}(u,v)$, $(u,v)\neq(0,0)$, are modified as follows,

$$d'_{m,n}(u,v)=\frac{d_{m,n}(u,v)}{\hat{d}_{m,n}(0,0)+c_1L},\ m,n=0,1,\ldots,(M/L)-1. \tag{13}$$

The variances of these modified 2-D DCT coefficients are denoted by $\sigma'^2_k(u,v)$, $(u,v)\neq(0,0)$, where $k=0,1,\ldots,K-1$, is the classification index. A weighted version of these variances

[8]The findings in [23] suggest the existence, within the nervous system, of *linearly* operating *independent* mechanisms selectively sensitive to limited ranges of spatial frequencies. This observation justifies, to a certain extent, the independent treatment of the quantization error effects of each of the 2-D DCT coefficients.

[9]The 2-D DCT could be performed either on the image itself or on its smooth-texture component.

is used for the bit allocation described below. The quantization is performed on $d'_{m,n}(u,v)$; the quantized version of $d_{m,n}(u,v)$ is obtained from that of $d'_{m,n}(u,v)$ by multiplying back by the scale factor, $(\hat{d}_{m,n}(0,0)+c_1L)$. We denote the quantization errors for $d'_{m,n}(u,v)$ and $d_{m,n}(u,v)$ by $e'_{m,n}(u,v)$ and $e_{m,n}(u,v)$, respectively. Obviously,

$$e_{m,n}(u,v) = (\hat{d}_{m,n}(0,0) + c_1L)e'_{m,n}(u,v). \tag{14}$$

To ensure that quantization error variance for the (u,v)th coefficient is inversely proportional to $m^2(f(u,v))$, we introduce, the weighted variances $\sigma'^2_k(u,v)m^2(f(u,v))$, $k = 0,1,\ldots,\ K-1$, $(u,v) \neq (0,0)$. These weighted variances are subsequently used for bit allocation among all 2-D DCT coefficients of the K classes except the dc coefficient; the number of bits used for the dc coefficient is as given before the perceptual weighting. The rationale behind this weighting of the variances is as follows.

Let us use $\epsilon'^2_k(u,v)$ to denote the variance of $e'_{m,n}(u,v)$ when the (m,n)th block is in class k. Then, using arguments based on the Shannon Lower Bound similar to those given in Subsection IV.A, we have the following approximate relationship:

$$m^2(f(u,v))\epsilon'^2_k(u,v) = \text{const.}, \tag{15}$$

which implies that $\epsilon'^2_k(u,v)$ is inversely proportional to $m^2(f(u,v))$ — exactly what we had set out to achieve.

We have modified the ADCT coding part of both the 1C-ADCT-HC and 3C-ADCT-HC schemes according to the above mentioned perceptual weighting and obtained simulations at different bit rates. The reconstructed images are shown in Fig. 16; figures (a), (c) and (e) illustrate the results for the 3C-ADCT-HC scheme while figures (b), (d) and (f) are for the 1C-ADCT-HC system. In these simulations, we have used $f_k(u,v) = 2.5\sqrt{u^2+v^2}$ [10] and $c_1 = 512$. The choice of c_1 is determined by experimentation.

Comparing the 3C-ADCT-HC results in Figs. 8-10 against those in Fig. 16, it is clearly seen that the perceptually-weighted scheme offers a better subjective performance; the improvement is quite visible at low bit rates. However, for the 1C-ADCT-HC results the

[10] As mentioned before, the $(0,1)$th and $(1,0)$th 2-D DCT coefficients have a spatial frequency of 2.5 c/deg. The choice of $f_k(u,v) = 2.5\sqrt{u^2+v^2}$ guarantees consistency with this observation.

reverse conclusion holds. Specifically, the perceptually-weighted 1C-ADCT-HC scheme suffers from a visible degradation near the strong edges. This phenomenon can be explained as follows. As compared with the straight 1C-ADCT-HC scheme, its perceptually-weighted version results in allocating more bits to the low frequency coefficients and fewer bits to the high frequency coefficients. Therefore, for texture regions, the quantization error generally increases but this has little perceptual effect. In smooth regions, the image is better reproduced and in particular the blockiness is reduced due to the extra bits received for low frequency coefficients. However, in those blocks dominated by strong edges, the perceptual weighting has a detrimental effect: The high frequency components (which are generally strong due to the sharp transitions of the strong edges) are more coarsely quantized and the resulting quantization errors are spread over the entire block in the spatial domain resulting in visible degradation of the uniform regions on both sides of the strong edge. In contrast with the 1C-ADCT-HC scheme, in the 3C-ADCT-HC scheme the strong edges are extracted and encoded separately; the perceptual weighting can thus be applied solely to the smooth and texture components without damaging the integrity of the strong edges.

VII Summary and Conclusions

We have developed a framework for image coding in which a combination of waveform coding and feature-based coding techniques is used to achieve high perceptual quality at low bit rates. The coding schemes revolve around a perceptually motivated 3-component model by means of which the image is decomposed into three components. The primary component, which contains the strong edge information is encoded separately with little or no perceptual distortion. The remaining two components (smooth and texture) are encoded by entropy-coded ADCT or entropy-coded SBC. While the novelty of the proposed schemes reside primarily in the use of the 3-component model and the separate encoding of its constituent components, there are some new elements in the adaptive DCT coding scheme, such as the classification procedure and the estimation of coefficient variances in the receiver, that have contributed to its good objective performance.

Based on extensive simulations and comparisons, we can make the following conclusions:

(i) The proposed encoding schemes based on the 3-component model perform very well at all bit rates with the ADCT-based scheme consistently giving the best results. In all cases, the PSNR of the proposed schemes are significantly better than that of JPEG.

(ii) The main advantage of the 3-component model is a more faithful reconstruction of the strong edge information; the difference is significant at low rates. In the SBC-based schemes, the 3-component model results in a significant reduction of ringing around the strong edges.

(iii) The use of arithmetic coding in both ADCT- and SBC-based schemes results in performance improvements in terms of reduced bit rate; the improvements are more substantial for the 256×256 version.

We have also shown that the well-understood contrast sensitivity of the HVS can be used for perceptual tuning of the ADCT-based encoding scheme. We have developed a method for weighting the 2-D DCT coefficients based on the contrast sensitivity. When this perceptual tuning is used in conjunction with the 3-component model, the subjective performance of the ADCT-based scheme is even further improved. Use of the perceptual weighting without the 3-component model leads to a visible degradation of the strong edges and hence is not desirable.

We should note that both in the ADCT- and SBC-based schemes a very simple and low complexity quantization scheme is used for the DCT coefficients or the subbands. Undoubtedly, use of more powerful quantization methods on the DCT coefficients or the subbands can result in performance improvements, of course, at the cost of some additional complexity. Specifically, it is well-known that the gap between the performance of entropy-coded quantizers used in this paper and the source rate-distortion function is about 1.53 dB. Therefore, other quantization methods that reduce this gap can be used to improve the overall performance. The entropy-constrained trellis coded quantization of Fischer and Wang [43] and the trellis-based scalar-vector quantization of Laroia and Farvardin [44] are two possible candidates. Work on these extensions is currently underway.

References

[1] A. Habibi and P. A. Wintz, "Image coding by linear transformation and block quantization," *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 50-62, Feb. 1971.

[2] W. H. Chen and C. H. Smith, "Adaptive coding of monochrome and color images," *IEEE Trans. Commun.*, vol. COM-25, pp. 1285-1292, November 1977.

[3] W. H. Chen and W. K. Pratt, "Scene adaptive coder," *IEEE Trans. Commun.*, vol. COM-32, pp. 225-232, March 1984.

[4] W. A. Pearlman, "Adaptive cosine transform image coding with constant block distortion," *IEEE Trans. Commun.*, vol. COM-38, pp. 698-703, May 1990.

[5] G. K. Wallace,, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, pp. 30-44, April 1991.

[6] J. W. Woods and S. D. O'Neil, "Subband coding of images," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 1278-1288, October 1986.

[7] H. Gharavi and A. Tabatabai, "Sub-band coding of monochrome and color images," *IEEE Trans. Circuits and Systems*, vol. 35, pp. 207-214, Feb. 1988.

[8] P. H. Westerink, D. E. Boekee, J. Biemond and J. W. Woods, "Subband coding of images using vector quantization," *IEEE Trans. Commun.*, vol. 36, pp. 713-719, June 1988.

[9] N. Tanabe and N. Farvardin, "Subband image coding using entropy-coded quantization over noisy channels," *IEEE Journal of Selected Areas in Communications,* vol. 10, pp. 926-943, June 1992.

[10] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing*, vol. 1, pp. 31-48, January 1992.

[11] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205-220, April 1992.

[12] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84-95, Jan. 1980.

[13] N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: a review," *IEEE Trans. Commun.*, vol. COM-36, pp. 957-971, August 1988.

[14] N. S. Jayant and P. Noll, *Digital coding of waveforms, principles and applications to speech and video*, Englewood Cliff, NJ: Prentice-Hall, 1984.

[15] A. Gersho and R. M. Gray, *Vector quantization and signal compression*, Kluwer Academic Publishers, Boston, 1992.

[16] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image-coding techniques," *Proc. IEEE,* vol. 73, pp. 549-574, April 1985.

[17] M. Kunt, M. Benard, and R. Leonardi, "Recent results in high-compression image coding," *IEEE Trans. Circuit and Systems*, vol. CAS-34, pp. 1306-1336, Nov. 1987.

[18] S. Carlsson, "Sketch based coding of grey level images," *Signal Processing*, vol. 15, pp. 57–83, July 1988.

[19] S. Carlsson, C. Reillo, and L. H. Zetterberg, "Sketch based representation of grey value and motion information," in *From Pixels to Features* by J. C. Simon (ed.) Elsevier Science Publishers B.V. (North-Holland), 1989.

[20] J. K. Yan and D. J. Sakrison, "Encoding of images based on a two-component source model," *IEEE Trans. Commun.*, vol. COM-25, pp. 1315-1322, Nov. 1977.

[21] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London*, Ser. B, vol. 207, pp. 187-217, 1980.

[22] X. Ran and N. Farvardin, "A perceptually motivated three-component image model," submitted to *IEEE Trans. Image Processing*, March 1992. Also, SRC Technical Report TR 92-32, University of Maryland, College Park, MD, March 1992.

[23] F. W. Campbell and J. G. Robson, "Application of Fourier analysis to the visibility of gratings," *J. Physiol.*, 197, pp. 551–566, 1968.

[24] A. N. Netravali and B. G. Haskell, *Digital pictures, representation and compression*, Plenum Press, New York, 1988.

[25] J. L. Mannos and D. J. Sakrison, "The effects of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 525-536, July 1974.

[26] X. Ran, "A three-component image model for human visual perception and its application in image coding and processing," Ph.D. dissertation, University of Maryland, College Park, MD, in preparation.

[27] H. Freeman, "On the encoding of arbitrary geometric configuration," *IRE Trans. Electron. Comput.*, EC-10, pp. 260-268, June 1961.

[28] T. S. Huang, "Coding of two-tone images," *IEEE Trans. Commun.*, vol. COM-25, pp. 1406-1424, November 1977.

[29] D. L. Neuhoff and K. G. Castor, "A rate and distortion analysis of chain codes for line drawings," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 53-67, January 1985.

[30] J. J. Rissanen, "Generalized Kraft inequality and arithmetic coding," *IBM J. Res. Develop.*, pp. 198-203, May 1976.

[31] I. H. Witten, R. M. Neal and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM,* vol. 30, pp. 520-540, June 1987.

[32] O. R. Mitchell and A. Tabatabai, "Adaptive transform image coding for human analysis," *Proc. ICC,* pp. 23.2.1-23.2.5, 1979.

[33] R. C. Reininger and J. D. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Trans. Commun.*, vol. COM-31, pp. 835-839, June 1983.

[34] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 485-497, May 1984.

[35] A. V. Trushkin, "Optimal bit allocation algorithm for quantizing a random vector," *Problems of Inform. Transmission,* vol. 17, pp. 156-161, 1981.

[36] H. Gish and J. N. Pierce, "Asymptotically efficient quantizing," *IEEE Trans. Inform. Theory,* vol. IT-14, pp. 676-683, September 1968.

[37] T. Berger, *Rate distortion theory*, Englewood Cliff, NJ: Prentice-Hall, 1971.

[38] M. H. Woehrmann, A. N. Hessenflow and D. Kettmann, *Image alchemy*^{tm}, *user's manual*, Version 1.4, Handmade Software, Inc., March 18, 1991.

[39] J. M. Shapiro,"An embedded wavelet hierarchical image coder," *Proc. IEEE ICASSP*, pp. IV.657–IV.660, March 1992.

[40] F. Kessentini, M. J. T. Smith and C. F. Barnes, "Image coding with variable rate RVQ," *Proc. IEEE ICASSP*, pp. III.369-III.372, March 1992.

[41] M. G. Perkins and T. Lookabaugh, "A psychophysically justified bit allocation algorithm for subband image coding systems," *Proc. IEEE ICASSP*, pp. 1815-1818, May 1989.

[42] M.K.H. Fan, A. L. Tits, J. Zhou, L.-S. Wang and J. Koninckx, "CONSOLE, User's Manual, version 1.1" *Technical Research Report*, TR 87-212r2, SRC, University of Maryland, College Park, MD, June 1990.

[43] T. R. Fischer and M. Wang, "Entropy-constrained trellis-coded quantization," *IEEE Trans. Inform. Theory,* vol. IT-38, pp. 415-426, March 1992.

[44] R. Laroia and N. Farvardin, "Trellis-based scalar-vector quantizer for memoryless sources," submitted to *IEEE Trans. Inform. Theory* for publication, July 1992.

Configuration	1-ring	2-ring	3-ring
A	0.048	0.038	0.032
B	0.017	0.013	0.011

Table 1: Bit rate (bits/pixel) for encoding the primary component for two different parameter sets.

Encoding Scheme	Design Bit Rate (bpp)			
	0.125	0.25	0.5	0.75
JPEG-HC	24.91 (0.148)	31.25 (0.283)	34.69 (0.513)	36.43 (0.748)
1C-ADCT-HC	30.10 (0.126)	32.92 (0.247)	35.91 (0.485)	37.91 (0.747)
3C-ADCT-HC	30.20 (0.126)	33.29 (0.250)	36.25 (0.492)	38.03 (0.747)
1C-SBC-HC	29.77 (0.125)	32.55 (0.250)	35.67 (0.500)	37.73 (0.748)
3C-SBC-HC	29.67 (0.125)	32.67 (0.251)	35.73 (0.494)	37.63 (0.750)
JPEG-AC	28.45 (0.128)	32.08 (0.265)	34.95 (0.491)	36.72 (0.755)
1C-ADCT-AC	30.10 (0.123)	32.92 (0.241)	35.91 (0.471)	37.91 (0.732)
3C-ADCT-AC	30.20 (0.123)	33.29 (0.243)	36.25 (0.479)	38.03 (0.731)

Table 2: PSNR (in dB) performance of various encoding schemes for 512×512 Lenna at different design bit rates. Numbers in parantheses indicate actual bit rates.

Encoding Scheme	Design Bit Rate (bpp)			
	0.125	0.25	0.5	0.75
JPEG-HC	23.27 (0.214)	25.95 (0.296)	29.17 (0.488)	31.62 (0.762)
1C-ADCT-HC	25.42 (0.129)	28.02 (0.256)	31.16 (0.503)	33.25 (0.739)
3C-ADCT-HC	25.61 (0.128)	28.41 (0.256)	31.62 (0.505)	33.80 (0.750)
1C-SBC-HC	25.55 (0.125)	27.98 (0.250)	31.07 (0.503)	33.36 (0.750)
3C-SBC-HC	25.50 (0.125)	28.15 (0.251)	31.18 (0.503)	33.36 (0.752)
JPEG-AC	23.27 (0.124)	27.49 (0.291)	30.21 (0.516)	32.18 (0.762)
1C-ADCT-AC	25.42 (0.120)	28.02 (0.238)	31.16 (0.470)	33.25 (0.696)
3C-ADCT-AC	25.61 (0.120)	28.41 (0.240)	31.62 (0.474)	33.80 (0.707)

Table 3: PSNR (in dB) performance of various encoding schemes for 256×256 Lenna at different design bit rates. Numbers in parantheses indicate actual bit rates.

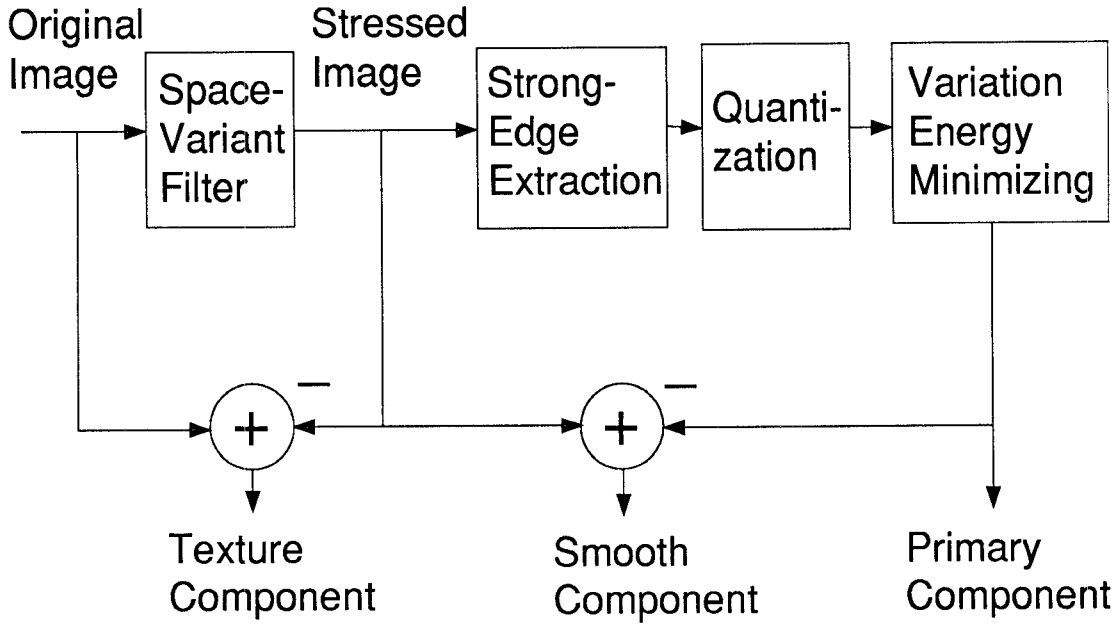


Figure 1: Generation of three-component image model.

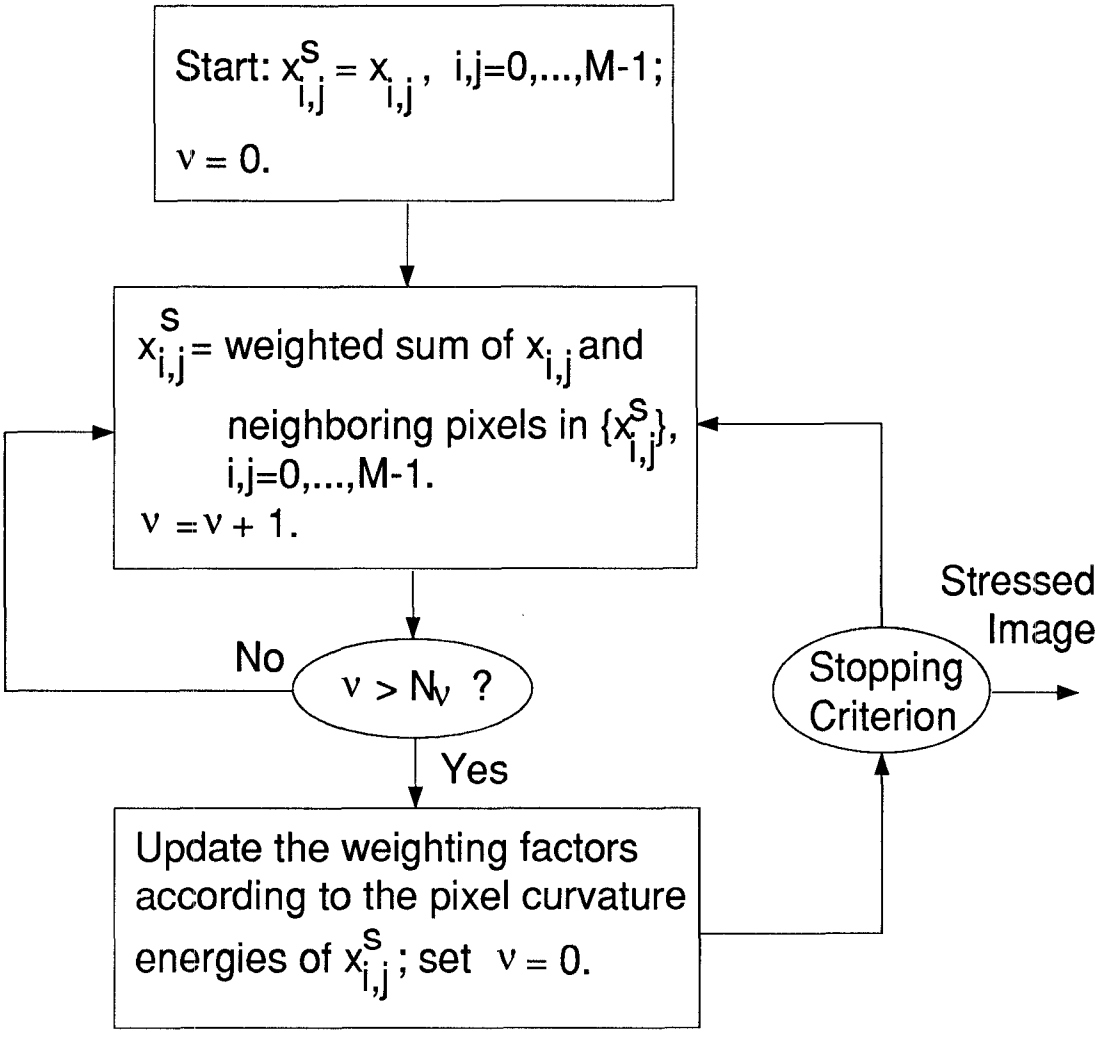


Figure 2: Flow chart describing the generation of the stressed image.

(a)

(b)

(c)

(d)

(e)

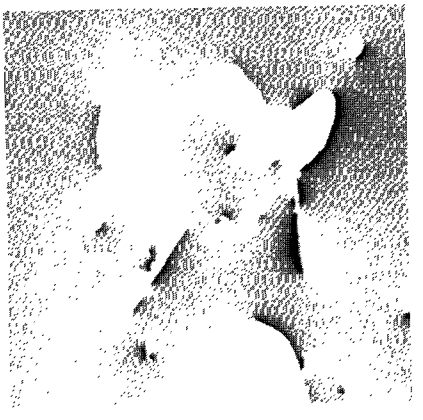
(f)

Figure 3: (a) Lenna image, (b) the stressed image, (c) the strong edge brim contours, (d) the primary component (1-ring coding), (e) the smooth component and (f) the texture component associated with A-configuration.

(a)

(b)

(c)



(d)

(e)

(f)

Figure 4: (a) Lenna image, (b) the stressed image, (c) the strong edge brim contours, (d) the primary component (1-ring coding), (e) the smooth component and (f) the texture component associated with B-configuration.

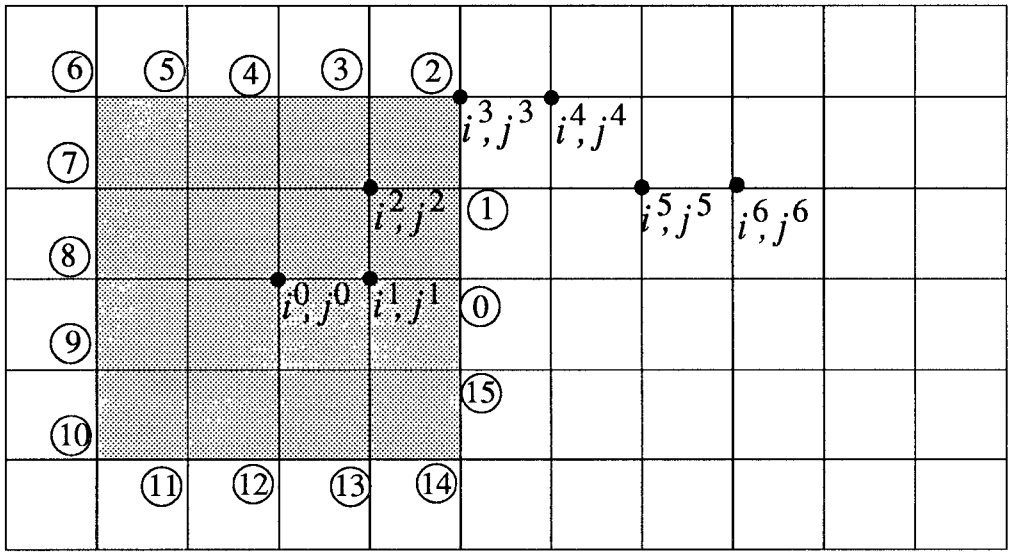


Figure 5: Example of 2-ring coding.

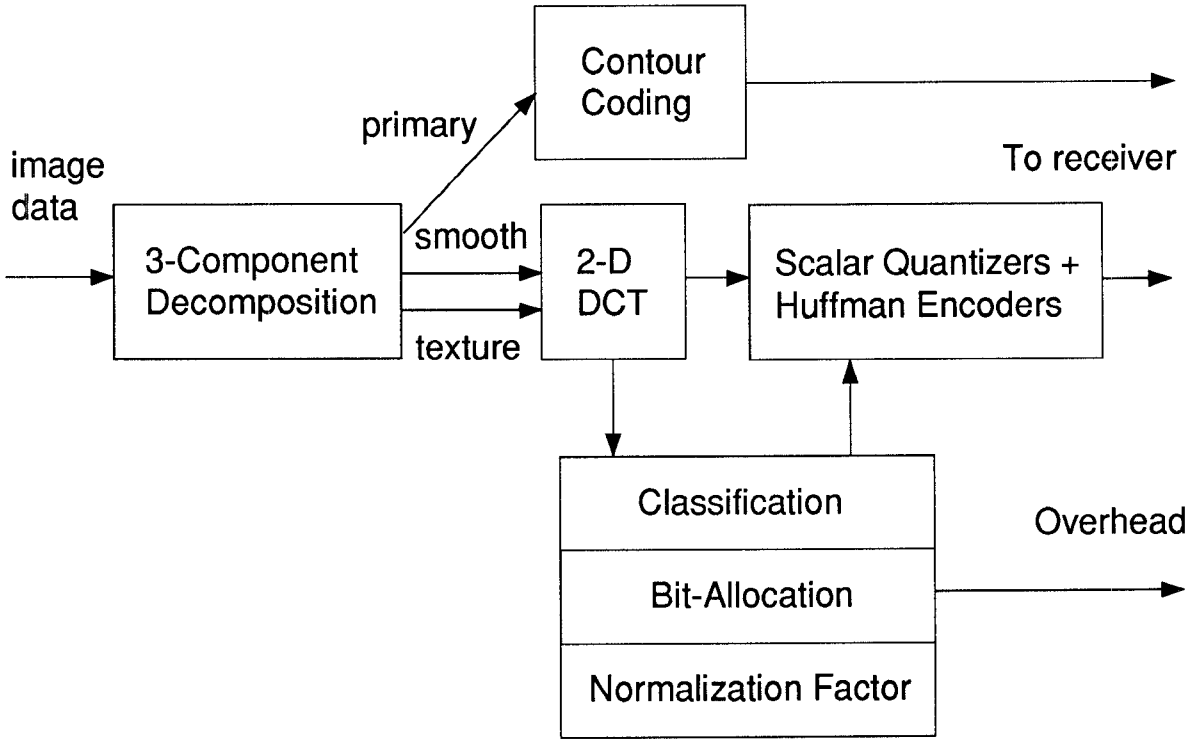


Figure 6: Block diagram of the ADCT-based image coding scheme using the 3-component model.

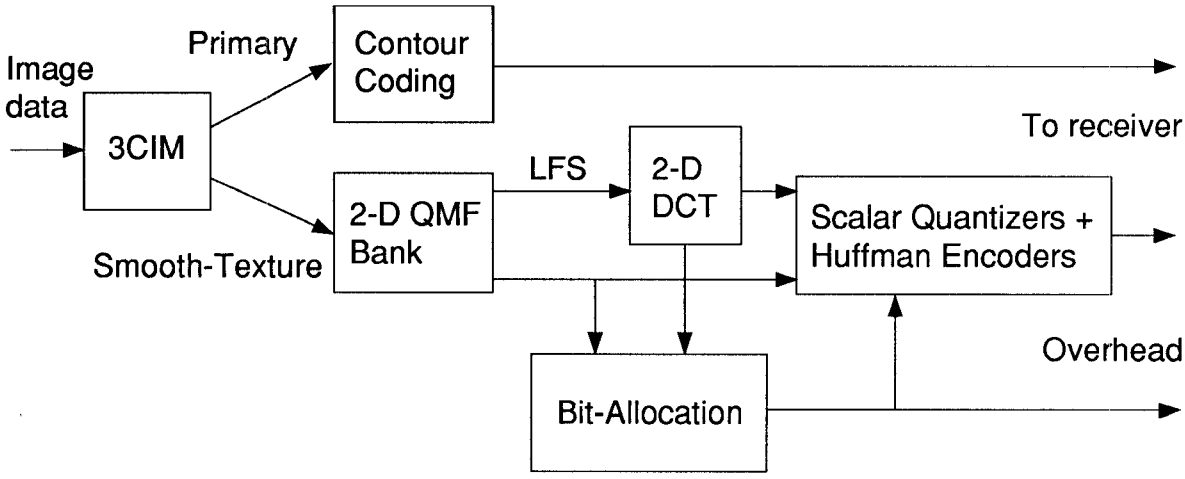


Figure 7: Block diagram of the SBC-based image coding scheme using the 3-component model.

(a)

(b)

(c)

(d)

(e)

(f)

Figure 8: (a) Original, (b) JPEG-HC, (c) 1C-ADCT-HC, (d) 3C-ADCT-HC, (e) 1C-SBC-HC, (f) 3C-SBC-HC; design bit rate 0.5 bpp; actual bit rates in Table 2.

(a)

(b)

(c)

(d)

(e)

(f)

Figure 9: (a) Original, (b) JPEG-HC, (c) 1C-ADCT-HC, (d) 3C-ADCT-HC, (e) 1C-SBC-HC, (f) 3C-SBC-HC; design bit rate 0.25 bpp; actual bit rates in Table 2.

(a)

(b)

(c)

(d)

(e)

(f)

Figure 10: (a) Original, (b) JPEG-HC, (c) 1C-ADCT-HC, (d) 3C-ADCT-HC, (e) 1C-SBC-HC, (f) 3C-SBC-HC; design bit rate 0.125 bpp; actual bit rates in Table 2.

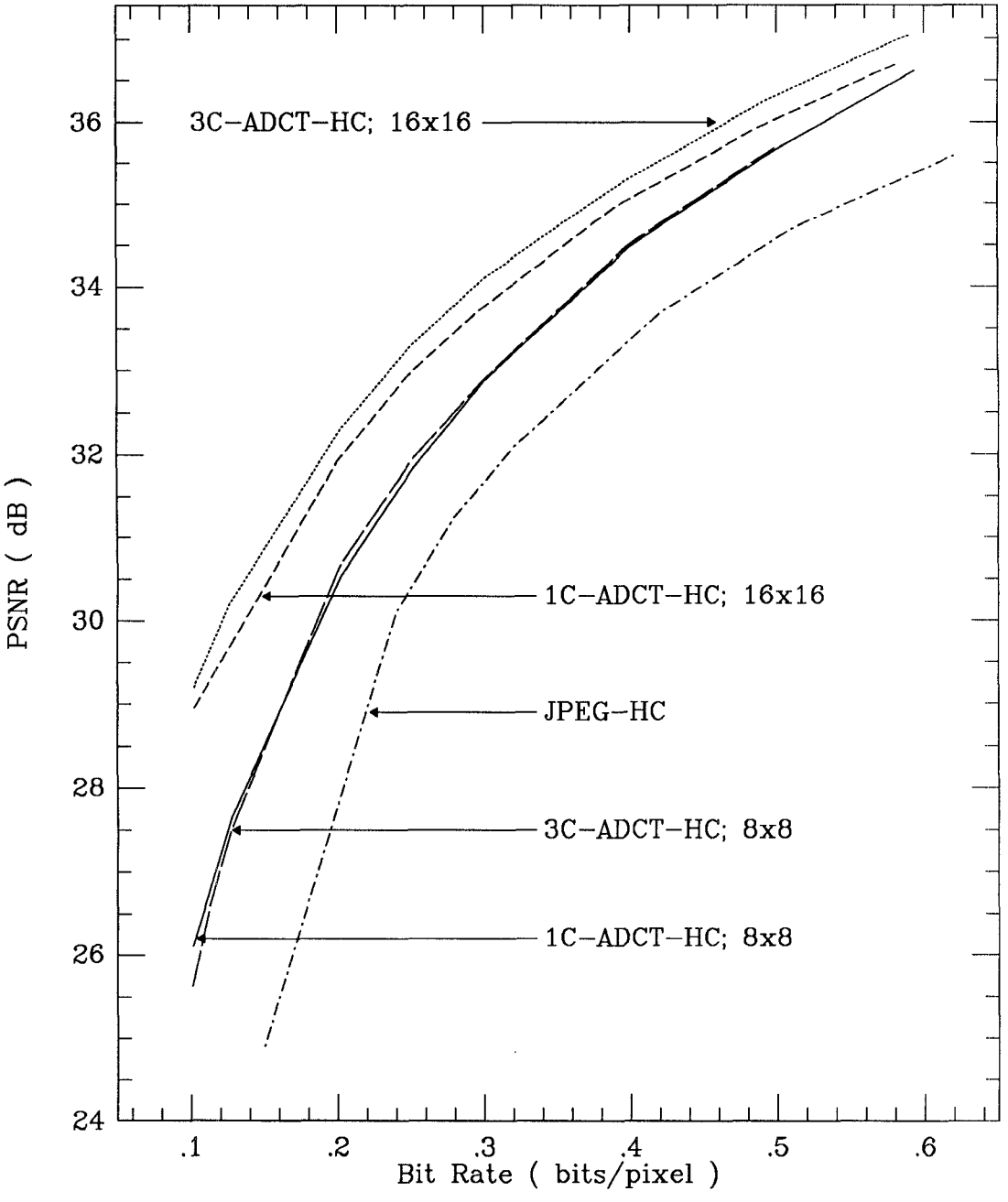


Figure 11: PSNR (in dB) vs. bit rate for JPEG-HC with blocksize 8×8 as well as 1C-ADCT-HC and 3C-ADCT-HC with blocksizes 8×8 and 16×16.

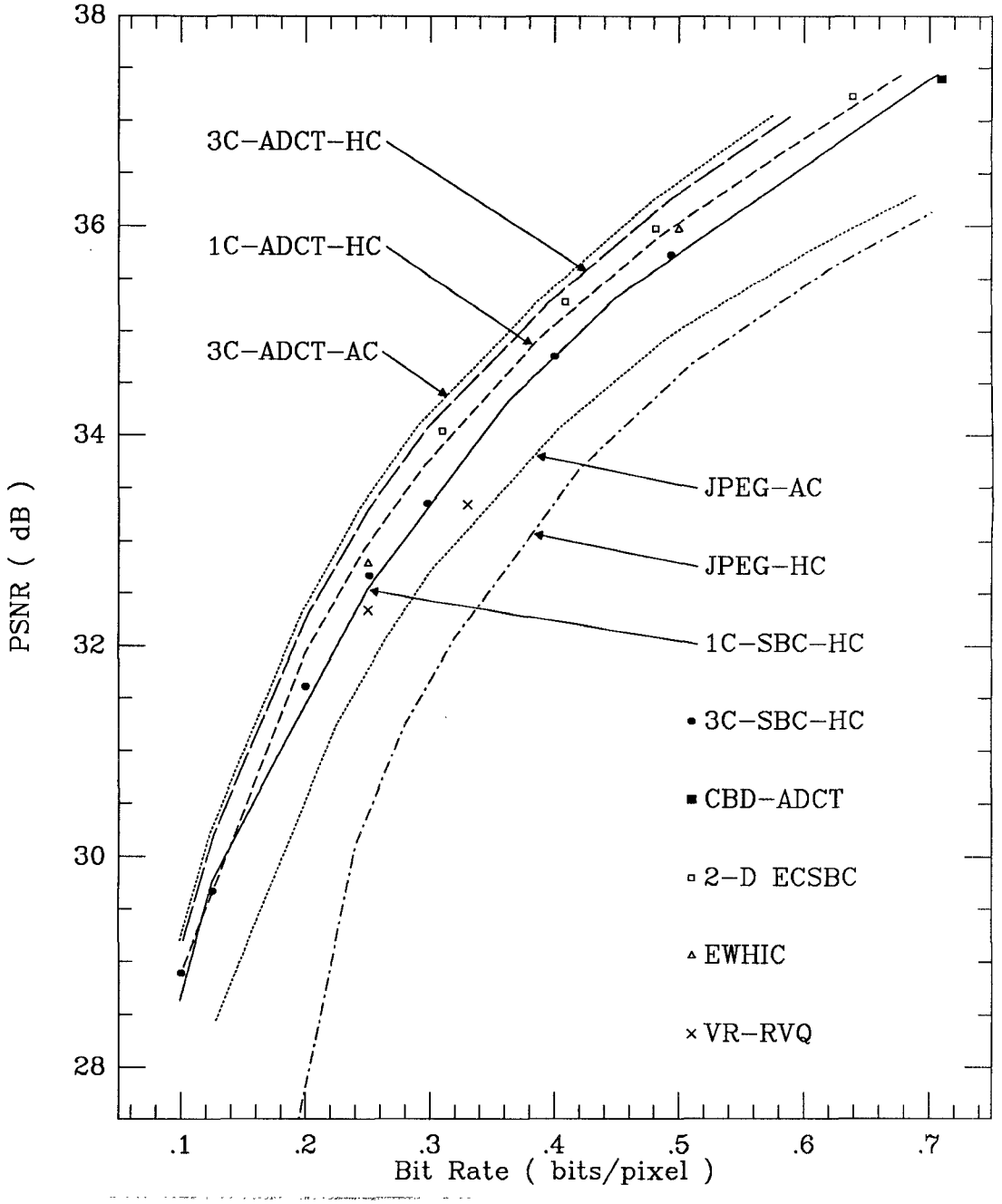


Figure 12: PSNR (in dB) vs. bit rate for various encoding schemes; 512 × 512 Lenna.

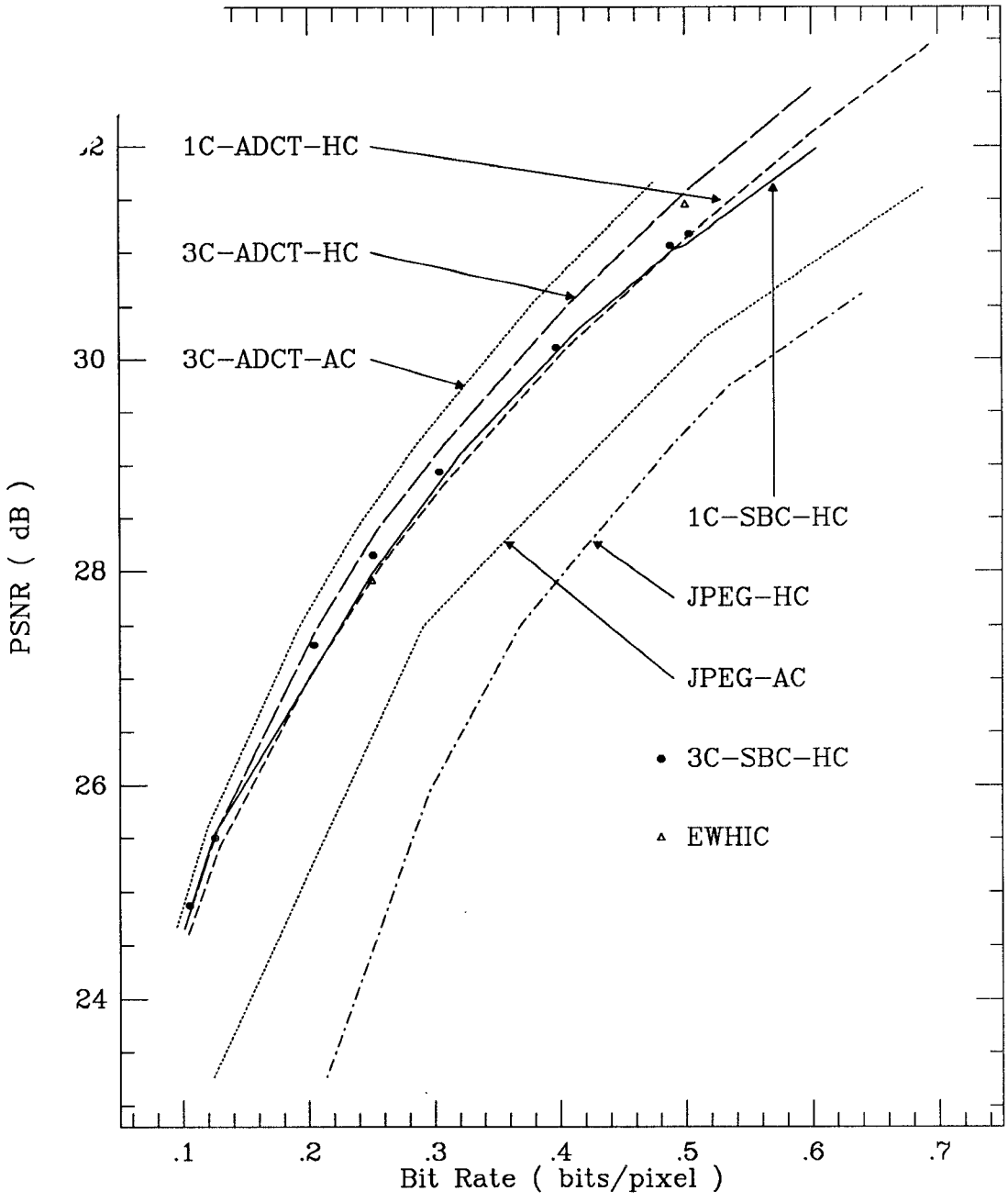
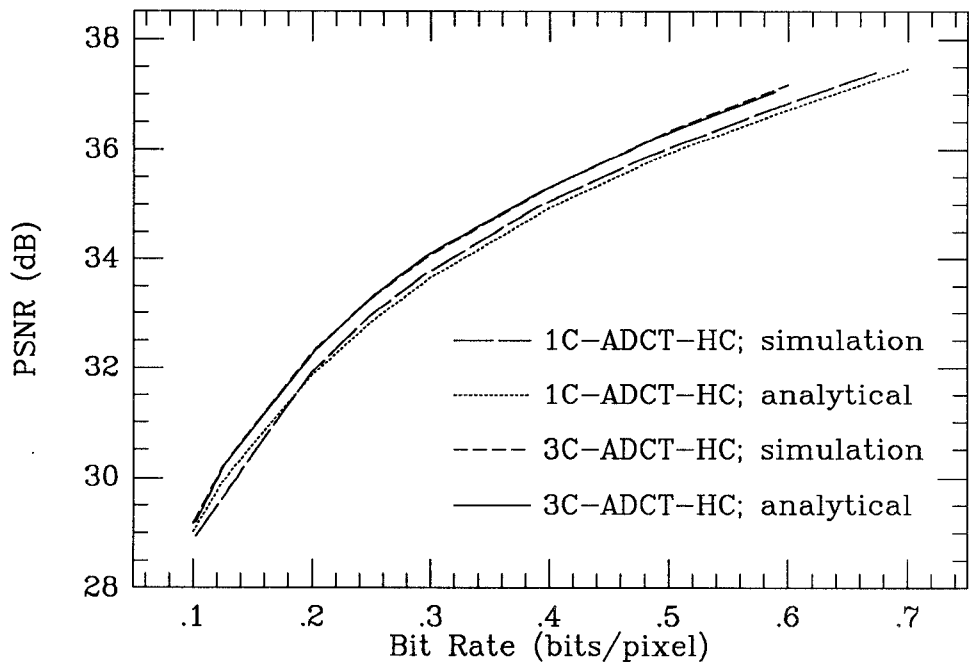
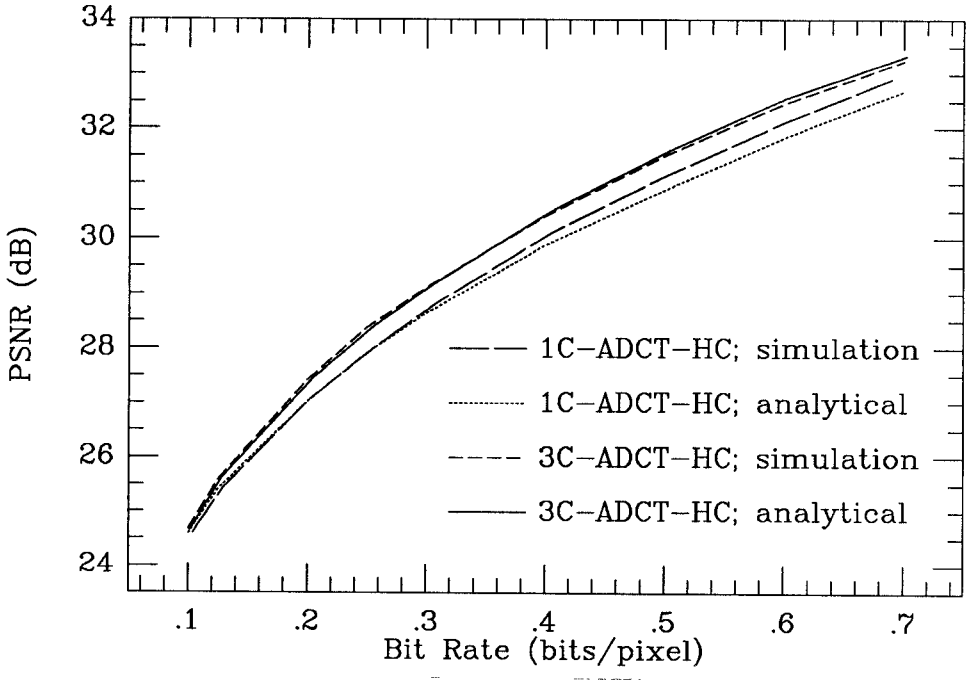


Figure 13: PSNR (in dB) vs. bit rate for various encoding schemes; 256×256 Lenna.

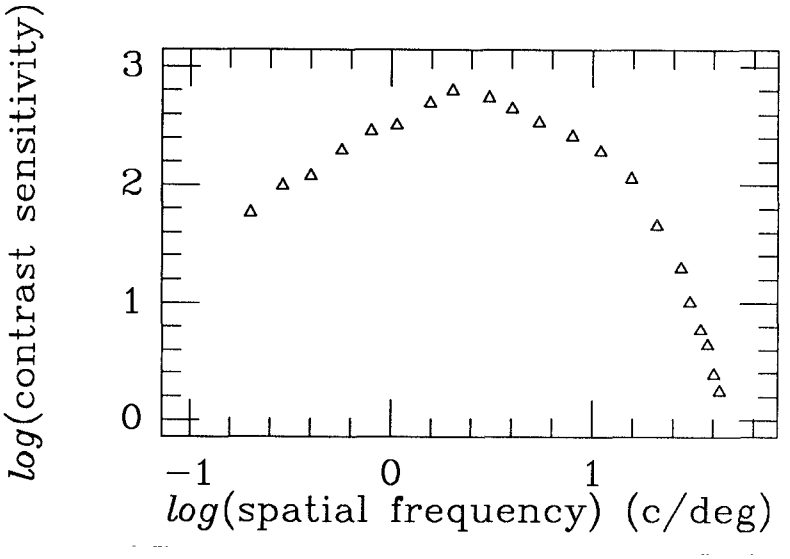


(a)

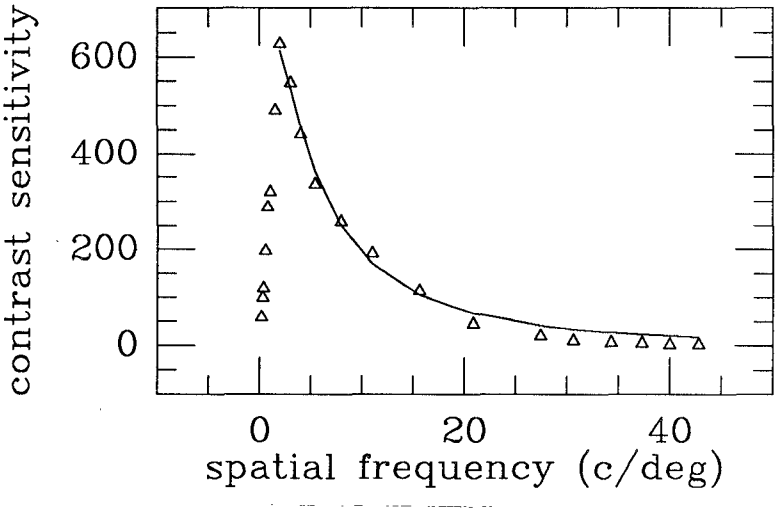


(b)

Figure 14: Analytical and simulation PSNR (in dB) vs. bit rate for 1C-ADCT-HC and 3C-ADCT-HC; (a) 512×512 Lenna, (b) 256×256 Lenna.



(a)



(b)

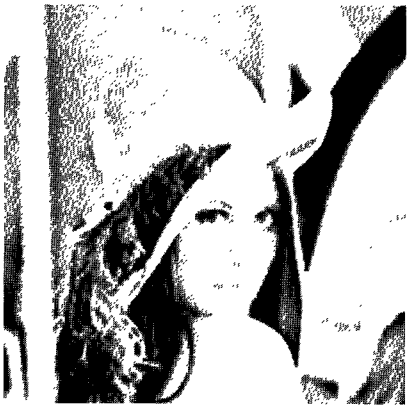
Figure 15: Contrast sensitivity for sinusoidal-intensity gratings (a) logrithemic scale and (b) normal scale along with the least-squares fit.

(a)

(b)

(c)

(d)



(e)

(f)

Figure 16: (a), (c), (e) 3C-ADCT-HC with perceptual weighting at design bit rates 0.5, 0.25, 0.125 bpp, respectively; (b), (d), (f) 1C-ADCT-HC with perceptual weighting at design bit rates 0.5, 0.25, 0.125 bpp, respectively.