Second-Generation Image Coding: An Overview

M. M. REID, R. J. MILLAR, AND N. D. BLACK

University of Ulster, Jordanstown, North Ireland

This article gives an overview of a diverse selection of currently used second-generation image coding techniques. These techniques have been grouped into similar categories in order to allow a direct comparison among the varying methods. An attempt has been made, where possible, to expand upon and clarify the details given by the original authors. The relative merits and shortcomings of each of the techniques are compared and contrasted.

Categories and Subject Descriptors: E4 [**Data**]: Coding and Information Theory—*data compaction*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*trees*; I.4.2 [**Image Processing**]: Compression (Coding)—*approximate methods*, *exact coding*; I.4.3 [**Image Processing**]: Enhancement—*filtering*; *smoothing*; I.5.4 [**Pattern Recognition**]: Applications—*waveform analysis*;

General Terms: Algorithms, Human Factors, Theory

Additional Key Words and Phrases: Compression, image coding, MRi

1. INTRODUCTION

Until approximately 1980, the majority of image coding methods relied on techniques based on classical information theory [Golomb 1966; Samet 1989b; Ziv and Lempel 1977; Welch 1984; Huffman 1952] to exploit the redundancy in the images in order to achieve compression. The techniques used were pixel-based and did not make any use of the information contained within the image itself. The compression ratios obtained with these techniques were moderate at around 2 to 1. Even with a lossy technique, such as discrete cosine transform, DCT [Wallace 1991], a higher ratio (greater than 30 to 1) could be achieved only at the expense of image quality (see Figure 1) [Reid et al. 1994].

Attempts have recently been made to develop new image-compression techniques that outperform these first-generation image coding techniques. These methods attempt to identify features within the image and use the features to achieve compression. These recent developments have been termed second-generation image coding [Kunt et al. 1985; Kunt 1988, 1990]. All second-generation techniques incorporate properties of the human visual system (HVS) into the coding strategy in order to achieve high compression ratios while still maintaining acceptable image quality. In other words, most of the techniques are of a lossy nature: however,

The work of M. M. Reid was financially supported by the Department of Education for Northern Ireland (DENI).

Authors' addresses: University of Ulster, Jordanstown, N. Ireland; email: ⟨m.reid,rj.millar,nd.black@ulst.ac.uk⟩.

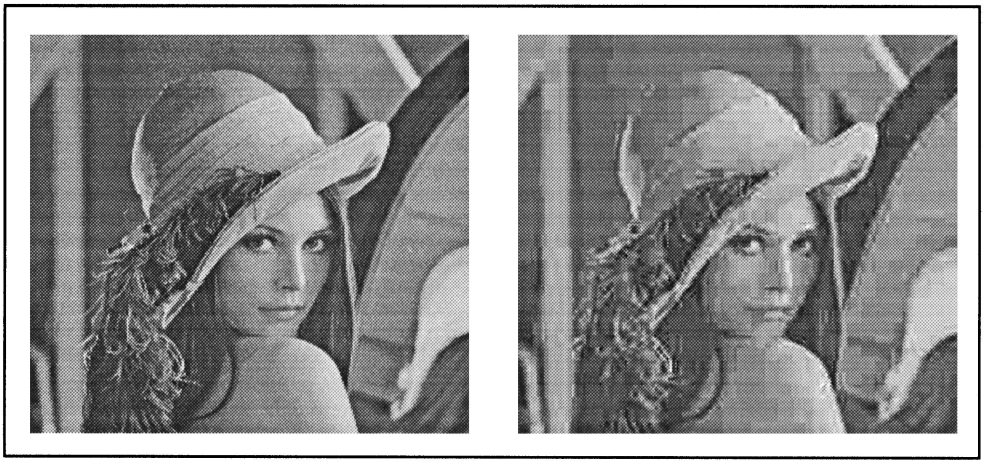

Figure 1. Effects of over-compression using DCT. Left-hand image is the original; right-hand image has been compressed via DCT to ratio of 65 to 1.

they attempt to identify and separate visually significant and visually insignificant areas of the image and apply appropriate coding techniques to each area.

In current image coding techniques, much effort has been expended in identifying what the human observer considers visually important [Cornsweet 1974; Rosenfeld and Kak 1982a; Jain 1989]. It is generally concluded that edge information is vital to the human perception of images [Kunt et al. 1985]. As a result, a vast majority of the work in the development of image coding techniques has concentrated on methods that preserve edge information and separate edge and texture information while coding them separately [Rosenfeld and Kak 1982b].

The loss of image detail in lossy second-generation techniques has been minimized in terms of human perception, so that when decoded, the new image does not appear to be different from the original. However, there are situations that require lossless compression of images, particularly in the medical image domain [Wittenberg 1993]. Lossless coding does not remove any of the image data in order to achieve compression and hence the decoded image is identical to the original. Most of the lossy techniques could be adapted to become lossless since it is normally the introduction of an error stage in the process that provides the facility for compression. However, if these methods were converted to a lossless mode, compression would almost certainly not occur. Therefore lossless techniques that are a combination of first- and second-generation methods have been developed in order to exploit the contents of the image but still preserve the image data [Miller and Nicholl 1994].

With second-generation coding techniques the original image is broken down into subcomponents. These subcomponents may consist of homogeneous regions [Jang and Rajala 1991, 1990; Civanlar et al. 1986; Kwon and Chellappa 1993; Kocher and Kunt 1983; Biggar et al. 1988; Cicconi and Kunt 1993; Cicconi et al. 1994; Leou and Chen 1991; Kocher and Leonardi 1986], predefined visual patterns [Chen and Bovik 1990; Silsbee and Bovik 1991], directional components [Ikonomopoulos and Kunt 1985; Zhou and Venetsanopoulos 1992], and pyramidal decompositions [Burt and Adelson 1983; Mallat and Zhong 1991]. These broad components are used to categorize the relevant techniques.

This article gives an overview of a diverse selection of second-generation image coding techniques. Although most of the techniques considered second generation are lossy, a single lossless technique based on second-generation principles is included for completeness. The article is not intended to cover every second-generation image coding technique that has been developed. However, it is intended to provide an introduction to the broad categories of image coding techniques that exist. Section 2 describes some of the commonly used properties of the human visual system. Section 3 describes techniques that decompose the image into pyramidal representations in order to code the original. Section 4 presents techniques that employ directional filters as part of the coding process. Section 5 describes techniques for coding images based on visual patterns. Section 6 describes the techniques based on image segmentation and presents some of the theoretical methods used to preprocess the image before segmentation takes place. Finally, Section 7 covers methods based on contour coding and describes two methods, one lossy and one lossless.

2. PROPERTIES OF THE HUMAN VISUAL SYSTEM

This section describes some of the properties of the human visual system, HVS, that have been incorporated into part of an image coding system. As yet, a biologically complete version of the HVS does not exist. However, common models for the HVS would include a low-pass filter, a logarithmic nonlinearity, and a multichannel signal-sharpening high-pass filter. The majority of the early work in vision research has used the frequency sensitivity of the HVS as described by the modulation transfer function, MTF [Jayant et al. 1993]. Experiments by Mannos [Mannos and Sakrison 1974] and Cornsweet [1974] have proposed a commonly used model for this function that relates the sensitivity of the eye to sine-wave gratings at various frequencies. The nonlinearities of the HVS at threshold sensitivity can be described by a set of laws, the best known of which is Weber's Law [Forchheimer and Kronander 1989]. This law states that the HVS threshold sensitivity, for spatially large stimuli and large background illumination can be expressed as

$$\Delta L = kL,$$

where k is a constant and ΔL is the just noticeable difference in luminance.

Another property commonly used is the spatial and temporal frequency dependencies of the visual system. From measurements made by Robson [1966] on the joint spatio-temporal threshold sensitivity, it can be seen that this joint sensitivity is not separable and that the threshold sensitivity depends on both the spatial and temporal frequencies. In the high spatial frequency range the visual system is essentially low-pass, whereas in the low range the response is band-pass.

Directional anisotropy, which is related to the decreasing sensitivity at high and low spatial frequencies, is often exhibited in the HVS [Olzak and Thomas 1986]. Considering sinusoidal waveforms, sensitivity is decreased by approximately 3dB at 45° rotation from the horizontal. This has been shown to be due to the lower sensitivity of the eye at oblique directions compared to vertical or horizontal directions.

Finally, another common property of the HVS is its spatial and temporal masking effects. This property is related to the fact that sensitivity is reduced in the neighborhood of stimuli with large intensity variations. A typical example of this is that sensitivity is reduced in the close surroundings of a high intensity luminance edge. This masking effect occurs in both spatial and temporal domains: however, the effect is only pronounced in a very small region around the masking stimuli [Limb 1978].

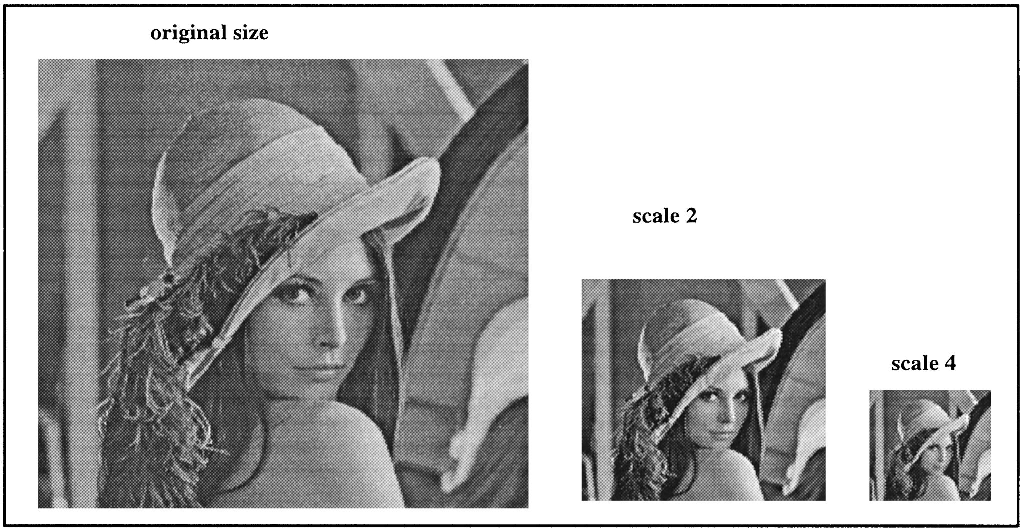

Figure 2. Set of images representing pyramidal decomposition at two levels.

3. MULTISCALE/PYRAMIDAL APPROACHES

Multiscale and pyramidal techniques operate on the original image and subsample it in order to produce various levels of image detail at progressively smaller details.

Burt and Adelson [1983] present a technique based on the Laplacian pyramid as a method of image compression. Laplacian pyramid image coding has several main stages. The first stage is to low-pass filter the original image using a weighted average function. This stage produces progressively smaller images, in both spatial intensity and dimension, which together form the Gaussian pyramid (Figure 2). Secondly, each stage in the Gaussian pyramid, starting with the lowest, smallest level, is interpolated to the size of its predecessor. This expanded version is then subtracted from the predecessor in order to produce the Laplacian pyramid.

In terms of the coding it is the Laplacian pyramid that is coded as opposed to the original image. Adding the levels of the Laplacian pyramid with appropriate interpolation results in the original image if the last, smallest-resolution image from the Gaussian pyramid is included. The final stage of the coding process is to quantize each level of the pyramid. This quantization is achieved by dividing the range of pixel values into bins of set width: quantization then occurs by representing each pixel value that occurs within the bin by the bin centroid.

The fact that compression is achieved by quantizing the pixels within each level of the Laplacian pyramid means that compression ratios can be altered by increasing or decreasing the amount of quantization. The nature of this technique means that it is particularly suited to progressive transmission. This is due to the fact that each level of the pyramid provides progressively coarser details of the original image. Hence, if only a rough approximation is required, initially the lowest, smallest level of the pyramid can be transmitted and used for decoding. If greater quality is required, then the next pyramid layer can be sent, and so on.

A more popular technique presently used is wavelet compression [Averbuch et al. 1996; O'Rourke and Stevenson 1995; Huffman 1994]. This technique is very similar to the preceding technique of Burt and Adelson [1983] with the

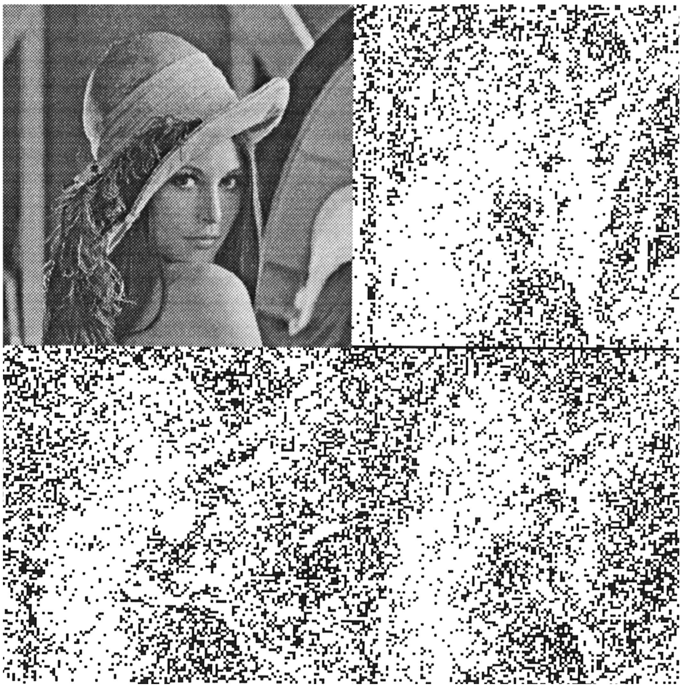

Figure 3. Wavelet decomposition.

bonus that wavelet decomposition does not increase the number of samples over that of the original image whereas pyramidal decomposition does. In image processing, most of the image area represents areas of high statistical correlation. However, edges, for example, have a perceptual significance that is greater than the numerical value of the energy contribution to the entire image. Other transform coders, for example, DCT, decompose images into representations where each coefficient corresponds to a fixed-size spatial area and frequency band. Edge information requires many nonzero coefficients to represent them sufficiently. At low bit rates other transform coders allocate too many bits to signal behavior that is more localized in the time or space domain and not enough to edges. Wavelet techniques offer benefits at low bit rates since information at all scales is available for edges and regions [Shapiro 1993]. The wavelet transform decomposes an image into a set of subimages called shapes with different resolutions corresponding to different frequency bands. The initial image is typically decomposed into a low-pass decomposition, a high-pass decomposition showing horizontal details, a high-pass decomposition showing vertical details, and a high-pass decomposition showing both vertical and horizontal (see Figure 3).

If, for example, the low-pass decomposition is transmitted and used to reconstruct the original (see Figure 4), a 4:1 compression has been achieved. However, in order to ensure high-resolution image results on decompression, the three high-pass bands must also be included. The decomposition process can be carried out a number of times depending on the level of compression required.

Mallat and Zhong [1991] point out that, in most cases, structural information required for recognition tasks is provided by the image edges. However, one major difficulty of edge-based representation is to integrate all the image

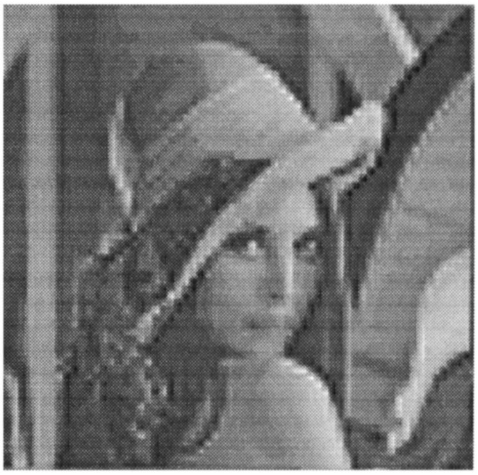

Figure 4. Image decompressed using only low-pass decompositions.

information into edges. Most edge detectors are based on local measurements of the image variations and edges are generally defined as points where the image intensity has a maximum variation. Multiscale edge detection is a technique in which the image is smoothed at various scales and edge points are detected by a first- or second-order differential operator. The image smoothing produces a structure similar to that illustrated in Figure 3. The coding method presented involves two steps: the edge points considered important for visual quality are selected, and then efficiently encoded. Edge points are chained together to form edge curves.

Selection of the edge points is performed at a scale of 2^2. This means that the edge points are selected from the image in the pyramidal structure that has been scaled to a factor of four. Boundaries of important structures often generate long edge curves, so first all edge curves whose length is smaller than a threshold are removed. Among the remaining curves, the ones that correspond to the sharpest discontinuities in the image are selected. This is achieved by removing all edge curves along which the average value of the wavelet transform modulus is smaller than a given amplitude threshold. After the removal procedures, it is reported that only 8% of the original edge points are kept, although it is not clear if this figure is constant for all images.

Once the selection has been performed, only the edge curves at scale 2^2 are coded in order to save bits: the curves at other scales are approximated from this. Chain coding is used to encode the edge curve at this scale.

Although the preceding techniques in this section decompose the original image into a set of subimages, other techniques exist that employ decomposition to further reduce storage required by parameters from other techniques. Examples of this are AVPIC [Silsbee and Bovik 1991], discussed in Section 5, and FMP [Zhou and Venetsanopoulos 1992] discussed in Section 4.

The techniques covered in this section rely on the technique of creating a pyramid of progressively smaller, subsampled images. The pyramidal structure has the ability to identify common features that occur in the image at all resolutions.

In the case of Burt and Adelson [1983], an attempt is made to exploit areas in the image that are largely similar. These similar areas appear at various resolutions. Hence, when the subsampled image is expanded and subtracted from the image at the next higher resolution, the difference image contains large areas of zero, indicating commonality between the two images. The larger the degree of commonality, the greater the amount of zero areas in the difference images. Standard first-generation coding methods can then be applied to the difference images to produce good compression ratios. With very good image quality, compression ratios in the range 10 to 1 are achievable.

Mallat and Zhong [1991] use the pyramidal structure to identify image edges that are most important to the perception of the image. These edges are identified in the image that has been generated at a scale of 2^2, although the reason for this choice is not clear. Once selected, the remaining edges are approximated using the edges at scale 2^2. The compression ratio reported with this method is approximately 27 to 1 with good image quality.

4. DIRECTIONAL FILTERING

Directional filtering is based on the relationship between the presence of an edge in an image and its contribution to the image spectrum. It is motivated by the existence of direction-sensitive neurons in the HVS [Kunt et al. 1985]. It can be seen that the contribution of an edge is distributed all over the spectrum: however, the highest-frequency component lies in the direction orthogonal to that of the edge. It can also be seen that the frequency of the contribution diminishes as we turn away from this direction, until it vanishes at right angles to it. A directional filter is one whose frequency response covers a sector or part of a sector in the frequency domain. If f and g are spatial frequencies and r_c is the cutoff frequency of the low-pass filter, then the ideal frequency response of the ith directional filter of a set of n is given by:

$$G_{Ii}(f, g) = \begin{cases} 1, & \text{if } \theta_i < \tan^{-1}(g/f) < \theta_{i+1} \\ & \text{and } f^2 + g^2 > r_c^2 \\ 0 & \text{otherwise,} \end{cases}$$

with

$$\theta_i = (i-1)\pi/2n, \quad \theta_{i+1} = (i+1)\pi/2n$$

and

$$|f|, |g| < 0.5.$$

A directional filter is a high-pass filter along its principal direction and a low-pass filter along the orthogonal direction (see Figure 5).

The directional filter response is modified, as in all filter design, by an appropriate window function [Harris 1978], to minimize the effect of the Gibbs phenomenon [Ziemer et al. 1989]. In the sum of a trigonometric series it can be seen that there tend to be overshoots in the signal being approximated at a discontinuity. This is referred to as the Gibbs phenomenon. An ideal filter can be viewed as a step or rectangular pulse waveform, that is, a discontinuous waveform. The reason for the overshoot at discontinuities can be explained using the Fourier transform. Consider a signal $x(t)$ with a Fourier transform $X(f)$. The effect of reconstructing $x(t)$ from its low-pass part shows that:

$$\tilde{x}(t) = F^{-1}\left[X(f)\prod\left(\frac{f}{2W}\right)\right],$$

where

$$\prod\left(\frac{f}{2W}\right) = \begin{cases} 1, & \text{when } |f| \leq W \\ 0, & \text{otherwise.} \end{cases}.$$

According to the convolution theorem of Fourier transform theory,

$$\tilde{x}(t) = x(t)*F^{-1}\left[\prod\left(\frac{f}{2W}\right)\right] \tag{1}$$

$$= x(t)*(2W\, sinc 2Wt). \tag{2}$$

Bearing in mind that convolution is a folding-product, sliding-integration process, it can be seen that a finite value of W will always result in $x(t)$ being viewed through the sinc window function, even though as W increases more of the frequency content of the rectangular pulse will be used in the approximation of $x(t)$.

In order to eliminate the Gibbs phenomenon it is important to modify the frequency response of the filter by a window function. There are many window functions available, each with different frequency responses. The chosen window function frequency response is convolved with the filter response to ensure that overall frequency response does not contain the sharp discontinuities that cause the ripple.

In a general scheme using directional filters, n directional filters and one low-

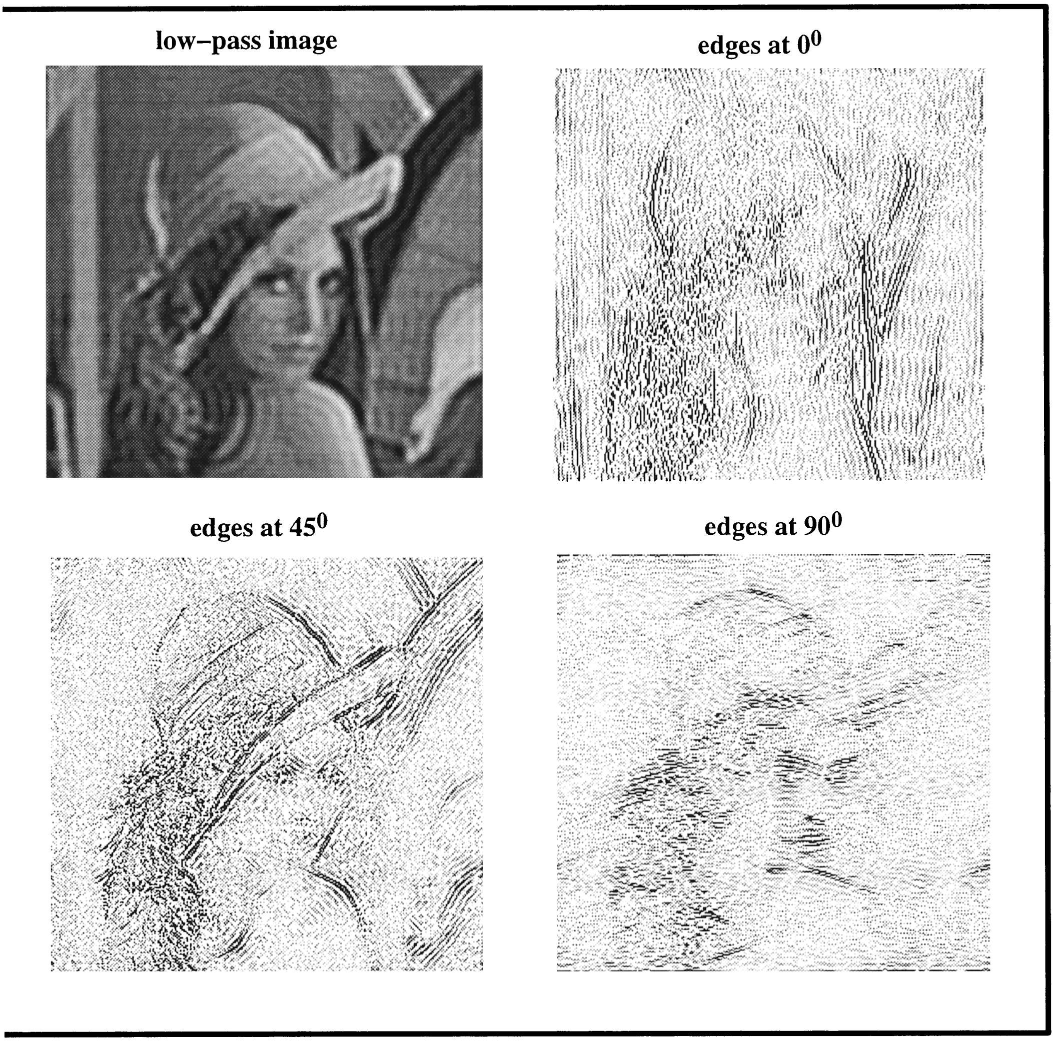

Figure 5. Set of directionally decomposed images. Low-pass cutoff frequency is 0.125.

pass filter are required. An ideal low-pass filter has the frequency response:

$$G_{Ii_p}(f, g) = \begin{cases} 1, & \text{if } f^2 + g^2 < r_c^2 \\ 0, & \text{otherwise.} \end{cases}$$

It should be noted that the superposition of all the directional images and the low-pass image leads to an exact reconstruction of the original image. Two parameters are involved in the design of a directional-filter-based image coding scheme: the number of filters and the cutoff frequency of the low-pass filter. The number of filters may be set a priori and is directly related to the minimum width of the edge elements. The choice of low-pass cutoff frequency influences the compression ratio and the quality of the decoded image.

As reported by Kunt et al. [1985], a very early technique in advance of its time was the synthetic high system [Schreiber et al. 1959; Schreiber 1963].

It is stated by Kunt that the better known approach of directional filtering is a refinement of the synthetic high system. In this technique the original image is split into two parts: the low-pass picture showing general area brightness and the high-pass image containing edge information. Two-dimensional sampling theory suggests that the low-pass image can be represented with very few samples. In order to reduce the amount of information in the high-pass image, thresholding is performed to determine which edge points are important. Once found, the location and magnitude of each edge point is stored. To reconstruct the compressed data, a two-dimensional reconstruction filter, whose properties are determined by the low-pass filter used to produce the low-pass image, is used to synthesize the high-frequency part of the edge information. This synthesized image is then added to the low-pass image to give the final output.

Ikonomopoulos and Kunt [1985] describe their technique for image coding based on the refinement of the synthetic high system, directional filtering. Once the image has been filtered, the result is 1 low-pass image and 16 directional images. The coding scheme proposed is not a lossless one since high compression is the goal. When the image is filtered with a high-pass filter the result gives zero crossings at the location of abrupt changes (edges) in the image. Each directional component is represented by the location and magnitude of the zero crossing. Given that a small number of points result from this process, about 6 to 10% of the total number of points, run-length encoding proves efficient for this purpose. The low-frequency component can be coded in two ways. Since the maximum frequency of this component is small, it can be resampled based on the 2D sampling theorem and the resulting pixels can be coded in a standard way. Alternatively, transform coding may be used, with the choice of transform technique being dictated by the filtering procedure used. The transform coefficients may then be quantized and coded via Huffman coding [Huffman 1952].

The compression ratios obtained with this technique depend on many factors. The image being coded and the choice of cutoff frequency all play an important role in the final ratio obtained. The compression scheme can be adapted to the type of image being compressed.

Zhou and Venetsanopoulos [1992] present an alternative spatial method called morphological directional coding. In their approach, spatial image features at known resolutions are decomposed using a multiresolution morphological technique called the feature-width morphological pyramid (FMP). Zhou and Venetsanopoulos [1992] report that the nontrivial spatial features, such as edges, lines, and contours within the image, determine the quality of the reproduced image for the human observer. It was this fact that motivated them to employ a stage in their coding technique that identifies these nontrivial features in order that they may be coded separately. Morphological directional coding schemes were developed to preserve nontrivial spatial features in the image during the coding phase. Such filtering techniques are used for feature separation as they are spatial methods that are capable of selectively processing features of known geometrical shapes. A multiresolution morphological technique therefore decomposes image features at various resolutions.

In this technique the image decomposition is a multistage process involving a filter called an open-closing (OC) filter. Each filtered image from the current stage is used as the input to the next stage, and in addition the difference between the input and output images of each stage is calculated. The first $N - 1$ decomposed subimages (L_1, . . . , L_{N-1}) are termed feature images and each contains image features at known resolutions. For example, L_1 contains image features of width 1, L_2 has features of width 2, and so on. Each OC filter has a structuring element associated with it. Each of these structuring elements for a stage is progressively larger than the previous stage. It is this structuring element that defines the information content in each of the decomposed images. The decomposed FMP images contain spatial features in arbitrary directions. Therefore directional decomposition filtering techniques are applied to each of the FMP images in order to group features of the same direction together. Before this is implemented, the features in the FMP images $L_2, \ldots, L_{N-1}$ must be eroded to 1 pixel width. Two reasons exist for this feature-thinning phase, as suggested by Zhou and Venetsanopoulos [1992]: the directional decomposition filter bank gives better results for features of 1 pixel width, and it is more efficient and simpler to encode features of 1 pixel width.

After the FMP images have been directionally decomposed, the features are further quantized by a nonuniform scalar quantizer. Each extracted feature is first encoded with a vector and then each vector is entropy-encoded. The coarse image L^N is encoded using conventional methods such as VQ.

Both of these methods employ directional decomposition as the basis of their technique. Ikonomopoulos and Kunt [1985] implement a more traditional approach in that the directional decomposition filters are applied directly to the image. In their method the compression ratio varies from image to image. The filter design depends on many factors, which in turn affect the compression ratio. Therefore Ikonomopoulos and Kunt state that these parameters should be tuned to the particular image since the quantity, content, and structure of the edges in the image determine the compression obtained. Despite these factors, compression ratios on the order of 64 to 1 are reported with good image quality.

The morphological filtering technique of Zhou and Venetsanopoulos [1992] separates the features into what are called FMP images. It is these FMP images to which traditional directional decomposition techniques are applied in order to perform the coding process. The compression ratios reported by this method are reasonable at around 20 to 1.

5. VISUAL PATTERN-BASED APPROACHES

Visual pattern-based techniques operate by coding the image using a small set of visual patterns that are localized subimages containing visually important information. The method bears similarities to both block truncation coding (BTC) [Delp and Mitchell 1979] and vector quantization (VQ) [Gray 1984]. However, there are two main differences: there is no training phase required as there is in the majority of VQ techniques, with the exception of Lattice VQ, since the visual patterns are predefined; and the assignment of visual patterns is not based on an error criterion, therefore no complicated search is required. The visual patterns used may vary with the application, but most correlate with psychophysically derived visual response characteristics.

Chen and Bovik [1990] developed a coding technique using visual patterns. In this method a subset of visual patterns is designed using knowledge of the HVS. The image is then coded by considering single blocks at a time and determining onto which subspace the image block should be mapped. For each block (a block may be, for example, a 4 × 4 region in the image), the coding scheme consists of computing the mean block intensity $\mu_{i,j}$, the local gradient measurement $|\nabla I|$, and the block orientation $\angle \nabla b_{i,j}$. Each block is then categorized according to the mean value of the block: if $(|\nabla b|_{i,j})^2 \leq (|\nabla I|_{\min})^2$, then the block is considered a uniform block; otherwise it is an edge block. In either case, the mean of the block is quantized and stored along with an additional bit indicating whether the block is uniform or edge. If the block is an edge block, then its measured orientation maps the

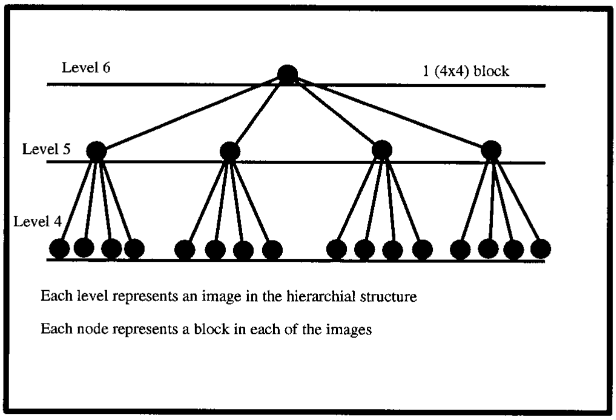
Level 6
1 (4x4) block
Level 5
Level 4
Each level represents an image in the hierarchial structure
Each node represents a block in each of the images

Figure 6. Quadtree for three levels of AVPIC.

image block to some number of edge subspaces. A unique mapping among those at a given orientation is achieved by selecting the pattern whose polarity distribution is closest to that of the image block. The index of the selected pattern is then stored in place of the image block.

Silsbee and Bovik [1991] developed a coding method that is an extension of the visual pattern-based image coding. Their method is an adaptive version of the hierarchical extension to visual pattern image coding HVPIC, which is referred to as AVPIC. In HVPIC the starting point is an N by N image, where N is a power of 2. A series of subsampled progressively smaller images is initially generated, thus forming a resolution pyramid with $n + 1$ levels, the original image being the bottom level. Images are encoded in the reverse order from generation. After each subimage has been encoded, it is immediately decoded. The decoded version is then expanded and subtracted from its original in order to produce the residual or error image. It is this residual image that is to be coded using standard VPIC methods. At this point no loss of information has been introduced into the image since the original may be exactly reconstructed by adding the error images.

The hierarchical structure inherent with HVPIC is useful in identifying areas of constant intensity within the image. Since these areas can be efficiently represented by single values, they require fewer bits to encode. Silsbee and Bovik [1991] suggest that a test is required, after each pyramid level has been encoded, to determine whether a block has been accurately described. If it has, the particular block need not be examined further in any finer detail. This test has been incorporated into a quadtree structure that has been constructed using the bottom-up approach. Each bit in the quadtree represents one block at some level of the pyramid (Figure 6).

The notation used for this construction is as follows.

$Q^{(k)}$ represents the quadtree level corresponding to the image in the pyramid $I^{(k)}$

$Q^{(k)}_{i,j}$ represents the ith bit of the jth column of the kth level of the quadtree.

Each bit at the lowest level of the tree is labeled as a terminal 0, assuming that no edge was detected and no large correction of mean occurred for the corresponding block. If the block was an edge, or a large mean resulted, the node is set to 1; that is,

$$Q^{(0)}_{i,j} = 0 \text{ if no significant detail occurs at block}$$
$$Q^{(0)}_{i,j} = 1 \text{ otherwise.}$$

Note that a large correction of the current mean value indicates that the block differs from its neighbors and hence some detail is present.

The remainder of the quadtree is then constructed by examining the children associated with a particular parent. If the parent's children were terminals (i.e., 0, and no edge or mean correction occurred at the block), the parent is marked as a terminal; otherwise it is set to 1. In relation to the images in the pyramid, this process is equivalent to examining the four blocks in the previous pyramid level, which is of finer resolution, and determining whether these blocks were of constant intensity. If so, then it follows that the equivalent block, covering each of these four blocks, at a lower resolution would be of constant intensity also. It would therefore be redundant to code the details at the finer level if the next resolution level up would suffice.

The preceding process continues until the entire quadtree has been constructed. At this point the quadtree may be used to determine which blocks in which pyramid levels should be stored/transmitted. Only branches and leaves in the tree that are not marked as terminal are processed, since the blocks at these nodes contain the image detail.

Using the standard VPIC method reported by Chen and Bovik [1990], the compression ratios obtained are reasonable at between 10 and 20 to 1. These ratios are comparable to both VQ and BTC on which the VPIC method is based. The decoded image quality is also reported to be similar to that attained by VQ. The method provides the facility to preset the compression ratio within a specified range. The adaptive version of VPIC, AVPIC, presented by Silsbee and Bovik [1991] is a direct extension of Chen and Bovik's [1990] work. It implements a standard VPIC technique, but instead of applying it directly to the image, they first create a residual pyramid structure and then apply VPIC methods to this. In addition, further redundancy is exploited by suggesting that to code and store each of the blocks in all levels of the pyramidal structure would be wasteful in terms of bit allocation. It is suggested by Silsbee and Bovik [1991] that it is sufficient to encode only the blocks in one level, assuming that this block accurately represents the area in the image at all subsequent finer resolutions. The compression ratios obtained by the AVPIC approach do make an improvement on the original VPIC method, values being in the range 16 to 33 to 1 for various images. However, in some cases where the image contains large amounts of fine detail, the addition of the quadtree section as a decision stage becomes redundant and HVPIC produces better results.

6. SEGMENTATION-BASED APPROACHES

In general, a technique that is considered to be segmentation-based has three main steps:

—preprocessing,

—segmentation, and

—coding of contours and texture components.

This section describes some of the image-coding techniques based on segmentation. In addition, overviews of the various techniques used in the coding techniques, for example, preprocessing, segmentation, coding of contours, and texture representation, are also presented in Sections 6.1 through 6.5.

6.1 Preprocessing

The purpose of preprocessing is to eliminate small regions within the image and remove noise generated in the sampling process. It is an attempt to model the action of the HVS and is intended to alter the image in such a way that the preprocessed image will more accurately resemble what the human brain processes. Various methods are used to preprocess the image, all derived from properties of the HVS (see Section 2). Two properties commonly used are Weber's Law and the modulation transfer function (MTF) [Jang and Rajala 1991, 1990; Civanlar et al. 1986].

Marqués et al. [1991] suggest the use of Steven's Law. This takes into account the greater sensitivity of the HVS to gradients in dark areas rather than light ones. For example, if B is the perceived brightness and I the stimulus intensity then:

$$B = K.I^{n}.$$

Therefore, by preprocessing according to Steven's Law, visually homogeneous regions will not be split unnecessarily and heterogeneous dark areas will not be falsely merged.

In addition, the inverse gradient filter [Wang and Vagnucci 1981] has also been implemented in order to give a low-pass response inside a region and an all-pass response on the region contour [Kwon and Chellappa 1993; Kocher and Kunt 1983]. This is an iterative scheme that employs a 3 × 3 mask of weighting coefficients. These coefficients are the normalized gradient inverse between the center pixel and its neighbors. If the image to be smoothed is expressed as an $n \times m$ array where $p(i, j)$ is the gray level of the pixel at (i, j) with $i = 1, \ldots, n$ and $j = 1, \ldots, m$, the inverse of the absolute gradient at (i, j) is defined as

$$\delta(i, j{:}k, l) = \frac{1}{|p(i+k, j+l) - p(i, j)|},$$

where $k, j = -1, 0, 1$, but k and l are not equal to zero at the same time. This means that $\delta(i, j : k, l)$s are calculated for the eight neighbors of (i, j); this is denoted the vicinity $V(i, j)$. If $p(i + k, j + l) = p(i, j)$, then the gradient is zero and $\delta(i, j : k, l)$ is defined as 2.

The proposed 3 × 3 smoothing mask is defined as

$$W(i,j) = \begin{bmatrix} w(i-1, j-1) & w(i-1, j) & w(i-1, j+1) \\ w(i, j-1) & w(i, j) & w(i, j+1) \\ w(i+1, j-1) & w(i+1, j) & w(i+1, j+1) \end{bmatrix}$$

where $w(i, j) = \frac{1}{2}$ and $w(i + k, j + l) = \frac{1}{2}[\Sigma_{V(i,j)}(i, j : k, l)]^{-1}\delta(i, j : k, l)$ for k, $l = -1, 0, 1$ but not 0 at the same time.

The smoothed image is then given as

$$\hat{p}(i,j) = \sum_{k=-1}^{1} \sum_{l=-1}^{1} w(I+k, j+l)p(i+k, j+l).$$

6.2 Segmentation Techniques

This section introduces some of the commonly used methods for segmenting an image. Segmentation groups similar pixels into regions and separates those pixels that are considered dissimilar. It may be thought of as representing an image by a disjoint covering set of image regions [Bigger et al. 1988]. Many segmentation methods have been developed in the past [Pal and Pal 1993; Haralick 1983] and it is generally the segmentation method that categorizes the coding technique.

6.2.1 *Region Growing.* Region growing is a process that subdivides a (filtered) image into a set of adjacent regions whose grey-level variation within the region does not exceed a given threshold.

The basic idea behind region growing is that, given a starting point within the image, the largest set of pixels whose grey level is within the specified interval is found. This interval is adaptive in

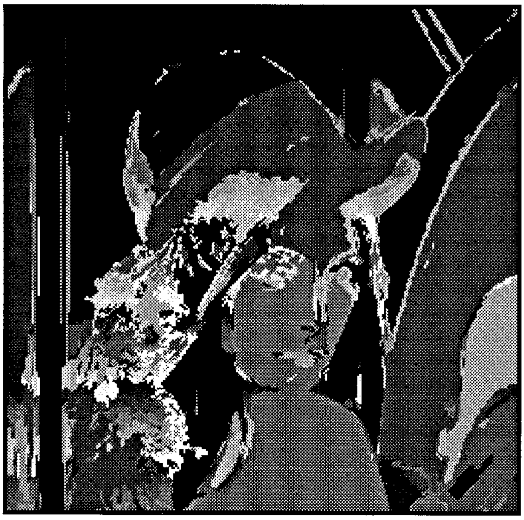

Figure 7. Image segmented via split-and-merge technique. Here 1405 regions have been produced.

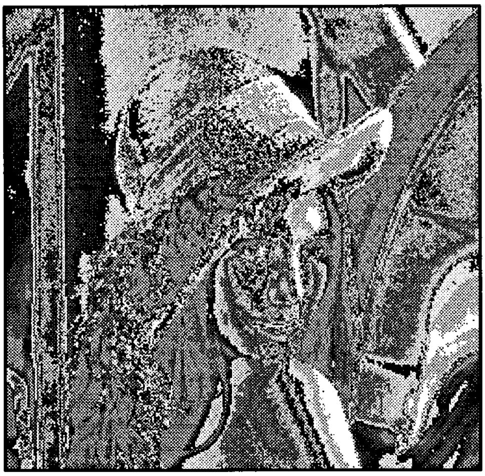

Figure 8. Test image segmented via k-means clustering. Here 20 initial clusters have been chosen.

that it is allowed to move higher or lower on the grey scale in order to intercept the maximum number of pixels.

6.2.2 *Split and Merge.* Split and merge algorithms [Pavlidis 1982] segment the image into sets of homogeneous regions (see Figure 7). In general, they are based around the quadtree [Samet 1989a] data structure. Initially the image is divided into a predefined subdivision, for example, 64 divisions; then, depending on the segmentation criteria, adjacent regions are merged if they have similar grey-level variations or a quadrant is split further if large variations exist.

6.2.3 *k-Means Clustering.* k-means clustering is a segmentation method based on the minimization of the sum of squared distances from all points in a cluster to a cluster center. First k initial cluster centers are taken and the image vectors are iteratively distributed among the k cluster domain. New cluster centers are computed from those results in such a way that the sum of the squared distances from all points in a cluster to a new cluster center is minimized (see Figure 8).

6.2.4 *Graph Theory.* A number of image-segmentation techniques are based on the theory of graphs and their applications [Morris et al. 1986]. A graph is composed of a set of *vertices* connected to each node by *links*. In a weighted graph the vertices and links have weights associated with them. Each vertex need not necessarily be linked to every other, but if they are, the graph is said to be complete. A partial graph has the same number of vertices but only a subset of the links of the original graph. A *spanning tree* can be referred to as a partial graph. A *shortest spanning tree* of a weighted graph is a spanning tree such that the sum of its link weights is a minimum for many possible spanning trees.

In order to analyze images using graph theory, the original image must be mapped onto a graph. The most obvious way to do this is to map every pixel in the original image onto a vertex in the graph (Figure 9). The link weight can be taken as the absolute difference between the vertex weights it joins. Segmentation using graphs can be achieved by using shortest spanning trees (SSTs) [Cheriton and Tarjan 1976]. An informal definition of an SST is that for any

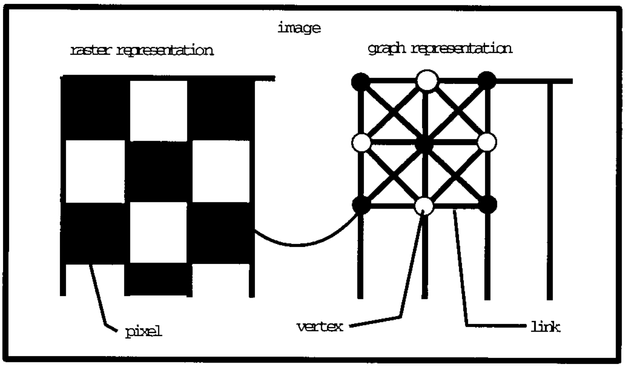

Figure 9. Graph theory: mapping from raster to graph format.

tree in a forest, if a link with the lowest weight connecting that tree to another is added, then the link will be in the SST. If the forest forms part of the SST, then the complete SST can be found by successively adding the new links to it.

In order to segment an image, the most similar vertices are located by finding the lowest-value link weights. This can be done for the entire graph by finding an SST. Once the SST has been found, segmentation is achieved by cutting the SST at its highest link weights, thus forming partitions that differ from neighbors by a maximum amount. Further regions may be located by making more cuts in the SST.

6.2.5 *Fractal Dimension.* The fractal dimension D is a characteristic of the fractal model [Mandelbrot 1982] that is related to properties such as length and surface of a curve. It provides a good measure of the perceived roughness of the surface of the image. Therefore, in order to segment the image, the fractal dimension across the entire image is computed (Figure 10). Once the fractal dimension for the entire image has been computed, various threshold values can be used to segment the original image according to its fractal dimension.

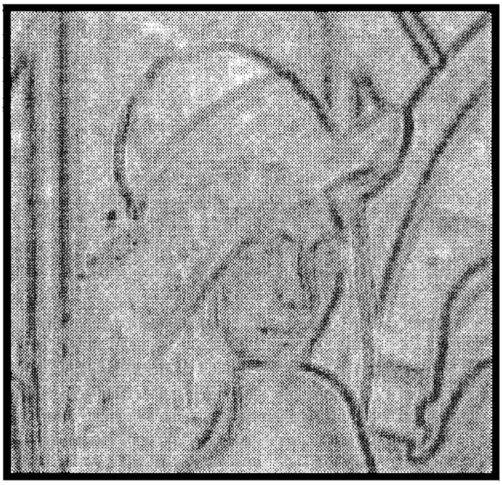

Figure 10. Fractal dimension image.

6.3 Polynomial Approximation

The regions produced by the segmentation process identify areas in the original image whose grey-level variation is similar. In order to code these areas efficiently they are often represented by an nth-order polynomial. The basic idea behind polynomial fitting is that an attempt is made to model the grey-level variation within a region by an order-n polynomial while ensuring that the mean-squared error between the predicted value and the actual is minimized. An order-0 polynomial would ensure that each pixel in the region was represented by the average intensity value of the region. An order-1 polynomial is represented by:

$$z = a + bx + cy,$$

where z = new intensity value at (x, y).

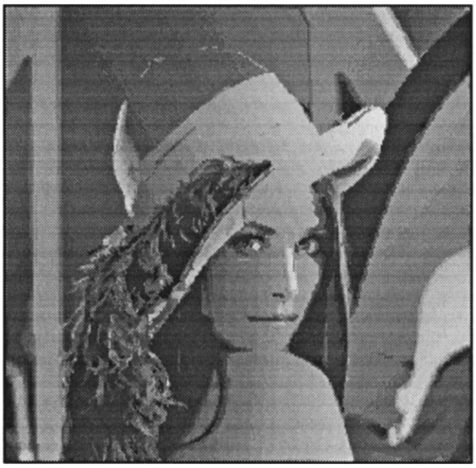

Figure 11. Test image segmented using split and merge.

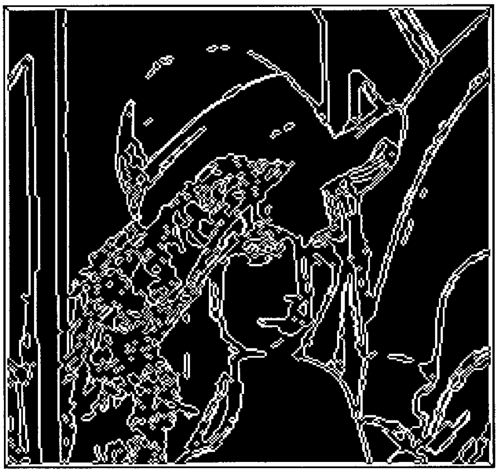

Figure 12. Contours of regions obtained after split and merge.

Figure 11 shows a typical image with regions represented by an order-1 polynomial.

6.4 Coding of Image Contours

In order that the modeled regions within the image may be decoded accurately, their position in the overall image must be recorded. Obviously it would be inefficient to record every pixel position in the region. Therefore more efficient methods have been developed. Each region is generally a closed structure and as such has a defined outline or contour (see Figure 12). One such method often cited in relevant literature for representing these contours is Freeman chain coding.

Freeman chain coding [Freeman 1976, 1974, 1961] represents the given contour by an initial starting position and a set of codes representing relative positions. The Freeman chain codes are shown in Figure 13. The basic principle behind this coding process is this: an initial starting point on the curve is stored via its (x, y) coordinates. After this, the position of the next point on the curve is located. This position can be in one of eight locations, as illustrated in Figure 13. If, for example, the next position is $(x, y - 1)$, then the pixel happens to lie in position 2 according to Freeman, and hence a 2 is output. This pixel is then updated as the current position and the coding process repeats. The coding is terminated when either the original start point has been reached (closed contour) or no further points on the curve can be found (open contour). The code is an efficient repre-

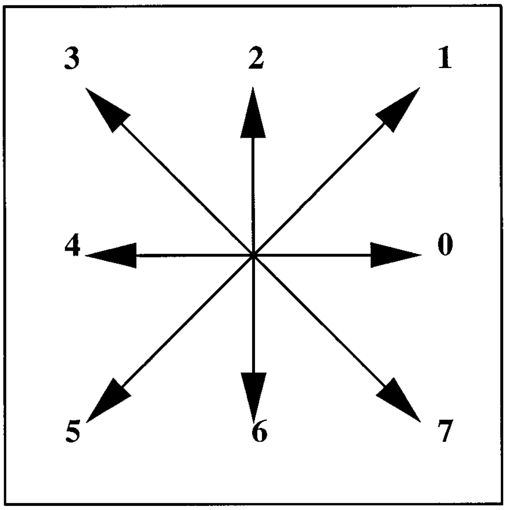

Figure 13. Possible directions of movement for a pixel at the center. Numbers then represent the Freeman chain codes.

sentation of the contour since at least 3 bits are required to store each code in the chain; however, further gains can be achieved by applying entropy coders or lossy contour-coding techniques to the contours.

6.5 Region-Growing Techniques

Kocher and Kunt [1983] presented a technique based on region growing called contour texture modeling. The original image is preprocessed by the inverse gradient filter [Wang and Vagnucci 1981] to remove picture noise in preparation for the region-growing process. After the growing process, a large number of small regions are generated, some of which must be eliminated. This elimination is necessary in order to reduce the number of bits required to describe the segmented image, and thus increase the compression ratio. It is performed on the basis of removal of small regions and merging of weakly contrasting regions. Regions whose grey-level variations differ slightly are considered weakly contrasting.

Once the segmentation process is complete, two components arise that must be coded, as illustrated in Figures 7 and 12:

—contours describing regions, and

—texture within regions.

Contour coding is performed in stages. First, it is possible that the contours describing two neighboring region boundaries may touch, so that describing each of these contours fully would incorporate some redundancy. Therefore the contour pixels that touch, or describe two regions, are removed from one of the contours. Second, the opened contours are approximated by a succession of connected segments whose sizes depend on the contour complexity. Texture coding is achieved by representing the grey-level variation within the region by an nth-order polynomial function.

This representation by itself does not produce a very natural-looking image upon reconstruction. Therefore the noise removed by the preprocessing stage is added back in the form of pseudorandom noise.

Civanlar et al. [1986] present an HVS-based segmentation coding technique. In this a variation of the centroid linkage-region-growing algorithm [Haralick 1983] is used to segment the image after preprocessing. In a centroid linkage algorithm the image is scanned in a set manner, for example, left to right or top to bottom. Each pixel is compared to the mean grey-level value of the already partially constructed regions in its neighborhood and if the values are close enough, the pixel is included in the region and a new mean is computed for the region. If no neighboring region has a close enough mean, the pixel is used to create a new segment whose mean is the pixel value. In Civanlar et al.'s [1986] technique, the centroid linkage algorithm previously described applies. However, if the intensity difference is less than an HVS visibility threshold, the pixel is joined to an existing segment. If the intensity differences between the pixel and its neighbor segments are larger than the thresholds, a new segment is started. The work by Kocher and Kunt [1983] provides the facility to preset the approximate compression ratio prior to the operation. This is achieved by setting the maximum number of regions to be generated after the region-growing process. The results obtained via their method are good both in terms of reconstructed image quality and compression ratio. However, they point out that the performance of their technique in terms of image compression and quality is optimal for images that are naturally composed of a small number of large regions.

Civanlar et al. [1986] report good image quality and compression ratios, comparable to those in Kocher and Kunt [1983]. The basic technique presented by Civanlar et al. [1986] is similar to that of Kocher and Kunt [1983]; however, the methods employed to code the texture and contour components are not apparent in the latter paper.

6.6 Split-and-Merge-Based Techniques

Kwon and Chellappa [1993] present a technique based on a merge and threshold algorithm—segmentation-based image coding [Kunt et al. 1987]. After the image has been preprocessed, the intensity difference between two adjacent regions is found. If this difference is less than or equal to k, which has been initialized to 1, the regions are merged and the average of the intensities is computed. A histogram of the merged image is computed and if separable clusters exist, the preceding steps are repeated; otherwise, the original image is segmented by thresholding the intensity clusters. When the overall process is complete, the regions obtained may be represented by an nth-order polynomial.

The preceding method of segmentation extracts only homogeneous regions and thus for textured regions a large number of small homogeneous regions are generated. In terms of image coding, it is more efficient to treat textured areas as one region than as several small regions. Therefore, in addition to the homogeneous region extraction scheme, textured regions are also extracted and combined with the results of the uniform region segmentation.

Multiple features are used in the texture-extraction process, along with the recursive thresholding method using multiple 1D histograms. First the image is regarded as one region. A histogram is then obtained within each region of the features to be used in the extraction process. The histogram showing the best clusters is selected and this corresponding region is then split by thresholding. These steps are repeated for all regions until none of the histograms exhibit clustering.

Final segmentation is achieved by labeling the extracted uniform regions. If the area of such a region is covered by more than 50% of a textured region of type X, then the uniform region is labeled as a textured region of that type. Adjacent uniform regions are merged with a texture region if they show at least one texture feature similar to the corresponding texture region. In terms of coding, uniform regions are represented by polynomial reconstructions and texture regions by a texture-synthesis technique using the Gaussian-Markov random field (GMRF) model [Chellappa et al. 1985]. Encoding the image therefore involves storing information about the contours of the regions, polynomial coefficients of the uniform regions, GMRF parameters for textured regions, and a means of identifying each region. Variable bits are allocated for each component.

Another approach based on a split-and-merge algorithm is that of Cicconi and Kunt [1993; Cicconi et al. 1994]—symmetry-based image segmentation. Segmentation is performed by initially clustering the image using a standard k-means clustering algorithm (Figure 8).

After the initialization process, any cluster, along with its associated vectors, is discarded if the membership measure of that cluster is below a specified threshold. Clusters exhibiting large variances are split in two, and clusters that are close enough to each other are merged. This sequence is repeated until no clusters are discarded, split, or merged, or if the maximum number of iterations has been reached. Once the image has been segmented into homogeneous areas, an attempt to further reduce the redundancy inside the regions is implemented by looking for symmetries within the regions. In order to do this the medial-axis transformation (MAT) [Pavlidis 1982] is used for shape description. The MAT is a technique that represents, for each region, the curved region descriptor. The MAT corresponds closely to the skeleton that would be produced by applying sequential erosion to the region. Values along the MAT represent the distance to the edge of the region and can be used to find minimum and maximum widths. The histogram of the values will give the variation of the width. Once the MAT has been found, a linear prediction of each pixel in one side of the MAT can be constructed from pixels symmetrically chosen in the other side.

Coding of the segmented image is performed in two stages: contour coding and texture coding. Since the MAT associated with a region can be reconstructed from a given contour, only contours have to be coded. Texture components in one part of the region with respect to the MAT may be represented by a polynomial function. However, to represent the polynomial coefficients precisely requires a large number of bits. Therefore, the proposed method suggests defining the positions of six pixels, which are found in the same way for all regions, then quantizing these six values. These quantized values allow the unique reconstruction of the approximating second-order polynomial [Cicconi and Kunt 1993].

Both of these techniques are similar in that they employ a split-and-merge algorithm to segment the original image. However, Kwon and Chellappa [1993] state that better compression ratios may be obtained by segmenting the image into uniform and textured regions. These regions may be coded separately and, in particular, the textured regions may be more efficiently represented by a texture-synthesis method, such as a GMRF model, as opposed to representing the textured region with many small uniform regions. Cicconi and Kunt's [1993; Cicconi et al. 1994] method segments the image into uniform regions and, in addition, they propose to exploit further redundancy in these regions by identifying symmetry within the regions. The grey-level variation within each of the uniform regions is represented using polynomial modeling. As stated, Cicconi and Kunt further developed a method for reducing the storage requirements for the polynomial coefficients.

Despite the different methods used to represent both the contours and the grey-level variations within the regions, both methods report similar compression ratios.

6.7 Tree/Graph-Based Techniques

Biggar et al. [1988] developed an image coding technique based on the recursive shortest spanning tree (RSST) algorithm [Morris et al. 1986]. The RSST algorithm maps the original image onto a region graph so that each region initially contains only one pixel. Sorted link weights, associated with the links between neighboring regions in the image, are used to decide which link should be eliminated and therefore which regions should be merged. After each merge, the link weights are recalculated and resorted. The removed links define a spanning tree of the original graph.

Once the segmentation is complete, the spanning tree is mapped back to image matrix form, thus representing the segmented image. The regions generated are defined by coding the lines that separate the pixels belonging to different regions. The coded segmented image has three sources: a list of coordinates from which to start tracing the edges, the edge description, and a description of the intensity profile within each region.

The intensity profile within the region could be represented as a simple flat intensity plateau; however, it has been suggested by Kunt et al. [1985] and Kocher and Kunt [1983] that a better result is achievable by higher-order polynomial representation. Biggar et al. [1988] suggest that to embed the polynomial fitting procedure at each stage of the region-merging process, as Kocher and Kunt [1983] do, would be computationally too expensive. Therefore in this case a flat intensity plane is used to generate the regions and polynomials are fitted after the segmentation is complete.

The edge information is extracted from the segmented image using the algorithm for thin line coding by Kaneko and Okudaira [1985].

A similar technique based on the minimum spanning forest, MSF, is reported by Leou and Chen [1991]. The segmentation and contour coding is performed exactly as described by Biggar et al. [1988]; however, the intensity values within a segmented region are coded with polynomial representation. Here a texture-extraction scheme is used, based on the assumption that lights are cast overhead on the picture and that the grey values vary according to the distance to the corresponding region centroid. After texture extraction, the regions have a high pixel-to-pixel correlation. Therefore, for simplicity and efficiency, a polynomial representation method is used to encode the texture. This is achieved by representing any row of the image by a polynomial.

A different graph-theory approach is presented by Kocher and Leonardi [1986] based on the region adjacency graph (RAG) data structure [Pavlidis 1982]. The RAG is again a classical map graph with each node corresponding to a region and links joining nodes representing adjacent regions. The basic idea of the segmentation technique is that a value that represents the degree of dissimilarity existing between two adjacent regions is associated with a graph link. The link that exhibits the lowest degree of dissimilarity is removed and the two nodes it connects are merged into one. This merging process is repeated until a termination criterion is reached. Once complete, the RAG representation is mapped back to the image matrix form, and thus a segmented image is created.

The segmented image is coded using a polynomial representation of the regions and gives very good compression ratios.

All of the preceding methods are based on similar graph structures that enable the image to be mapped to the graph form in order to perform segmentation. The techniques by Biggar et al. [1988] and Kocher and Leonardi [1986] both model the texture within the image via a polynomial modeling method. However, Kocher and Leonardi [1986] report on compression ratios of much larger proportions than Biggar et al. Leou and Chen [1991] implement a segmentation technique identical to that presented by Biggar et al.; however, Leou and Chen point out that better compression ratios can be achieved by first performing a texture extraction process and then modeling the texture by polynomials as opposed to polynomial functions. The compression ratio achieved via this method does improve on that reported by Biggar et al. [1988].

6.8 Fractal-Based Techniques

In the previous sections various methods for image segmentation have been suggested that lend themselves to efficient compression of the image. Most of these techniques segment the image into regions of homogeneity and thus, when a highly textured image is encountered, the result of the segmentation is many small homogeneous regions.

Jang and Rajala [1990, 1991] suggest a technique that segments the image in terms of textured regions. They also point out that in many cases previous segmentation-based coding methods are best suited to head- and shoulder-type images and that results obtained from complex natural images are often poor.

In their technique the image is segmented into textually homogeneous regions as perceived by the HVS. Three measurable quantities are identified for this purpose: the fractal dimension, the expected value, and the just noticeable difference. These quantities are incorporated into a centroid linkage region-growing algorithm that is used to segment the image into three texture classes: perceived constant intensity, smooth texture, and rough texture. An image coding technique appropriate for each class is then employed.

The fractal dimension value D is then thresholded to determine the class of the block. The following criteria are used to categorize the particular block under consideration:

$D < D^1$	perceived constant intensity
$D^1 < D < D^2$	smooth texture
$D > D^2$	rough texture.

After this segmentation process the boundaries of the regions are represented as a two-tone image and coded using arithmetic coding [Witten 1987; Nelson 1992]. The intensities within each region are coded separately according to their class. Those of class 1, perceived constant intensity, are represented by the average value of the region; class 2, smooth texture, and class 3, rough texture, are encoded by polynomial modeling. It should be noted from the description in section 6.3 that polynomial modeling leads to some smoothing and hence may not be useful for rough texture. Therefore, it is not clear why Jang and Rajala choose this method of representation for the class-3 regions.

Each of the various segmentation techniques used group pixels according to some criterion, whether it is homogeneity, texture, or pixels within a range of grey-level values. The problem that arises after segmentation is how to efficiently code the grey-level values within the region. The most basic representation of grey level within a region is by its mean value. This will result in a good compression, especially if the region is large; however, the quality of the decoded image will be poor. In most cases grey-level variation is approximated by a polynomial function of order two. The results obtained by polynomial approximation can be visually poor, especially for highly textured images. It is for this reason that more researchers are representing highly textured regions by texture synthesis techniques such as GMRF. These methods do not gain over the compression ratios obtained using polynomial approximation, but the quality of the reproduced image is claimed to be improved [Kwon and Chellappa 1993]. Another approach used to encode the grey level variation is by representing the variations by polynomials, as per Leou and Chen [1991]. This method is of similar computational complexity, but the results in terms of compression ratio and image quality are claimed to be better than polynomial reconstruction.

As stated by Jang and Rajala [1990, 1991], many of the preceding segmentation-based techniques are not sufficient when the input image is a natural one, that is, an image of a real scene. These may be images that contain highly textured regions and when segmented using conventional methods, the resulting textured region is composed of a large number of small regions. These small regions are often merged or removed in order to increase the compression ratio and as a result the decoded image looks very unnatural. Therefore Jang and Rajala [1990, 1991] used the fractal dimension to segment the image. This ensures that the region is segmented into areas that are similar in terms of surface roughness, this being the result of visualizing the image in three dimensions with the third dimension being grey-level intensity. However, once segmented into regions of similar roughness, the method employed to code the identified areas is similar to that of traditional segmentation-based coding methods, that is, polynomial modeling. This polynomial modeling, as reported by Kwon and Chellappa [1993], does not suffice for the representation of highly textured regions and they suggest the use of texture synthesis. Therefore, it may be concluded that a better segmentation-based coding method might employ the fractal dimension segmentation approach coupled with texture synthesis for textured region representation. Table I summarizes the methods used in the techniques that employ a segmentation algorithm as part of the coding process.

Table I. Results from Segmentation-Based Methods

Technique	Texture coding method
HVS-based segmentation [Civanlar et al. 1986]	polynomial function
segmentation-based [Kwon and Chellappa 1993]	polynomial & GMRF
symmetry-based coding [Cicconi et al. 1994]	polynomial function
RSST-based [Bigger et al. 1988]	polynomial function
MSF-based [Leou and Chen 1991]	polynomial function
RAG-based [Kocher and Leonardi 1986]	polynomial function
fractal dimension [Jang and Rajala 1991]	polynomial function

7. CONTOUR-CODING-BASED APPROACHES

This section presents two image coding techniques that use the notion that contours can be used solely in the coding process. It was shown in Section 6 that, in general, an image is composed of regions and that each region can be described by its outer contour and some representation of the grey level in the region. The techniques presented in this section make no attempt to model the content of a region; instead, they consider the image as composed of many separate contours, and it is these contours and their properties that are used in the coding process.

7.1 Lossy Coding

Marshall [1992, 1987] presents a paper based on applications of image contours to various aspects of image processing. There are two aspects of image processing that until recently have been researched separately, namely, analysis and compression. The idea of using contours to record constant intensity was suggested by Graham [1967] and, as Marshall points out, could prove to be as valid a coding method as a raster scan approach. The coding algorithm proposed by Marshall has four main steps: conversion of the image to contours using a contour-tracing algorithm, efficient representation of the contours, recreation of contours from their descriptions, and image reconstruction by interpolation between contours.

As with all sampling processes, an error is introduced that may be reduced by decreasing the sampling distance. Therefore in this case the spacing between samples in the contour tracing is reduced to a reasonable amount. A large number of small contour segments are produced by the tracing procedure and, in order to achieve better compression, these were removed as they had little significance on the reconstructed image. Remaining contours are coded using a 4- or 8-way chain code [Kaneko and Okudaira 1985; Freeman 1976, 1974, 1961]. This involves coding the starting point of the contour segment and the directions of the contour points of the remainder of the segment. Various methods can be used to code the results of the chain coding procedure, for example Huffman [1952], Fourier, or vector coding.

7.2 Lossless Compression

The common feature of all the preceding techniques is that compression is achieved by ignoring or removing some of the original image data and thus the technique can be said to result in lossy compression. In these cases great effort has been devoted to ensuring that the loss of image detail is minimized in terms of human perception. This argument is perfectly valid when the image domain is that of natural images, and compression is necessary for, say, multimedia applications. However, the question arises of the validity of these methods when applied to medical images. Wittenberg [1993] states that there are legal reasons why lossy compression of medical images is not acceptable. He suggests that a radiologist cannot assume he or she has made a correct diagnosis based on lossy compression and for this reason in some countries, for example, Germany, it is illegal to use lossy compression for medical images.

Millar and Nicholl [1994] present a new lossless compression technique based on contour coding for the compression of images that contain at least one main feature, that is, medical images. In this method the image is processed initially in order to identify a single feature in the image. The path of this major feature is then recorded and is used to direct the path of the coding technique. Coding of the image consists of a run-length-type coding the path of which is dictated by the outline of the feature. Once the current outer contour has been encoded, it is removed and the coding continues on the next inner contour. Each subsequent contour in the image follows the inner path of the previous contour exactly. In terms of image reconstruction, each of the contours contained in the image can be reconstructed from the path of the outer contour. Hence only the path of this outer contour must be stored along with the run-length tuples.

Both of these techniques implement a contour-based coding scheme. However, Marshall's [1992, 1987] technique treats each contour within the image separately. It is assumed that the image is composed entirely of small contours. This method could have been implemented in a lossless mode; however, due to the number of small contours generated, loss is introduced by assuming a sampling size greater than one pixel width.

In contrast, Millar and Nicholl [1994] present a lossless coding technique that identifies the major feature within the image and use this to perform a path-directed run-length coding. This coding technique achieves compression ratios on the order of 1.5 to 1 for typical magnetic resonance images [Cho et al. 1982; Morris and Peter 1986; Kean and David 1986]. In terms of compression ratio this does not compare to the previously reported lossy techniques. However, the technique does outperform standard lossless first-generation techniques that were applied to the same images. This technique not only provides some compression but has the advantage that structural information is stored as part of the coding process. Thus image manipulation, such as translation and rotation, may also be performed on the compressed data.

8. SUMMARY

Image coding is, in general, a two-stage process commonly stated as:

$$\text{modeling of the image data} \\ + \text{coding of the model parameters.}$$

In this article various second-generation image coding techniques have been presented. These techniques have been grouped in terms of their modeling stage, which for second-generation coding tends to be a modeling process that mimics a property of the HVS.

In the segmentation-based techniques, the modeling stage has been based on the fact that the eye is good at identifying regions that appear similar and grouping them accordingly. The segmentation algorithm employed in the overall technique is responsible for separating the original image into its various regions. The contents of these regions will then be represented via some criterion, that is, polynomial approximation or a texture-representation scheme, such as GMRF. These modeling schemes can represent the contents of a region very accurately and hence produce good-quality reproduced images. However, they do rely heavily on the initial segmentation. If the image has been segmented unrealistically, for example, if too few regions have been produced, then the resulting decoded image will be visually poor. In general, most of the segmentation techniques perform well for images that contain large areas of similar texture or intensity. However, it has been pointed out by Jang and Rajala that the performance for real-world images, such as landscape images, is poor. It was for this reason that they employed the fractal dimension as a segmentation method, and it can be seen that for a real-world image this method does identify larger regions than, for example, region growing.

Visual-pattern-based coding techniques make use of the fact that the eye can decompose the overall image into a set of smaller "visual patterns." The initial work in this field was presented by Chen and Bovik [1990], and their work resulted in compression ratios of similar size to that of VQ (on which the technique is based). Further improvements were reported by Silsbee and Bovik [1991], who implemented a hierarchical version of their predecessors. Compression ratios reported by Silsbee and Bovik are comparable to those achieved with some of the early segmentation-based approaches.

Directional filtering has been employed as a modeling scheme in order to exploit the fact that the eye is composed of direction-sensitive neurons. The scheme can be applied directly to the original image, as Ikonomopoulos and Kunt [1985] do, with the resulting directional images being coded in traditional ways, for example, Huffman coding to produce good-quality decoded images at reasonable compression ratios. Alternatively, the directional filtering stage may be applied to the output of a feature separation stage, as in Zhou and Venetsanopoulos [1992].

Multiscale techniques take the original image and subsample it in order to produce various levels of image detail at progressively smaller details. This idea exploits the fact that the eye is capable of identifying such levels of progressively finer details in the image. By creating sets of progressively smaller subsampled images, a mechanism is developed that can be used to identify common features in the image that are present at various levels of detail.

The final grouping for the modeling schemes is based on the eye's ability to identify edges in the images. Contour coding considers that the image to be coded is composed of many line segments or closed contours that spiral towards the center of the main body of the image. These methods identify the line segments, code their position and description, and then use this description to code the image via standard coding methods, for example, path-directed run-length schemes.

As stated previously, second-generation image coding is, in general, lossy in nature. The majority of the schemes discussed previously compress the image by removing some of the visual redundancy inherent in the image. As indicated, the coding schemes are based on some property of the human visual system and hence the redundancy is removed on the principle that if we cannot see the redundant information then it may be successfully removed without impairing the "look" of the image. The recent explosion in multimedia PCs and applications has opened the way for such possibilities as image databases and photo CDs. These applications may contain many thousands of digital images and hence massive storage space is required. At present a vast majority of "commercial" image storage software employs techniques such as JPEG and fractal compression [Waite 1990; Jacquin 1993; Fischer 1985] (this is a separate technique based on VQ principles and fractals and is not related to Jang and Rajala's method). These methods can produce good compression ratios but often at the expense of image quality. Second-generation coding removes only visually redundant information from the image and as such is ideal for such applications as image database storage. However, the problem arises of which technique and hence which particular property of the HVS to use.

One drawback with the majority of second-generation techniques is that they are inherently lossy. Despite the fact that the loss cannot be detected by the eye, this is not acceptable for those cases that do not tolerate any loss, for example, diagnostic medical imaging. In these cases we cannot apply such techniques that merge small regions and remove insignificant ones as in this case the region may actually be of diagnostic value. In medical imaging, similar applications to those in multimedia are found, such as storage in an image database or picture archive and communication system (PACS). At present many of these images are stored using standard file compression utilities, Unix compress, for example. Therefore it is vital that lossless techniques are continually developed.

All the techniques presented here have their relative merits and drawbacks. They all exhibit similar computational complexity and in general are linear in their compress/decompress cycle; that is, similar complexity is involved in both the compression and decompression phases. The choice of algorithm will be very much influenced by nonrelated items such as ready availability of a particular algorithm and/or prewritten software, that is, image-processing libraries. The techniques presented in this article cover a diverse selection, with each group of techniques being based on one different aspect of the HVS, which makes a direct comparison between techniques difficult. This is compounded by the fact that each technique works well for some images and not for others. In general, most of the techniques will perform well on head-and-shoulders-type images, as in the test image used here.

Finally, it should be noted that no direct comparison has been made between techniques that rely on different modeling schemes. In general, results from image coding are normally quoted in terms of compression ratio, or bits per pixel, of the compressed image. The compression ratio is the ratio of the original image size to the compressed image size. In the past, various metrics have been used as an indication of the error associated with the compression/decompression cycle of first-generation coding techniques. These include root-mean-squared (RMS) error and signal-to-noise ratio (SNR). The only comparison that could be made between the methods presented here is via their reported compression ratios. However, the quality of the decoded image is also an important factor for second-generation coding. As a result, direct comparisons on compression ratio are not considered valid since one particular method may produce a good compression ratio but unacceptable image quality. Therefore, until a quantitative measure of image quality has been adopted, direct comparisons between these second-generation techniques will not be possible.

REFERENCES

AVERBUCH, A., LAZAR, D., AND ISRAELI, M. 1996. Image compression using wavelet transform and multiresolution decomposition. *IEEE Trans. Image Process. 5*, 1, 4–15.

BIGGAR, M., MORRIS, O., AND CONSTANTINIDES, A. 1988. Segmented-image coding: Performance comparison with the discrete cosine transform. *IEEE Proceedings 135*, 2, 121–132.

BURT, P. AND ADELSON, E. 1983. The Laplacian Pyramid as a compact image code. *IEEE Trans. Commun. COM-31*, 4, 532–540.

CHELLAPPA, R., CHATTERJEE, S., AND BAGDAZIAN, R. 1985. Texture synthesis and compression using Gaussian-Markov random field models. *IEEE Trans. Syst. Man Cybern. SMC-15*, 298–303.

CHEN, D. AND BOVIK, A. 1990. Visual pattern image coding. *IEEE Trans. Commun. 38*, 12, 2137–2145.

CHERITON, D. AND TARJAN, R. 1976. Finding minimum spanning trees. *SIAM J. Comput. 5*, 724–742.

CHO, Z., KIM, H., SONG, H., AND CUMMING, J. 1982. Fourier transform nuclear magnetic resonance tomographic imaging. *Proc. IEEE 70*, 10, 1152–1173.

CICCONI, P. AND KUNT, M. 1993. Symmetry-based image segmentation. *Soc. Photo Optical Instrumentation Eng. (SPIE)* 1977, 378–384.

CICCONI, P. ET AL. 1994. New trends in image compression. *Comput. Med. Imaging Graph. 18*, 2, 107–124.

CIVANLAR, M., RAJALA, S., AND LEE, W. 1986. Second generation hybrid image-coding techniques. *SPIE—Visual Commun. Image Process. 707*, 132–137.

CORNSWEET, T. 1974. *Visual Perception.* Academic Press, New York.

Delp, E. and Mitchell, O. 1979. Image compression using block truncation coding. *IEEE Trans. Commun. COM-27*, 9, 1335–1342.

Fischer, Y. 1985. *Fractal Compression: Theory and Applications to Digital Images*. Springer-Verlag, New York.

Forchheimer, R. and Kronander, T. 1989. Image coding—from waveforms to animation. *IEEE Trans. Acoustics, Speech Signal Process. 37*, 12, 2008–2022.

Freeman, H. 1961. On the encoding of arbitary geometric configurations. *IRE Trans. Electron. Comput. EC-10*, 260–268.

Freeman, H. 1974. Computer processing of line drawings. *Comput. Surv. 6*, 1, 57–97.

Freeman, H. 1976. Analysis of line drawings. *NATO Advanced Study Institutes series E 20*, 187–209.

Golomb, S. 1966. Run length encodings. *IEEE Trans. Inf. Theor. IT-12*, 399–401.

Graham, D. 1967. Image transmission by two-dimensional contour coding. *IEEE Proc. 55*, 3, 320–330.

Gray, R. 1984. Vector quantization. *IEEE Trans. Acoustics, Speech, Signal Process. 1*, 4–29.

Haralick, R. 1983. Image segmentation survey. In *Fundamentals in Computer Vision*. Cambridge University Press.

Harris, F. 1978. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE 66*, 1, 51–83.

Huffman, D. 1952. A method for the construction of minimum redundancy codes. *Proc. IRE 40*, 9, 1098–1101.

Huffman, J. 1994. Wavelets and image compression. *SMPTE J. Soc. Motion Picture Television Eng. 11*, 723–727.

Ikonomopoulos, A. and Kunt, M. 1985. High compression image coding via directional filtering. *Signal Process. 8*, 179–203.

Jacquin, A. 1993. Fractal image coding: A review. *Proc. IEEE 81*, 10, 1451–1465.

Jain, A. 1989. *Fundamentals of Digital Image Processing* (first ed.). Prentice-Hall International.

Jang, J. and Rajala, S. 1990. Segmentation based image coding using fractals and the human visual system. In *Proceedings of ICASP '90*, 1957–1960.

Jang, J. and Rajala, S. 1991. Texture segmentation-based image coder incorporating properties of the human visual system. In *Proceedings of ICASSP '91*, 2753–2756.

Jayant, N., Johnston, J., and Safranek, R. 1993. Signal compression based on models of human perception. *Proc. IEEE 81*, 10, 1385–1421.

Kaneko, T. and Okudaira, M. 1985. Encoding of arbitary curves based on the chain code representation. *IEEE Trans. Commun. COM-33*, 7, 697–706.

Kean, D. and David, M. 1986. *Magnetic Resonance Imaging, Principles and Applications* (first ed.). Heinemann Medical.

Kocher, M. and Kunt, M. 1983. Image data compression by contour texture modelling. *Proc. Soc. Photo-Optical Instrumentation Eng. (SPIE) 397*, 132–139.

Kocher, M. and Leonardi, R. 1986. Adaptive region growing technique using polynomial functions for image approximation. *Signal Process. 11*, 47–60.

Kunt, M. 1988. Progress in high compression image coding. *Int. J. Pattern Recogn. Artif. Intell. 2*, 3.

Kunt, M. 1990. Object based image and image sequence coding. In *Time Varying Image Processing and Moving Object Recognition*, 19–29.

Kunt, M., Benard, M., and Leonardi, R. 1987. Recent results in high compression image coding. *IEEE Trans. Circuits Syst. CAS-34*, 1306–1336.

Kunt, M., Ikonomopoulos, A., and Kocher, M. 1985. Second generation image coding. *Proc. IEEE 73*, 4, 549–574.

Kwon, O. and Chellappa, R. 1993. Segmentation- based image compression. *Optical Eng. 32*, 7, 1581–1587.

Leou, F. and Chen, Y. 1991. A contour based image coding technique with its texture information reconstructed by polyline representation. *Signal Process. 25*, 81–89.

Limb, J. 1978. On the design of quantizer for dpcm coder: A functional relationship between visibility, probability and masking. *IEEE Trans. Commun. COM-26*, 573–578.

Mallat, S. and Zhong, S. 1991. Compact coding from edges with wavelets. In *Proceedings of ICASSP '91*, 2745–2748.

Mandelbrot, B. 1982. *The Fractal Geometry of Nature* (first ed.). Freeman, New York.

Mannos, J. and Sakrison, D. 1974. The effects of a visual fidelity criterion on the encoding of images. *IEEE Trans. Inf. Theor. IT-20*, 4, 525–536.

Marques, F., Gasull, A., Reed, T., and Kunt, M. 1991. Coding-oriented segmentation based on Gibbs-Markov random fields and human visual system knowledge. In *Proceedings of ICASSP '91*, 2749–2752.

Marshall, S. 1987. *Shape Description*. Univ. of Newcastle, Newcastle, England.

Marshall, S. 1992. Application of image contours to three aspects of image processing: Compression, shape recognition and stereopsis. *IEE Proc. I*, 139, 1.

Millar, R. and Nicholl, P. 1994. Magnetic resonance image compression by spiral coding. In

Proceedings of IEEE First International Conference on Image Processing, 356–360.

MORRIS, O., LEE, M. D. J., AND CONSTANTINIDES, A. 1986. Graph theory for image analysis: An approach based on the shortest spanning tree. *IEE Proc. F, 133*, 2, 146–152.

MORRIS, P. AND PETER, G. 1986. *Nuclear Magnetic Resonance Imaging in Medicine and Biology* (first ed.). Oxford Science Publications.

OLZAK, L. AND THOMAS, J. 1986. *Seeing Spatial Patterns*. Wiley, New York.

O'ROURKE, T. AND STEVENSON, R. 1995. Human visual system based wavelet decomposition for image compression. *J. Visual Commun. Image Representation 6*, 2, 109–121.

PAL, N. AND PAL, S. 1993. A review on image segmentation techniques. *Pattern Recogn. 26*, 9, 1277–1294.

PAVLIDIS, T. 1982. *Algorithms for Graphics and Image Processing* (first ed.). Computer Science Press, Rockville, MD.

REID, M., MILLAR, R., AND BLACK, N. 1994. A comparison of first generation image coding techniques applied to the same magnetic resonance image. *Innovation et Technologie en Biologie et Medicine 15*, 4, 512.

ROBSON, J. 1966. Spatial and temporal contrast-sensitivity functions of the visual system. *J. Optical Soc. Am. 56*, 1141–1142.

ROSENFELD, A. AND KAK, A. 1982a. *Digital Picture Processing: Vol. 1* (second ed.). Academic Press, New York.

ROSENFELD, A. AND KAK, A. 1982b. *Digital Picture Processing: Vol. 2* (second ed.). Academic Press, New York.

SAMET, H. 1989a. *Applications of Spatial Data Structures* (first ed.). Addison-Wesley, Reading, MA.

SAMET, H. 1989b. *The Design and Analysis of Spatial Data Structures* (first ed.). Addison-Wesley, Reading, MA.

SCHREIBER, W. 1963. The mathematical foundation of the synthetic highs system. *MIT, RLE Q. 68*, 140.

SCHREIBER, W., KNAPP, C., AND KAY, N. 1959. Synthetic highs, an experimental TV bandwidth reduction system. *J. SMPTE 68*, 525–537.

SHAPIRO, J. 1993. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans. Signal Process. 41*, 12, 3445–3462.

SILSBEE, P. AND BOVIK, A. 1991. Adaptive visual pattern image coding. In *Proceedings of ICASSP '91*, 2765–2768.

WAITE, J. 1990. A review of iterated function system theory for image compression. *IEE Colloquium on The Applications of Fractal Techniques in Image Processing*.

WALLACE, G. 1991. The JPEG still picture compression standard. *Commun. ACM 34*, 4, 31–44.

WANG, D. AND VAGNUCCI, A. 1981. Gradient inverse weighted smoothing scheme and the evaluation of its performance. *Comput. Graph. Image Process. 15*, 167–181.

WELCH, T. 1984. A technique for high performance data compression. *IEEE Computing 17*, 6, 8–19.

WITTENBERG, U. 1993. Applications of the JPEG standard in a medical environment. *Society of Photo-Optical Instrumentation Engineers (SPIE)* 1977, 121–127.

ZHOU, Z. AND VENETSANOPOULOS, A. 1992. Morphological methods in image coding. In *Proceedings of ICASSP '92*, 481–484.

ZIEMER, R., TRANTER, W., AND FANNIN, D. 1989. *Signals and Systems: Continuous and Discrete* (second ed.). Macmillan, New York.

ZIV, L. AND LEMPEL, A. 1977. A universal algorithm for sequential data compression. *IEEE Trans. Inf. Theory 23*, 3, 337–343.

Received June 1995; revised August 1996; final revision accepted October 1996