compression techniques is segmentation-based image compression [3,10-12]. In segmentation-based image compression, the image to be compressed is segmented, i.e. the pixels in the image are separated into mutually exclusive spatial regions based on some criteria. Properties of the HVS can also be incorporated into the criteria to obtain a reconstructed image with a small visual loss. Once the image has been segmented, information is extracted describing the boundaries (shapes) and textures (interiors) of the image segments, and compression is achieved by efficiently encoding this information. Unfortunately, there are limitations with existing segmentation-based image compression techniques. The main limitation is due to the fact that the image data have been segmented into regions of constant intensity [3,11-12]. In highly textured areas, a good representation of the texture requires many small segments. However, in order to get low bit rates, the number of segments must be limited and thus the quality is degraded.

We overcome the texture representation problem in this paper by proposing a methodology for segmenting an image into texturally homogeneous regions with respect to the degree of roughness as perceived by the HVS. The fractal dimension, the expected value, and the just noticeable difference (JND) are the measures used to characterize the texture information. The measured quantities are incorporated into a centroid-linkage region growing algorithm [7] which is used to segment each image into three texture classes: perceived constant intensity, smooth texture, and rough texture. The region growing algorithm is directed by the texture feature distance between image blocks. After segmentation, the image can be viewed as being composed of region boundaries and texturally homogeneous regions. As image coding system with high compression and good image quality is achieved by developing an efficient coding technique for the region boundaries and the three textural classes. The proposed algorithm is applied to two different types of imagery. The first is a head and shoulder image with little texture variation. This image is typical of video teleconferencing applications and one which the previously proposed segmentation-based compression techniques are best suited. The second is a natural outdoor image with highly textured areas. The existing segmentation-based compression techniques [3,10,12] do not work well for the second image. However, the proposed texture-based image compression technique works well not only for the first but also the second type of image.

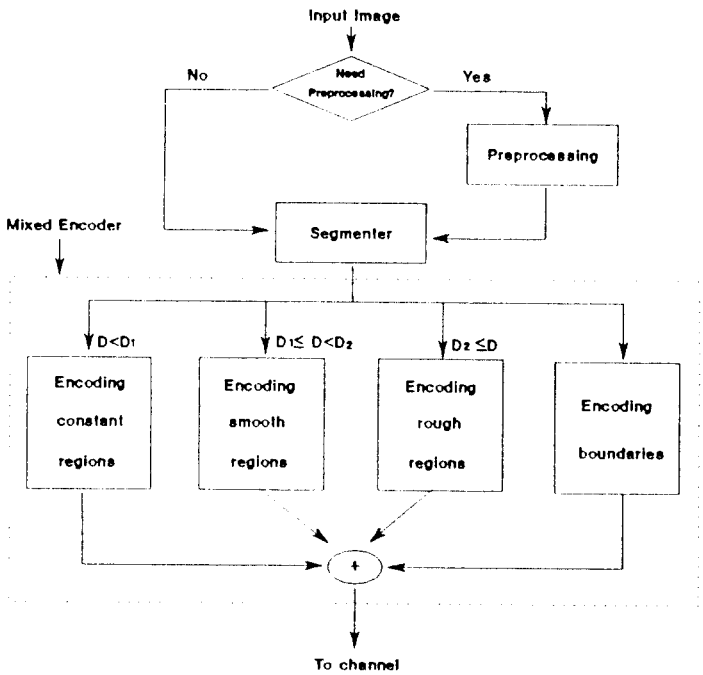
In section 2, we describe the texture segmentation-based image compression system and its major components. In section 3, the coder performanc is evaluated using computer simulated data of actual images. Finally, conclusions are provided in section 4.

2. A Texture Segmentation-Based Image Coder

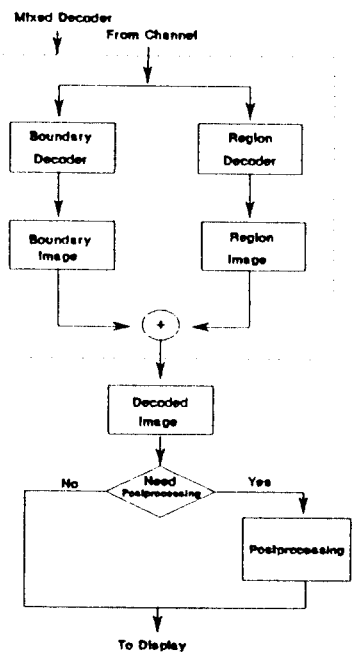
The goal of the image segmentation process is to decompose an image into texturally homogeneous regions with respect to the degree of roughness as perceived by the HVS. Textural regions are classified into three classes: perceived constant intensity, smooth texture, and rough texture. The segmentation algorithm we employ is based on a centroid linkage region growing method. It was chosen because it produces disjoint segments with closed boundaries and can be implemented with a sequential algorithm. The texture fea-

tures used are the mean, the JND, and the fractal dimension of an image block. The class type is determined by thresholding the fractal dimension. If a block has a fractal dimension less than D_1, it is assigned to class Ⅰ (perceived constant intensity). If a block has fractal dimension greater than or equal to D_1 and less than D_2, it is assigned to class Ⅱ (smooth texture). If a block has fractal dimension greater than or equal to D_2, it is assigned to class Ⅲ (rough texture). The proposed texture segmentation-based image coding system is given in figure 1. The block diagram of the transmitter for our texture-based segmentation image coder is shown in figure 1(a). It includes three main stages; the preprocessor, the segmenter, and the mixed encoder. The block diagram of the receiver is shown in figure 1(b). It includes two main stages; the mixed decoder and the postprocessing.

To better appreciate the proposed image compression technique, the relevant properties of the HVS are first covered in subsection 2.1. In subsection 2.2, fractal models in texture analysis are described, including the use of the fractal dimension. In subsection 2.3 and 2.4, an efficient method to estimate visual threshold for regions belonging to class Ⅰ and a texture-based segmentation image



(a) The block diagram of the transmitter.



(b) The block diagram of the receiver.

Figure 1. The proposed texture segmentation-based image coding system.

codec are described respectively. A complete description of the proposed texture-based segmentation algorithm is then given in subsection 2.5.

2.1. The Human Visual System (HVS)

One property of the HVS we use is contrast sensitivity. Contrast sensitivity is a measure of the HVS's ability to distinguish two adjacent intensity patches. To obtain the contrast sensitivity, a subject is shown a test pattern, for example, two adjacent squares. The luminance of one square is varied until it is just noticeably different in luminance than the adjacent square. The JND between the two luminance values is used to calculate the contrast, see [5] for details. It has been shown that the HVS has greatly reduced contrast sensitivity in very bright or very dark intensity regions of an image. Here the contrast sensitivity is used in defining the threshold for the split-merge condition for regions belonging to the perceived constant intensity in the proposed texture segmentation-based compression technique.

A second property of the HVS is the modulation transfer function (MTF), a measure of the spatial sensitivity of the HVS. The MTF is obtained by showing an observer two sine wave grating transparencies, a reference grating of constant and spatial frequency, and a variable-contrast test grating whose spatial frequency is set at some value different from that of the reference. The contrast of the test grating is varied until the brightness of the bright and dark regions of the two transparencies appear identical. The shape of the MTF curve is similar to a bandpass filter and suggests that the HVS is more sensitive to middle spatial frequencies and less sensitive to low and high spatial frequencies. This property is used in conjunction with fractal dimension to determine to which class each texture region is assigned.

2.2. Texture Analysis

Among the various characteristic of an image, texture has been recognized as one of the most important. It is important because it provides the essential structure information in an image and allows us to group pixels into relatively large, homogeneous regions. For example, grass, sky, and tree each define relatively large homogeneous regions with their own textural structure. When a relatively large region can be represented by a single texture, a significant amount of redundancy can be removed. A good representation of the texture information in an image is necessary for developing a system with high compression and good quality.

Fractal Geometry in Image Analysis

If we regard the pixel intensity in an image as the height above a plane, then the intensity surface of a texture image can be viewed as a rugged surface. The fractal model provides an excellent description of the ruggedness of the natural surface. The fractal model has been used in the computer graphic simulation of natural phenomena like mountains, trees, human faces, and animals [1]. The fractal model has been applied to texture image analysis [2,4,6,8,10,19]. One important characteristic of fractal is the fractal dimension D, which is related to the metric properties, length and surface of a curve. D provides a good measure of perceptual roughness of the curve or surface, with increasing values in D representing perceptually rougher curves and surfaces [19]. The most useful fractal model has been the fractional

Brownian function (FBF) introduced by Mandelbrot and Van Ness [13-14,17] since it produces surfaces that closely resemble natural surfaces. The FBF model regards naturally occurring rough surfaces as the end result of random walks. Such random walks are basic physical processes in our universe. An intensity surface of a texture image can also be viewed as the end result of a random walk, so the FBF can be used for the analysis of image texture.

Fractal Dimension

The definition of the fractal dimension is a set for which the Hausdorff-Besicovich dimension is strictly greater than the topological dimension. We consider object X in an E-dimensional space. $N(\varepsilon)$ is the number of E-dimensional sphere of diameter ε needed to cover X, where E is an integer and the E-dimensional space is the minimum integer dimensional space among all possible integer dimensional spaces which can envelop X. Thus, if $N(\varepsilon)$ is given by

$$N(\varepsilon) = K\cdot(\frac{1}{\varepsilon})^{D}, \qquad as\ \varepsilon \to 0, \tag{1}$$

where K is a constant and X has Hausdorff dimension D. If D is fractional, D is also called the fractal dimension. For fractal objects, D is independent of ε.

If the fractal dimension is to be used to characterize the texture in an image, we need a method for estimating the fractal dimension from the given dataset. Many different estimators have been proposed: box counting [1], yardstick [6], blanket [18], and power spectrum [19]. In our case, a blanket method is adopted since it is computationally efficient. The blanket method is described in detail in paper [18]. A brief explanation of the procedure for estimating the fractal dimension is given here. All points in the three-dimensional space at distance ε from the surface are considered, covering the surface with a blanket of thickness 2ε. The covering blanket is defined by its upper surface u_ε and its lower surface b_ε. Initially, given the gray level function $g(i,j)$, $u_0(i,j)=b_0(i,j)=g(i,j)$. For ε=1, 2, 3,..., the blanket surfaces are defined as follows:

$$u_\varepsilon(i,j)=\max\{u_{\varepsilon-1}(i,j)+1,\ \max_{|(m,n)-(i,j)|\le 1} u_{\varepsilon-1}(m,n)\} \tag{2}$$

and

$$b_\varepsilon(i,j)=\min\{b_{\varepsilon-1}(i,j)+1,\ \min_{|(m,n)-(i,j)|\le 1} b_{\varepsilon-1}(m,n)\} \tag{3}$$

The image points (m,n) with distance less than one from (i,j) were taken to be the four immediate neighbors of (i,j). Similar expresseions exist when the eight-neighborhood is desired. A point (x,y,f) will be included in the blanket for ε when $b_\varepsilon(x,y)<f\le u_\varepsilon(x,y)$. The blanket definition uses the fact that the blanket of the surface for radius ε includes all the points of the blanket for radius ε-1, together with all the points within radius 1 from the surfaces of the blanket. Expression (2), for example, ensures that the new upper surface u_ε is higher by at least 1 from $u_{\varepsilon-1}$, and also at distance at least 1 from $u_{\varepsilon-1}$ in the horizontal and vertical directions.

The volume of the blanket is computed from u_ε and b_ε by

$$v_\varepsilon=\sum_{i,j}(u_\varepsilon(i,j)-b_\varepsilon(i,j)) \tag{4}$$

As the surface area measured with radius ε we take the volume of the added layer from radius ε-1, divided by 2 to account for both upper and lower layers:

$$A(\varepsilon)=\frac{(v_{\varepsilon}-v_{\varepsilon-1})}{2}=\varepsilon^{2}\cdot N(\varepsilon)=K\cdot\varepsilon^{2-D} \quad (5)$$

where K is a constant.

From a theoretical viewpoint, if a surface is a perfect fractal surface, then the fractal dimension will remain constant over all ranges of scale ε. In practice, there are scale range limitations of fractal dimensions due to limitations in textural images. For example, the resolution limit of the image system sets a lower limit on the fractal scaling behavior. An upper limit may be set by the structure being examined. Thus, a real surface will be fractal over some range of scales rather than over all scales. These limiting scales can be expressed as upper (ε_{max}) and lower cutoff (ε_{min}) scales.

To compute the fractal dimension, we apply the log function to both sides of Eq. (2). A least square linear regression is applied to fit a straight line to the plot of $\log A(\varepsilon)$ vs. $\log(\varepsilon)$ for ε_{min} and ε_{max}. The fractal dimension D is equal to 2 minus the slope of the straight line. To illustrate the point, we show the results of applying this algorithm to a natural image with 32×32 pixels and 256 gray levels in figure 2(a). Figure 2(b) is a plot of measured surface area, $A(\varepsilon)$ versus ε in log-log scale respectively for ε=1, , 7. The fit is good for ε=1,…, 5, which implies that the natural image is a fractal surface for ε=1,…, 5. Therefore, ε_{min} and ε_{max} are 1 and 5 respectively. The value of the estimated slope is -0.62729. Therefore, the estimated fractal dimension is 2.62729.

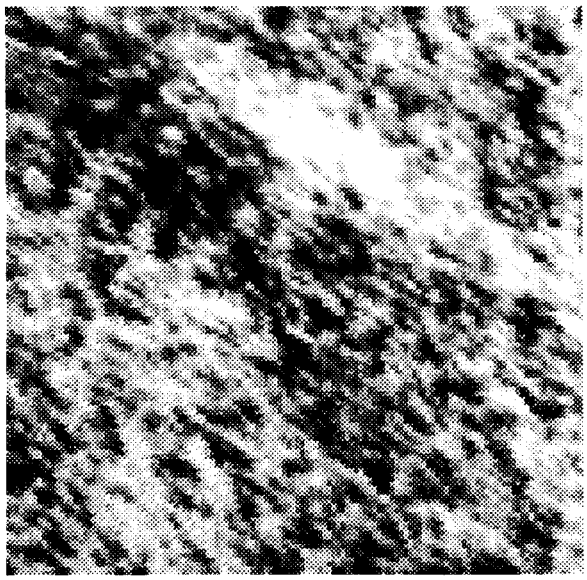
It proved that the higher the value of D, the higher the spatial frequency content, the rougher the waveform from the papers dicussed in [4, 17]. In the following subsection we incorporate this relationship into the proposed codec.

2.3. Visual Threshold for Regions Belonging to Class I

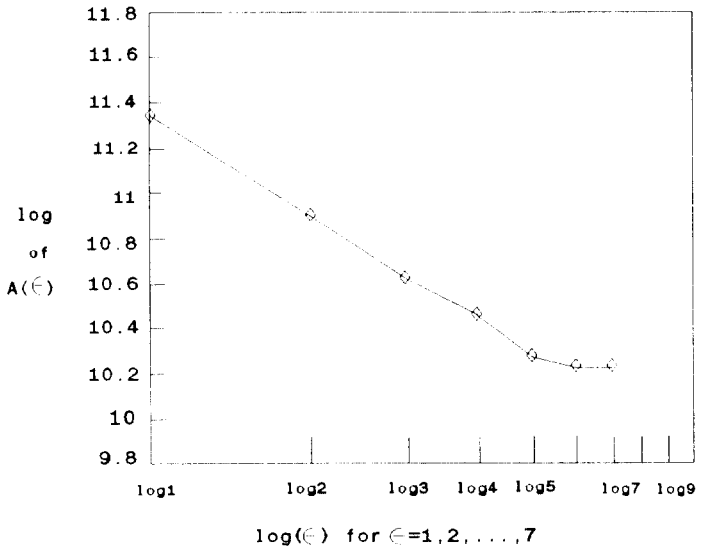
To determine the split-merge condition between the regions belonging to class I, we use a visual threshold based on the HVS properties in the segmentation algorithm. The HVS-based threshold is adapted to local intensity characteristics in the image by using the JND as the visual threshold.

A split-field method is used to measure the JND on SUN SPARC workstation with 1024 × 768 19″ color monitor. The image display device is divided down the middle into two equal-size fields. The left field is a constant reference intensity and increases linearly up to 40 steps above the constant reference intensity. To perform the tests, the viewer simply clicks the mouse at the point where the difference between the left and right fields is no longer discernible. This point is the JND between the reference intensity on the left and the test intensity on the right.

Five test subjects were asked to take a measurement of JND. Each test subject sat at a distance of six times the image height away from the screen. The test subject was given approximately three minutes before the start of the experiments, to allow for adaptation to the laboratory's illumination. The test subject was asked to take five seconds in each click to allow for adaptation to the screen. The average of the results of five test subjects is shown in figure 3. It is seen that the experimental results agrees with the HVS contrast sensitivity properties [5] described in subsection 2.1. The JND is largest in the lowest and highest intensity areas of the image. The JND is smallest and nearly constant in the middle intensity areas of the image. To



(a)



(b)

Figure 2. (a) A natural image with 32 ×32 pixels and 256 gray levels. (b) The calculation of the fractal dimension of a natural image given in Figure 2(a). The values of the estimated slope and estimated fractal dimension are -0.62729 and 2.62729 respectively.

determine threshold, the approximated JND curve is derived in figure 3. The bold line corresponds to the approximated JND curve. Based on this result, a larger threshold is chosen for the lowest (0 to 74) and highest intensity areas (241 to 255), while a smaller threshold (about 5) is used in the middle intensity area (75 to 240).

2.4. A Texture-Based Segmentation Image Codec

In a segmentation-based compression algorithm, the number of segments and number of bits representing the textures of the segments are directly proportional to the bit rate of the coded image. Thus, the main purpose of the preprocessor which is the first stage of the proposed transmitter in figure 1(a) is to alter the image in such a way that fewer segments and textures are produced by the segmenter, but without degrading the visual quality of the segmented image. It should be noted, however, that it is possible that no preprocessing will be required in this paper.

After preprocessing, the image data is segmented into texturally homogeneous regions with respect to the degree of roughness as perceived by the HVS. The segmentation is accomplished by thresholding the fractal dimension so that membership is in one of three textural classes: perceived constant intensity, smooth texture, and rough texture.

The last stage in the transmitter is the mixed encoding to the segments of each class and their boundaries. The objective of the coding is to obtain an efficient representation of the segmented image data for transmission or storage. The image coder should use more

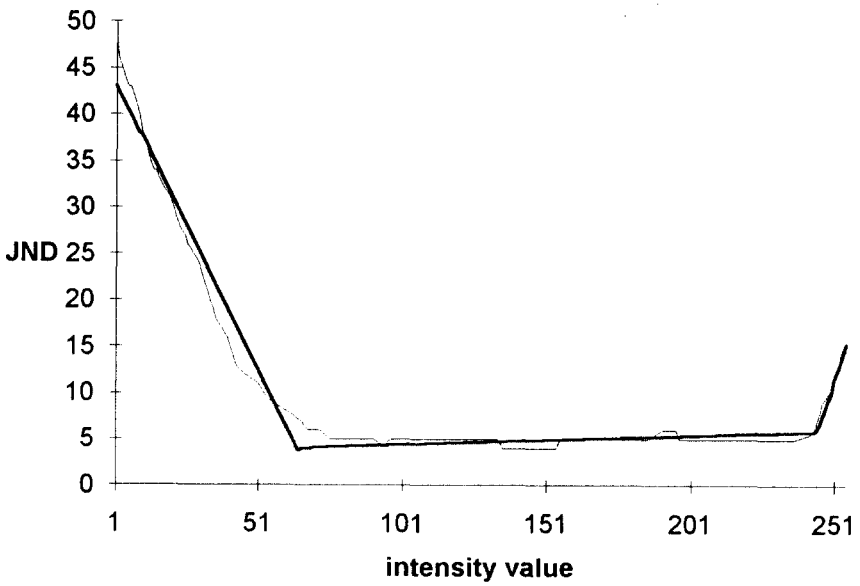


Figure 3. The mean of five subjects's JND measurements on SUN SPARC workstation with 1024 × 768 19″ color monitor. The bold line corresponds to the approximated JND curve.

bits to encode the information for which the HVS is more sensitive and use fewer bits to encode the information which the HVS is less sensitive. To accomplish this we propose a mixed encoder. It consists of four alternative stages: the boundary encoding and three textural class encodings, see figure 1(a).

For boundary coding, accurate representation of the boundary is necessary to describe the location of the region boundary because of the HVS sensitivity of the edges. As a result, we choose an errorless coding scheme to represent the boundaries. A binary image representing the boundaries is created. Then, the binary data is encoded using an arithmetic code [20].

In our approach, the boundary information was represented using blocks, not pixels. Therefore, the number of bits to represent the boundaries is reduced by almost the block size. The compression ratio of the boundaries does not govern the overall compression ratio.

For regions which belong to class Ⅰ (perceived constant intensity), only the mean intensity values need be transmitted to describe the textures of the regions. In this case, lossy compression has already taken place since we are approximating each region texture with a constant value. We do not wish to introduce any further compression so a lossless arithmetic code is again employed to achieve further compression. Since an intensity mean requires 8 bits, the mean intensity value is converted into a vector of an $8\times N$ binary array, where N is the number of segments belonging to perceived constant regions. The mean vector is then encoded using an arithmetic code.

Regions belonging to class Ⅱ and class Ⅲ are not directly encoded. To get higher compression, their regions are modeled first using polynomial functions. The coefficients of the polynomial functions are encoded because the variance of the coefficients is less than that of the original data. An arithmetic code is used to encode the coefficients. A smaller amount of error between the original image data and the modeled image data is chosen for class Ⅱ than class Ⅲ because of the sensitivity of the HVS. In general, modeling these regions by functions of higher order polynomials is computationally excessive. The first order polynomial functions are used because the sum squared error (SSE) for the first order polynomial functions as given in Eq. (7) is not much greater than that of the SSE for a second-order polynomial function, while the SSE for the first order polynomial function is much greater than the SSE for the zero order polynomial function[3].

$$\sum_i \sum_j (g(i,j)-z(i,j))^2 \tag{6}$$

where $g(i,j)$ and $z(i,j)$ are the intensity of the original image and the modeled image at index (i,j) respectively. The experimental results of the mixed encoder will be given in subsection 3.1.

At the receiver, see figure 1(b), two types of coded information come into the mixed decoder, boundary information and region texture information. The boundary decoder must regenerate the boundaries for the decoded image. The region decoder must fill in the missing texture information within each region. Regions belonging to class Ⅰ are perceived constant regions, thus their mean values are painted within the appropriate regions. The texture informations for classes Ⅱ and Ⅲ are reconstructed by reproducing the appropriate polynomial functions.

2.5. Texture-Based Segmentation Algorithm

Incorporating the HVS and the fractal model, the proposed texture-based image segmentation algorithm for image coder is defined as follows.

Step 1) Divide the image into $NR \times NC$ blocks (NR and NC are the numbers of row and column blocks, respectively).

Step 2) Calculate the feature set: the mean and the class type for each block and the JND lookup table.

Step 3) Calculate the distance between an observing block and its 4-connected neighboring blocks. The distance is given by

$$D(OB,NB)=\begin{cases} 0 \text{ if } \begin{cases} F(OB)<D_1,\ C(OB)=C(NB) \\ |M(OB)-M(NB)|<JNB(OB,NB) \\ \text{or} \\ D_1 \le F(OB)<D_2,\ C(OB)=C(NB) \\ \text{or} \\ F(OB)\ge D_2,\ C(OB)=C(NB) \end{cases} \\ 1 \text{ otherwise} \end{cases}$$

where $F(OB)$ is the fractal dimension of an observing block. $C(OB)$ and $C(NB)$ are the class types of an observing block and its neighboring block respectively. $M(OB)$and $M(NB)$ are the means for an observing block and its neighboring block respectively. $JND(OB,NB)$ is the just noticeable difference between an observing block and its neighboring block.

Step 4) If there is a neighboring block with distance 0, then merge the observing block into it; else declare a new region. If there are more than two good neighboring blocks, merge the observing block into a neighboring block whose mean value is closest to the mean value of the observing block.

Step 5) Repeat step 3 to step 4 until all blocks are segmented and stop.

3. Coder Performance

In this section, the performance of the proposed image coding technique is presented. First, we investigate the number of segments for each class and the boundary points using D_1=2.033 and D_2=2.371.

3.1. The Experimental Results of the Mixed Encoder

Segmented images were obtained with D_1=2.033 and D_2=2.371. Information about the number of total segments and the boundary points and the number of the bits to represent the boundaries is summarized in table 1 using an arithmetic code. Number of segments in Miss USA is almost equal to that of segments in House and since Miss USA has a very large background and House has large sky in the top, trees and bushes in the middle, and wall in the bottom. The numbers of boundary points of Miss USA and House are 1910 and 1945 respectively. After encoding, numbers of bits required to represent the boundary are 1344 and 1358 respectively which are fairly small. Recall, the boundary information is represented using blocks, not pixels. Therefore, the number of bits to represent the boundaries is reduced by almost the block size (8×8).

To encode the regions belonging to class I, their mean values are converted into an $8 \times N$ binary. Each mean value is represented by 8 bits and there are N segments belonging to class I. The total number of the regions and the number of constant regions and the number of bits to represent the constant regions are given in table 2 using an arithmetic code. After encoding, the number of bits required to represent the regions to class I are 1022 and 1156 respectively.

To encode the texture information of the regions belonging to class Ⅱ and class Ⅲ, the first order 1-D polynomial functions were used to represent textural regions. Error amounts tolerated between the original image data and the modeled image data were 40 for class Ⅱ and 80 for class Ⅲ. The polynomial coefficients were encoded using an arithmetic code. The result is given in table 3. The numbers of regions in Miss USA and House are 56 and 62 respectively. Based on these results, House has a small number of regions belonging to class Ⅱ and class Ⅲ although House looks the most complicated. That is why trees, lawns, and bushes in House are segmented into few large regions instead of many small regions. After encoding, the numbers of bits required to represent those regions are 6244 and 10315 respectively.

Table 1. Summary of the numbers of the total segments and the boundary points using D_1=2.033 and D_2=2.371

image	number of total segments	number of boundary points	number of bits for arithmetic code
Miss USA	231	1910	1344
House	245	1945	1358

Table 2. Summary of the numbers of the total segments and the regions belonging to class Ⅰ using D_1=2.033 and D_2=2.371.

image	number of constant regions	number of bits for arithmetic code
Miss USA	175	1022
House	183	1156

Table 3. Summary of the numbers of the segments belonging to class Ⅱ and class Ⅲ and the number of bits to represent those regions using a 1-D polynomial. The polynomial coefficients were encoded using the arithmetic code.

image	number of segment regions for class Ⅱ and Ⅲ	number of bits for arithmetic code
Miss USA	56	6244
House	62	10315

Figure 4. The two test images. Each image is 256×256 pixels with 256 gray levels. Miss USA and House are on the left and right respectively.

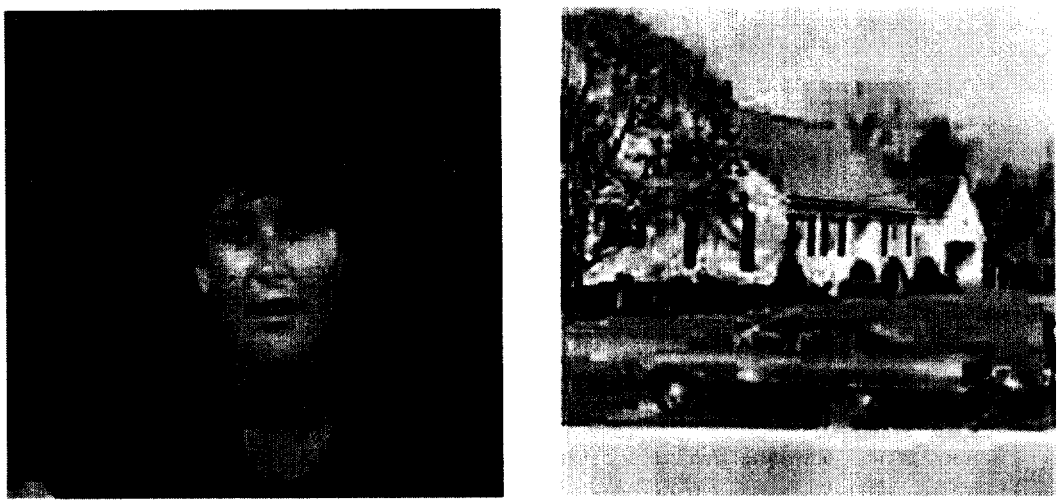


Figure 5. The decoded images of two test images with D_1=2.033 and D_2=2.371. The decoded images of Miss USA and House are on the left and right respectively. The compression ratios for the two test images are 0.13 and 0.19 respectively.

3.2. Bit Rate Computation

The number bits calculated for the boundary, constant region, and smooth/rough texture as given in table 1,2, and 3 are added. We see that Miss USA requires 1344+1022+6244=8610 bits for encoding and House requires 1385+1156+10315=12829 bits for this coding method. The compression ratios, BR for the test images are 8610/65536=0.13 and 12829/65536=0.19 bit per pixel, respectively.

The two test images and the decoded images for each test image are given in figure 4 and figure 5 respectively. The image quality of the decoded images are good with these bit rates. Through our experiment, our segmentation-based image compression method works well for a wide variety of images including a natural image with highly textural areas referred as House.

4. Conclusion

These results indicate that, using the proposed texture segmentation-based image compression system combining fractals and properties of the HVS, compression ratios in the neighborhood of 0.13 to 0.19 bpp are attainable with good image quality. The segmentation is good and conforms to the human perception of roughness. These compression ratios are almost the same as those achieved by a segmentation-based compression method using flat segments [3,12]. However, proposed compression technique produces better image quality and is quite useful for a wider variety of images. This is because our segmentation compression method was developed using texture features as well as gray levels. Specifically, our technique works well for images with highly textured areas, while previous compression techniques were not useful for those images.

References

1. M. Barnsley, Fractal everywhere, Academic Press, Inc. 1988.
2. M. Barnsley and A. D. Sloan, A better way to compress image, Byte, pp.215-223, 1988.
3. M. J. Biggar, O. J. Morris, and A. G. Constantinides, "Segmented-image coding: performance comparison with the discrete cosine transform," Proc. IEE, Part F, 135(2): 121-132, April, 1988.
4. C. Chen, J. S. daponte, and M. D. Fox, "Fractal feature analysis and classification in medical imaging," IEEE Trans. Medical Imaging, 8(2):133-142, June. 1989.
5. T. N. Cornsweet, Visual Perception, New York : Academic Press, 1971.
6. T. J. Dennis and N. G. Dessipris, "Fractal modeling in image texture analysis," IEE Proceedings, F(5):227-235, Oct. 1989.
7. R. M. Haralick and L. G. Shaprio, "Image segmentation techniques," Comput. Graphics Image Processing, 29:100-132, 1985.
8. A. E. Jacquin, "A novel fractal block-coding technique for digital images," In Proc. ICASSP'90, Vol. 4, pp. 2225-2228, April 1990.
9. A. K. Jain, Fundamentals of Digital Image Processing, Prentice Hall. 1989.
10. J. W. Jang and S. A. Rajala, "Texture segmentation-based image coder incorporoating properties of the human visual system," In Proc. ICASSP' 91, May. 1991.
11. J. W. Jang and S. A. Rajala, "Performance comparison of texture segmentation-based image coding technique with DCT image coding technique," IEEE Workshop on Visual Signal Processing and Communications, pp.200-205, Sep. 1992.

12. M. Kunt, M. Benard, and R. Leonardi, "Recent results in high compression image coding," IEEE Trans. Circuit Syst., Vol. CAS-34, No. 11, pp.1306-1336, Nov. 1987.
13. B. B. Mandelbrot, The Fractal Geometry of Nature, San Francisco, CA: Freeman, 1982.
14. B. B. Mandelbrot and B. J. Van Vess, "Fractional brownian motion, fractional noises and applications," SIAM, 10(4):422-438, 1968.
15. H. G. Musmann, P. Pirsch, and H. J. Grallert, "Advances in picture coding," Proc.IEEE, 73:523-548, April. 1985.
16. N. B. Nill, "A visual model weighted cosine transform for image compression and quality assessment," IEEE Trans. Commun. 33(6):551-557, 1985.
17. Heinz-Otto Peitgen and Dietmar Saupe Ed., The Science of Fractal Images. Springer-Verlag, 1988.
18. S. Peleg, J. Naor, R. Hartley, and D. Avnir, "Multiple resolution texture analysis and classification," IEEE Trans. Pattern Anal. Machine Intell., PAMI-6(4):518-523, July. 1984.
19. A. P. Pentland, "Fractal-based description of natural scenes," IEEE Trans. Pattern Anal. Machine Intell., PAMI-6(6):664-674, Nov. 1984.
20. J. Rissanen, "Generalized Kraft-inequality and arithmetic coding," IBM J. Res Devel., 20:198-203, 1976.

張 鍾 煥(Jong Whan Jang) 정회원

1979년 2월 : 한양대학교 전자통신공학과 졸업(공학사)
1986년 5월: 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학석사)
1990년 12월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학박사)
1990년 5월~현재 : 배재대학교 정보통신공학과 조교수
1992년 3월~현재 : 배재대학교 전자계산소장
※주관심 분야 : 영상통신 및 영상처리, 마이크로프로세서응용

鄭 在 吉(Gae Gil Jeong) 정회원

1980년 2월 : 한양대학교 전자공학과 졸업(공학사)
1987년 5월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학석사)
1991년 8월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학박사)
1979년 12월~1985년 6월: 국방과학연구소 연구원
1991년 8월~1992년 8월: 한국전자통신연구소 선임연구원
1992년 9월~현재: 배재대학교 전자공학과 조교수
※주관심 분야 : 병렬처리구조, 영상통신 및 영상처리, 마이크로프로세서응용

朴 斗 映(Doo Yeoun Park) 정회원

1981년 2월 : 한양대학교 전자공학과 졸업(공학사)
1987년 5월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학석사)
1993년 8월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학박사)
1993년~1994년 : 한국전자통신연구소 선임연구원
1994년 3월~현재 : 배재대학교 정보통신공학과 조교수
※주관심 분야 : 초고속대역통신망, ATM스위치성능평가, 컴퓨터통신

梁 雨 錫(Woo Suk Yang) 정회원

1979년 2월 : 서울대학교 전기공학과 졸업
1986년 6월 : 미국 University of Toledo 전기공학과 졸업(공학석사)
1991년 5월 : 미국 North Carolina State University 전기 및 컴퓨터공학과 졸업(공학박사)
1978년 11월~1984년 8월 : (주)대우
1990년 12월~1991년 8월 : 금성사 책임연구원
1991년 8월~현재 : 홍익대학교 전기공학과 조교수
※주관심 분야 : 로보틱스, 컴퓨터비젼, FA.