Segmentation-based Image Compression With Enhanced Treatment of Textured Regions

Iftekhar Hussain and Todd R. Reed

Department of Electrical & Computer Engineering and
the Center for Image Processing and Integrated Computing
University of California, Davis, CA 95616

Abstract

This paper presents a method to preprocess an image so that when segmented it yields a partitioning in which textured regions are approximated with a substantially reduced number of uniform regions (which is desirable for the coding). The segmentation method used to form this representation combines a Gaussian texture model and Gibbs-Markov contour model in order to find regions with boundaries which correspond closely to the objects in the image. Given the image segmentation, an approximation to the original image is generated by filling each region with its mean value. If higher quality reconstruction is desired, the quantized approximation error is also encoded. In order to exploit the reduced sensitivity of the human visual system to the error around edges (visual masking), the error is quantized using three nonlinear quantizers corresponding to the smoothly varying, textured, and remaining areas of the image, respectively.

1 Introduction

Segmentation-based compression approaches typically consist of representing images in terms of region and contour descriptions, each of which is then encoded [1]. If higher quality reconstruction is required, the quantized approximation error may also be encoded (see Figure 1). It is clear that the compression ratio resulting when only the region and contour descriptions are encoded (as typical in high compression applications) is proportional to the number of regions and the contours. Ideally, we would like to approximate the image (at reasonable reconstruction quality) with the least number of regions.

An undesirable feature of a segmentation-based compression approach (in which image is approximated by uniform sub-regions) is the inefficient coding of textured regions. This is because textured regions are generally approximated with a large number of uniform sub-regions. This paper presents a method to preprocess an image so that when segmented it results in a partitioning in which textured regions are approximated with a reduced number of regions, while retaining segmentation accuracy in the balance of the image. The residual (or approximation error) entropy trade-off resulting from the coarse approximation of the textured regions and the application of nonlinear quantizers to quantize the residual are described in section 3. The details of the preprocessing are given in the next section.

2 Preprocessing of the Image to be Compressed

The preprocessing consists of two steps: extraction of the textured (and smooth) regions and nonlinear smoothing of the extracted regions. Each of these steps is described in detail in the following subsections.

2.1 Extraction of the Textured and Smooth Regions

The image to be compressed is partitioned into 8×8 square windows and each window is processed independently to determine if it belongs to a textured region or a smooth region. The underlying assumption is that the grey level population of a window belonging to regions of these types is uniform. In contrast, the grey level population is bimodal if an edge falls in that window. Hence, windows belonging to textured or smoothly varying regions can be extracted based on bimodality detection.

Let P_w denote the grey level population of the wth window. If P_w is bimodal then it can be divided into two subpopulations $P_{w_l(i)}$ and $P_{w_g}(i)$. Where $P_{w_l(i)}$ consists of all pixels of the wth window having intensity less than or equal to some i, and $P_{w_g(i)}$ comprises of all pixels of the wth window which have intensity greater than i. We want to find the grey level i_{opt} such

1058-6393/95 $4.00 © 1995 IEEE

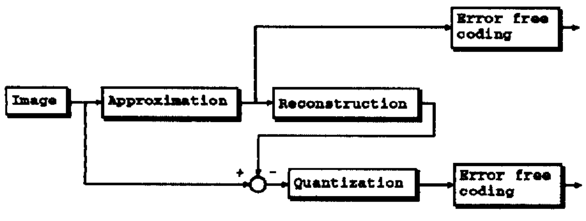

Figure 1: Block diagram of the coder

that the function J_w (bimodality measure) is minimized:

$$J_w(i) = \frac{N_{w_g(i)}\sigma^2_{w_g(i)} + N_{w_l(i)}\sigma^2_{w_l(i)}}{N\sigma^2_w}, \quad i_{min} \leq i \leq i_{max}$$

where N and σ^2_w are size and variance of the population P_w. Similarly N_{w_g}, N_{w_l}, $\sigma^2_{w_g}$, $\sigma^2_{w_l}$ are the size and variance of the subpopulations P_{w_g} and P_{w_l}, respectively. This bimodality measure is a slightly modified version of the Fisher criterion [3]. The value of J_w is a measure of bimodality because if the population is bimodal (i.e., there are two well clustered populations), $N_{w_g(i)}\sigma^2_{w_g(i)} + N_{w_l(i)}\sigma^2_{w_l(i)}$ will be small, hence making J_w small.

To determine the grey level i_{opt}, the *wth* window grey levels are sorted and $J_w(i_{opt})$ is assigned a large number (e.g., the largest floating point number that can be represented on a finite precision computer). Then, for each i from i_{min} to i_{max}, P_w is partitioned into two populations $P_{wl}(i)$ and $P_{wg}(i)$ and the corresponding $J_w(i)$ is computed. If $J_w(i) < J_w(i_{opt})$ then $J_w(i_{opt})$ is assigned the value of $J_w(i)$. The value of J_w for a uniform population is found to be 0.25; suggesting that a threshold $<$ 0.25 can be used to discriminate between uniform and bimodal populations [4]. A window is marked as bimodal if the value of J_w is less than a threshold. In this work, a threshold of 0.25 was used. Based on bimodality detection, a window which is marked as unimodal is considered a part of a textured region or a smoothly varying intensity region. Extraction of smoothly varying intensity windows in addition to the textured windows does not pose a problem because segmentation of the smoothly varying regions, in general, is not affected by the preprocessing.

Bongiovanni *et.al.* [4] have used a similar bimodality measure to perform hierarchical segmentation of a bimodal image. Their underlying assumption is that the entire image is bimodal, which generally is not the case in real images. Our goal is different, in that we are interested in bimodality detection over 8×8 windows, not the image as a whole.

2.2 Smoothing of the Extracted Regions

In order to satisfy our goal to approximate textured regions with a reduced number of regions, the extracted regions (which include textured as well as smoothly varying regions) are iteratively smoothed using 3×3 averaging. The purpose of this selective smoothing is to smear the weak textural boundaries, thus enabling approximation of the textured regions by a reduced number of uniform subregions. If smoothing is performed blindly in the extracted regions we could loose some of the visually important structure (such as fine structure nontextural edges). To reduce this effect, the magnitude of the gradient is used as a guiding mechanism in smoothing. A pixel in the extracted region is modified only if the magnitude of the gradient at that pixel is less than a threshold.

The magnitude of the gradient of an image $I(x,y)$ at a location (x,y) is given by:

$$|\Delta I| = \sqrt{G_x^2 + G_y^2}$$

where $G_x = \frac{\partial I(x,y)}{\partial x}$ and $G_y = \frac{\partial I(x,y)}{\partial y}$. Given $|\Delta I|$ for each pixel in the extracted region, a threshold is computed based on the statistics of gradient magnitude belonging to the extracted region (collection of all the extracted windows). Thus averaging in the extracted region is carried out according to the following decision function:

$$D(x,y) = \begin{cases} \textit{smooth} & \text{if } |\Delta I(x,y)| < T \\ \textit{do not} & \text{otherwise} \end{cases}$$

where the threshold $T = \hat{\mu} + k\hat{\sigma}$. If we denote the number of pixels in the extracted region R by N then $\hat{\mu}$ and $\hat{\sigma}^2$ are given by:

$$\hat{\mu} = \frac{1}{N} \sum_{(x,y)\in R} |\Delta I(x,y)|$$

$$\hat{\sigma}^2 = \frac{1}{N} \sum_{(x,y)\in R} (|\Delta I(x,y)| - \hat{\mu})^2.$$

Large values of k correspond to enhanced smoothing while small values to reduced smoothing of the fine structure non-textural edges. Depending upon the specific applications k can be chosen in the range $0 \leq k \leq 3$. In the results that follow, a value of 1.50 was used.

3 Segmentation using Gibbs-Markov random fields

This section describes a previously developed [2] image segmentation method in which a GMRF model is used to describe the region boundaries and a stationary Gaussian model to describe the grey level information or texture inside the regions. The initial segmentation (found via simple region growing) is refined by applying contour relaxation iteratively to all boundary pixels. For each boundary pixel, this relaxation consists of changing the label of the pixel, if necessary, to the label of the neighbor that locally maximizes the likelihood ratio:

$$r(Y, Q_{new}, Q_{old}, s) = e^{-(n'_a - n_a)\alpha - (n'_b - n_b)\beta} \cdot \left(\frac{\left[\hat{\sigma}^2_{new}(R_{jk})\right]^{-\frac{(N_{jk}-1)}{2}} \cdot \left[\hat{\sigma}^2_{new}(R_{jk'})\right]^{-\frac{(N_{jk'}+1)}{2}}}{\left[\hat{\sigma}^2_{old}(R_{jk})\right]^{-\frac{(N_{jk})}{2}} \left[\hat{\sigma}^2_{old}(R_{jk'})\right]^{-\frac{(N_{jk'})}{2}}} \right) \cdot e^{-w\left(\frac{|\Delta s_{jk'}|}{s_{jo'} + |\Delta s_{jk'}|}\right)}$$

where Y is the original image data, Q_{old} is the current image segmentation, and Q_{new} is the hypothetical segmentation with the pixel label switched. For region number j and iteration k, R_{jk} is the region of the partition whose boundary pixel is being examined for relaxation, N_{jk} is the size of the region, and $\hat{\sigma}^2_{old}(R_{jk})$ and $\hat{\sigma}^2_{new}(R_{jk})$ are estimated variances of R_{jk} before and after relaxation at the k^{th} iteration (i.e, before and after the label of R_{jk} boundary pixel is switched with the label of a neighboring region). $R_{jk'}$ denotes a neighbor of the region R_{jk} with size $N_{jk'}$ and variances $\hat{\sigma}^2_{old}(R_{jk'})$ and $\hat{\sigma}^2_{new}(R_{jk'})$ (before and after relaxation at the k^{th} iteration). The clique potential parameters for the inhomogeneous (i.e., having different labels) nearest and diagonal cliques are denoted by α and β, respectively. The number of inhomogeneous nearest and diagonal pairs that exist in the 3×3 neighborhood of the R_{jk}'s boundary pixel being examined, are denoted by n_a, n_b (before) and n'_a, n'_b (after the label is hypothetically switched), respectively. $s_{jo'}$ is the size of $R_{jk'}$ at the $zero^{th}$ iteration, $\Delta s_{jk'} = s_{jo'} - s_{jk'}$ is the change in the $R_{jk'}$ region size and w is a weighting factor.

3.1 Application to Image Compression

Given the segmentation of an image to be compressed, an approximation to the original image is reconstructed by approximating (in the least square sense) the smoothly varying regions by linear polynomial intensity functions. The remaining regions in the partitioned image are approximated with the corresponding mean values (constant intensity functions). If higher quality reconstruction is required, the error between the original image and its approximation is also encoded. Figure 2 shows image approximations for the 'cameraman' (which contains textured regions such as grass), and 'sand texture' (which is purely textured) images for the case when no preprocessing is performed before segmentation. The original images are shown in the top, the contour maps in the middle, and the image approximations in the bottom row, respectively. Notice that the textured regions are approximated with a large number of regions. The total number of regions and the compression ratios (without encoding the error) for these images are given in Table 1.

Image	*# Regions before preprocessing*	*# Regions after preprocessing*
cameraman	751	567
sand texture	1230	180
Image	*# C.R. before preprocessing*	*# C.R. after preprocessing*
cameraman	16.19:1.0	21.72:1.0
sand texture	9.87:1.0	54.84:1.0
Image	*C.R. with error coding (no preproc.)*	*C.R. with error* coding (with preproc.)
cameraman	1.64:1.0	1.68:1
sand texture	2.09:1.0	2.44:1.0

Table 1: *Number of regions and compression ratios (without error encoding) for the 'cameraman' and 'sand texture' for the case when images are segmented before and after the preprocessing. The compression ratios when the error is also encoded are given in the bottom two rows.*

The contour maps and image approximations as a result of preprocessing and then segmenting the original 'cameraman' and 'sand texture' images are given in Figure 3. The contour maps are given in the left column and the corresponding image approximations are shown in the right column, respectively. These results show that the textured regions are approximated with a considerably reduced number of regions (see in particular, the grass region in 'cameraman'). It should be noted that this has been accomplished without degrading the accuracy of the rest of the segmentation. The number of regions and compression ratios for the case when the images are preprocessed before segmentation but the corresponding approximation error is not encoded, are also given in Table 1.

4 Residual Entropy Trade-off and Quantizers Design

In our previous work, it was shown that large errors in the approximation occur in the vicinity of edges (which is desirable in the context of image compression because large errors around the edges are masked by human visual system). Thus the error was quantized using a nonlinear quantizer which quantizes large errors coarsely [5]. In order to allocate more bits to the smoothly varying regions, a separate quantizer was designed for these regions.

It is clear that reducing the number of regions to approximate the textured areas will result in higher compression for the base image approximation. However, approximation of the textured regions with a reduced number of sub-regions results in a larger residual (increasing its entropy). This does not necessarily lead to an increase in the number of bits needed to encode the residual, however. Textured areas typically have a large number of edges per unit area, which means that the corresponding approximation error can be quantized very coarsely without affecting the reconstruction quality. In order to exploit this fact, we employ a third nonlinear quantizer (using 3-bits or 8 levels) to quantize portions of the error corresponding to the textured regions. Figure 4 shows the reconstructed images when the quantized error is also encoded. The images in the left column were reconstructed by adding the quantized error (using two quantizers) to the image approximations obtained by segmenting the original images. The images in the right column were reconstructed by adding the quantized error (using three quantizers) to the image approximations obtained by segmenting the preprocessed images. It can be seen that compression ratios (bottom rows in Table 1) as well as the reconstruction qualities in both cases are comparable. The increase in the entropy of the residual as a result of the coarse approximation of the textured regions is compensated for by the coarse quantization of the residual corresponding to the textured regions. This coarse quantization is possible because of the substantial degree of visual masking in the textured areas.

5 Concluding Remarks

A method was presented to preprocess an image to be compressed so that when segmented, textured regions are approximated with substantially fewer regions. This reduction in regions results in a substantial increase in the compression ratios for the base image approximation. It was also shown that the increase in the entropy of the residual as a result of coarse approximation of the textured regions was compensated for by the coarse quantization made possible by visual masking in textured areas. Hence, encoding of the error did not cost more (in terms of bits) compared to encoding of the error resulting from approximation of the textured regions with a large number of regions.

Acknowledgements

This work was supported in part by Hewlett Packard, Lockheed and Pacific Bell through the UC MICRO program, and by the High Quality Imaging and Television Laboratory, an international consortium.

References

[1] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image coding techniques," *Proc. of the IEEE*, vol. 73, no. 4, pp. 549-574, April 1985.

[2] I. Hussain and T. R. Reed, "Compression of still images using segmentation-based approximation" in *Proc. SPIE/IS&T Image and Video Compression*, San Jose, CA, February 6-10, 1994, vol. 2186, pp. 134-145

[3] R.O. Duda, *Pattern Classification and Scene Analysis,* John Wiley, New York, 1973, pp. 114-120

[4] G. Bongiovanni, L. Cinque, S. Levialdi, and A. Rosenfeld, " Image Segmentation by a multiresolution approach," *Pattern Recognition*, vol. 26, no. 12, pp. 1845-1854, 1993.

[5] T. R. Reed, V. R. Algazi, G. E. Ford, and I. Hussain, "Perceptually based coding of monochrome and color images" in *Proc. IEEE Data Compression Conference '92*, Snowbird, Utah, March 24-26, 1992, pp. 142-151.

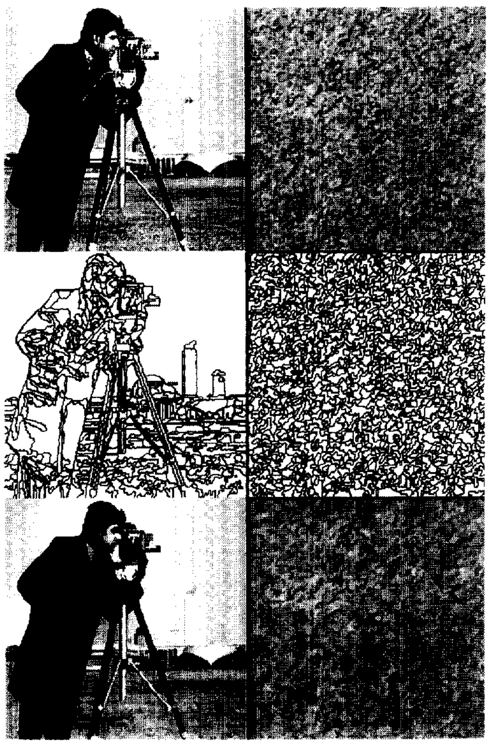

Figure 2: (top row) The original 'cameraman' and 'sand texture' images, (middle row) the contour maps after segmentation, (bottom row) corresponding image approximations, respectively. The original images were not preprocessed before segmentation.

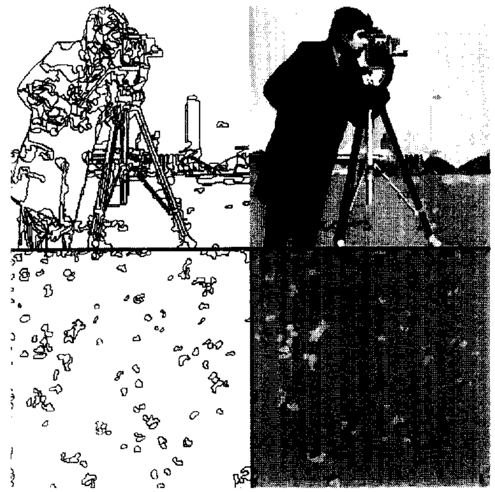

Figure 3: (top row) Contours map and the corresponding image approximation for the 'cameraman' and (bottom row) for the 'sand texture'. In both cases, the original images were preprocessed before segmentation.

Figure 4: (left) The reconstructed images obtained by adding the quantized error (using two quantizers) to the image approximations formed via segmentation of the original images, (right) the reconstructed images after the addition of the quantized error (using three quantizers) to the image approximations generated via preprocessed images.