8

REGION SEGMENTATION AND TEXTURE ANALYSIS

A complementary procedure to edge and boundary detection by locating image discontinuities is to find areas or regions such that the pixels comprising them are homogeneous in some property, and then locate the boundaries of these regions. As an example, if a desired region is known to consist solely of pixels of a certain intensity or color (hue) range, then these pixels can be picked out directly from the image and grouped into one or more connected regions.

Regions may also be defined by having uniform pixel properties computed over larger neighborhoods. The property describing the pattern of pixel attributes over a region will be known as the *texture* of the region. In this chapter, we first discuss region-segmentation techniques using single-pixel attributes, and then the various techniques of texture analysis.

8.1 REGION SEGMENTATION

Two approaches to region segmentation are the "region-growing" and the "region-splitting" techniques. In the *region-growing*, approach, the pixels are first grouped in regions based on the similarities of some attribute, say intensity, and then the resulting regions are examined for merging with the neighboring regions based on their average properties and spatial relationships. In the second approach, large regions are successively split into smaller regions based on finer distinctions between the properties of the pixels contained in them. These techniques are called *region-splitting* or *iterative (recursive)* segmentations techniques. Combination of the two techniques may be called *split-and-merge* techniques.

It should be clear that the region-segmentation techniques always give closed boundaries, by construction. For this reason, it is often easier to use their output for higher-level processing than the fragmented boundary segments obtained by edge-detection procedures. However, the fundamental difficulties of segmentation are caused by factors that are common to both techniques. A comparison of the edge and region-segmentation techniques is presented later in this chapter.

8.1.1 Thresholding and Recursive Segmentation

The simplest technique for image segmentation is that of *thresholding*. The pixels exceeding a certain threshold in some image attribute, say intensity, belong to one group and the rest to a second group. Regions are formed by collecting pixels such that each pixel in a region is a neighbor of one or more pixels in that region. A simple generalization is to form regions of pixels having attribute values within a certain range. Thresholding is well suited for scenes of objects with relatively uniform regions or surfaces against high-contrast backgrounds, such as, dark characters on a white page and a bright aircraft against a relatively dark sky, but it has also been used with more complex images (see [1] for early work with biological cell images).

Selection of proper thresholds is of prime importance. For some applications, such as segmentation of dark characters on a light paper, the fraction of pixels belonging to the desired character regions may be known from a priori statistics. For more general cases, a model of the expected image characteristics is required. A common model, usually assumed implicitly, is that the different object attributes are distributed about two different average values. The distribution functions need not be known, but the number of pixels having values much different than mean is assumed to fall off rapidly with this difference. A histogram of intensity values for such a model with two objects is expected to look characteristically like Fig. 8-1. This histogram is bimodal and a reasonable value for threshold is at the bottom of the valley between the two peaks. A better threshold selection is possible if a priori distributions of the intensity values are known. Experiments with some threshold-selection techniques are reported in [2]; Chow and Kaneko have used a statistical model for their analysis [3].

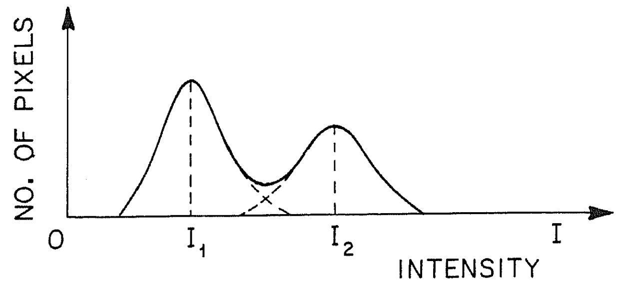


Figure 8-1: Ideal histogram for region with two objects

Ohlander-Price-Reddy Segmentor. An effective segmentor operating on complex natural images, based on some generalizations of the above ideas, has been developed at Carnegie-Mellon University by Ohlander, Reddy, and Price [4, 5]. In this segmentor, a number of attributes of the image pixels are histogrammed. For color images, hue, saturation, intensity, and other parameters derived from them (and dependent on them) are used. Additional features representing texture could be added. The histogram of the attribute having the most distinct peak, according to precedence criteria given below, is chosen for determining the segmentation attribute and range.

This process is repeated for each segmented region and the remaining image until no more new regions are found. Some distinctions between pixels of a single region become more apparent after this region has been separated from a larger region, thus permitting finer segmentation. This technique has been called *recursive segmentation* (a similar technique was independently developed by Tsujii and Tomita [6]).

As an example, consider the image shown in Fig. 8-2. (The image used for analysis is a color image, but only a grey-level equivalent is shown here.) Figure 8-3 shows the intensity histogram for this image. The dark peak in the intensity histogram is chosen for initial segmentation. Figure 8-4 shows the histograms of nine attributes after the points belong to the dark peak are excluded. The green-intensity histogram is now bimodal, and the first peak in it is chosen for further

Figure 8-2: An image (Figs. 8-2 through 8-6, courtesy of Dr. K. E. Price)

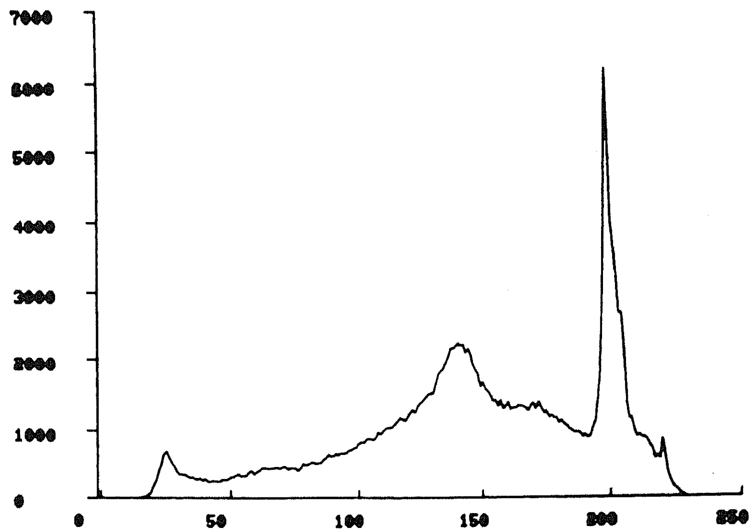


Figure 8-3: Intensity histogram of image in Fig. 8-2

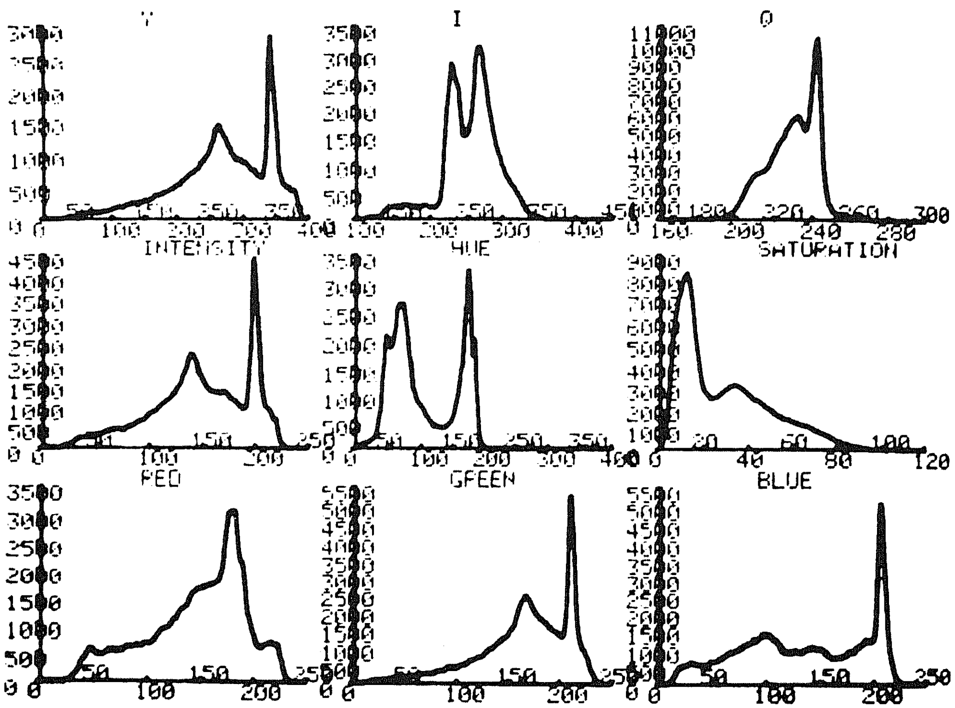
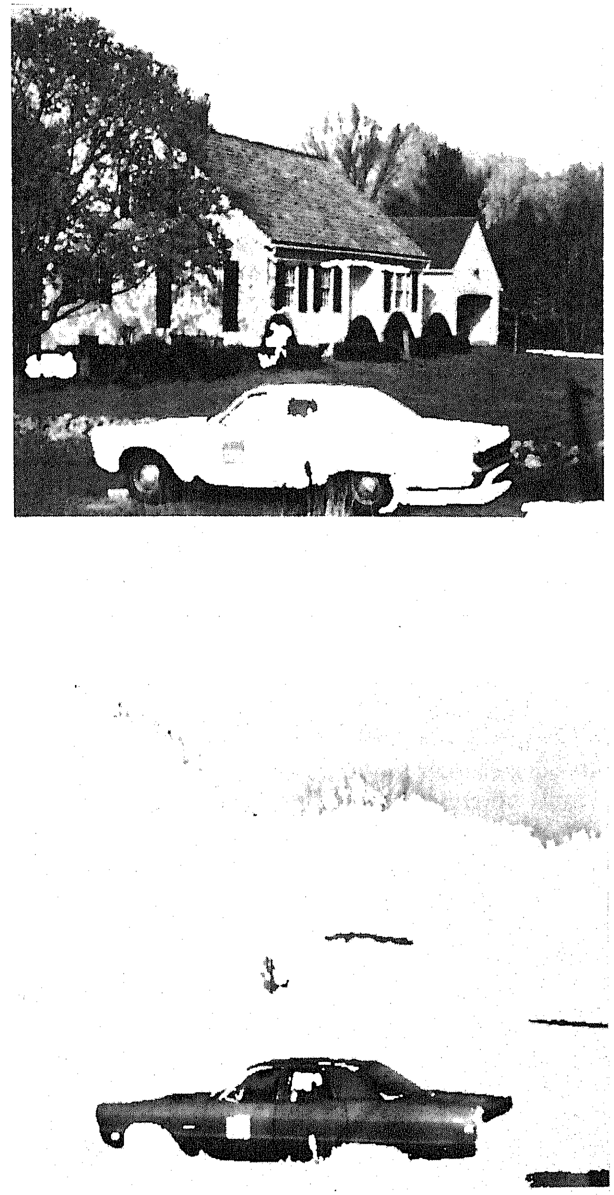


Figure 8-4: Histogram of nine attributes of image of Fig. 8-2 with dark pixels removed



(a)

(b)

Figure 8-5: Regions from Fig. 8-2: (a) pixels in one peak of green histogram, (b) remaining pixels segmentation. Figure 8-5(a) shows the pixels in the first hue peak and Fig. 8-5(b) shows the remaining ones. Figure 8-6 shows the final segmentation.

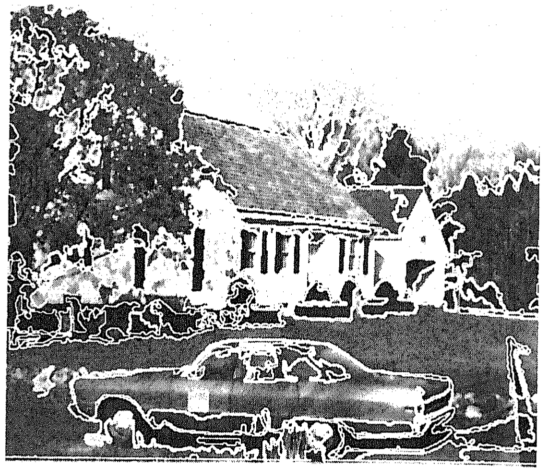


Figure 8-6: Final segmentation for image of Fig. 8-2

Selection of peaks in the image histograms is not always obvious. In the segmentor described by Ohlander et al., peaks are chosen in the order given below:

1. An intensity peak in the low or high quarter of the range. This step removes very dark or bright regions that do not have good chromaticity information.

2. A peak with both minima < 10% of the highest value in the histogram of that attribute and the ratio of peak maximum to minimum exceeding 4. Another peak with maximum/minimum ratio exceeding 2 must also be present.

3. Same as in step 2, except that both minima < 25% of peak value.

4. Two peaks, each having maxima/minima ratio > 2. (This indicates a bimodal distribution.) If both maxima are within 10% of each other, then both peaks are used, at successive steps, else the higher peak is chosen.

5. (This measurre applies to the saturation histogram only.) Both peak minima must be in the lower 20% of the saturation range, maximum/minimum ratio > 2, and these minima must separate another peak with maximum/minimum ratio > 1.2.

6. Peak minima are < 10% of the highest value, and 10% of all points are outside the peak. This and the following condition handle cases where only weak evidence for segmentation is available.

7. Peak minima < 70% of the highest value and maximum/minimum ratio > 1.7.

In practice, computing peak maxima and minima is complicated, as minor perturbations must be distinguished from major changes; here, however, we ignore such details, which may be found in [5].

One deficiency of histogram-based techniques is that small regions in a large image may not produce a distinct histogram peak, even if they are distinct from their surround. Some improvement can be obtained by analyzing smaller parts of an image, say the four quadrants, separately.

The Ohlander segmentor applied to a high-resolution image is computationally expensive. Considerable improvement can be obtained by using *planning*, in the sense used by Kelley for edge detection [7] and described in the previous chapter. Initial segmentation is performed on a reduced resolution image. The resulting region are used as a "plan," from which finer segmentation can be obtained by using higher resolution.

Multiple resolution images, or a pyramid of images, can also be used for region segmentation [8-11]. Improved efficiency is obtained if segmentation can be performed at the higher levels of the pyramid.

Clustering. A generalization of the region splitting technique using multiple attributes is to base the segmentation on all of the image attributes simultaneously, rather than choosing one attribute based on histogram separations. If the different attributes are viewed to span a multidimensional feature space, the pixels in one region should cluster in this space. Many clustering algorithms exist in the pattern recognition literature (for example, see [12]). Clustering is more difficult when the number of clusters is unknown, as is typical for segmentation application. Coleman and Andrews have investigated such segmentation [13]. The performance is determined largely by availability of suitable features for natural images.

8.1.2 Region Growing

In this approach, neighboring pixels whose attribute values are within a fixed predefined range are grouped together to form "atomic" regions. Neighboring atomic regions are then examined for merger based on properties and relations of these regions. Typically, the atomic regions are small and numerous if tight constraints are used on similarities of pixel attributes.

Brice and Fennema suggested a region growing procedure that first merges atomic regions with average properties (such as intensity) within a specified threshold range [14]. This threshold may be different and less stringent than the one used to form the atomic regions. Further mergers are based on relative properties of two neighboring regions. If one region largely surrounds another, the two are merged. If P_1 and P_2 are the perimeters of two neighboring regions and L is the common boundary length between them, they are merged if L/P_1 or L/P_2 exceeds a threshold. Also a merging of two regions resulting in a more "regular" new region is favored.

Performance of such region growers is strongly affected by the choice of various thresholds mentioned above. They have been successfully applied for simple scenes, but for complex textured scenes, the segmented regions tend to be small and scattered over the objects. Region-growing techniques are also described in [15, 16].

Split-and-merge procedures. One approach to reducing the dependence on choice of threshold values is to preserve a complete tree (or more generally a graph) of regions produced at various levels of region growing. The root of this tree is the complete image, and the leaf nodes are the atomic regions. The intermediate levels of the tree contain regions formed by merging of the next-level regions. The higher-level processes may now examine various alternative segmentations rather than a unique output, possibly modifying the segmentations based on possible interpretations. Such techniques are described in [17-19].

The tree describing the relations between regions at different levels is sometimes called a picture tree. The term *quad tree* is used if each pixel at one level is split into four pixels at the lower level. The quad tree is similar to pyramids mentioned earlier, but the term pyramids is more often used to represent grey-level images, where successively higher levels are obtained by averaging the grey-levels of four lower-level pixels.

Pavlidis, Horowitz, and Tanimoto have used the picture tree for a split-and-merge algorithm, where at a given stage of the processing, the regions are examined for further splitting or merging according to predefined criteria. The process switches from merging to splitting if no

more regions can be merged, and vice versa. Experiments with various criteria are described in [20-22]. The split-and-merge procedure can be expected to be more tolerant than a simple region-growing procedure to the effects of initial atomic region formation.

8.1.3 Semantically Guided Region Growing

Performance of region growers can be improved if some a priori knowledge about the scenes being viewed is available. Feldman and Yakimovsky take a statistical view and assume that the a priori probabilities of the image attributes for a given region and of the boundary between two specific regions, such as sky and grass, are known [23]. The segmentation objective is to maximize the probability of correct interpretation assigned to each region; it is formulated as a Bayesian optimization problem. However, the actual solution used is a heuristic approximation of merging the regions with the "weakest" boundary between them at each step, based on some measures of the boundary, the regions surrounding it, and the assumed prior probabilities. Feldman and Yakimovsky present successful results on outdoor scenes containing cars, trees, and sky, and also on biomedical images.

In related work, Barrow and Tenenbaum attempt to combine interpretation with region growing [24]. For each region, a set of interpretations (classifications) is obtained based on its properties. Two neighboring regions with same unique interpretation are merged. If a certain interpretation of a region is incompatible with its neighbors—that is, an interpretation is not allowed to be next to any of the possible interpretations of the neighboring region—then the incompatible interpretation is removed from the list. In a variation, they also associate degrees of compatibilities for given interpretations and search for minimal-cost interpretations.

The main constraint in the use of a priori information for segmentation is the availability of such information, particularly if continuous probability distributions, and not just binary adjacency relations, are required.

8.1.4 Tracing Region Boundaries

It is useful to obtain the boundaries of a region, once the connected group of pixels forming the region have been extracted. A simple algorithm is to scan the image along a horizontal row until a pixel inside the region is found. Then a left turn of one pixel is taken. If we are still in the figure we turn left again, and if the outside of the region is reached a right turn is taken (see Fig. 8-7, where the dots are the pixels within the region). This step is repeated until the starting point is reached again. This algorithm is adequate for most cases, but the traced boundary is sensitive to the starting point. Further, parts of the region that are a single pixel wide may not be traced adequately.

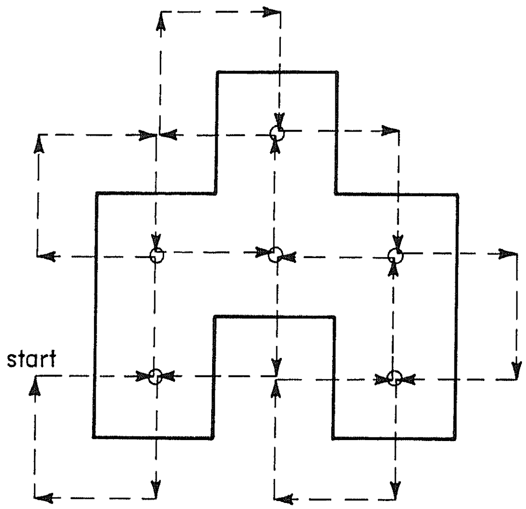


Figure 8-7: Tracing the boundary of a region

These difficulties can be avoided by a more complex algorithm, which chooses the next boundary point by examining all (8-connected) neighbors. The next point is the one with the largest counterclockwise direction from the previous path. A yet more complex algorithm, using table look-up for choosing the next boundary point, is described in [25].

Procedures are described in [26, 27] for tracing boundaries from regions represented in a quad tree, and for constructing quad-tree region representations given the boundary.

8.2 EDGE VERSUS REGION SEGMENTATION

We are now in a position to compare the edge- and region-segmentation approaches. Both techniques are aimed at finding object boundaries by finding boundaries between image attributes such as intensity and color. If useful and precise mathematical definitions of the desired discontinuities were available, we could expect the two methods to yield the same results, but perhaps with differing computational effort. However, in the absence of such models, the two methods implement similar but different intuitive notions of the desired discontinuities.

In the absence of suitable mathematical models for images and desired boundaries, it is difficult to compare the performance of the different segmentation techniques. However, the following qualitative observations can be made. These observations should hold in spite of the differences among the various edge- and region-segmentation techniques themselves.

1. Region segmentation necessarily yields closed boundaries. Edge-based approaches typically give only parts of the boundary segments. However, a tenacious "line follower" could be forced to choose a path at each point until a closed boundary is obtained.

2. Edge detection is an inherently local process. Hence, local failures may prevent complete segmentation; for example, the line-finding systems have difficulty tracing objects with irregular boundaries. On the other hand, since region segmentors are more global, they are less sensitive, and more likely to miss low-contrast boundaries, or small objects. An important case is that of long, thin objects, such as roads, which are small in area and hence have small effects on larger area statistics.

3. Improvement in performance by addition of color seems to have been much more dramatic for region segmentors than for edge-based segmentors. This is related to the observations above, as color is also believed to be used by humans as more of a global property.

4. The position of detected edges is relatively insensitive to the parameters of the procedure, such as a threshold. However, position of region boundaries can be quite sensitive to the choice of range for the segmentation attribute.

These observations imply that region segmentation may be better suited for some tasks and edge segmentation for some others, and that the two approaches are, to some extent, complementary. Homogeneous regions, with irregular boundaries. and possibly surrounded by nonhomogeneous regions, for example bodies of water, such as lakes and rivers surrounded by textured areas in aerial images, are easily detected by simple region segmentors, but present difficulties for line finders. Long, thin features, and finer details within regions, are more easily handled by edge-based approaches. Figure 8-8 shows a region segmentation of the San Francisco image of Fig. 1-9 (the region segmentation used color). It is instructive to compare this segmentation with the line segments detected in the same image and shown in Fig. 7-16. A system that uses both techniques for object location, depending on the features of the desired objects, is described in [28], and some results of this system for the San Francisco image are presented in Chapter 10.

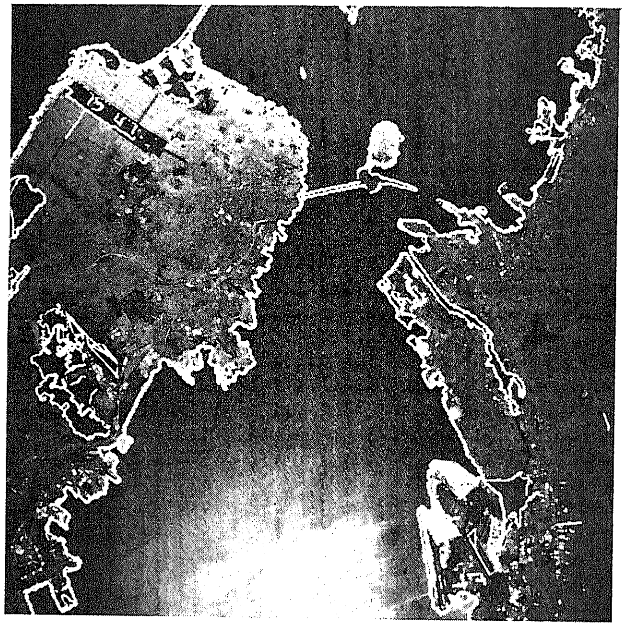


Figure 8-8: Segmented regions in the San Francisco image of Fig. 1-9

Milgram has made an interesting suggestion to combine the two approaches [29]. In this approach, different regions are computed by using a number of threshold ranges. Boundaries of these regions are compared with edges detected in the same image. The region segmentation yielding the maximum overlap of the edge and region boundary points is selected as the final segmentation. This technique has been called *segmentation by convergent evidence* and also a *superslice*

X

technique. In a variation, Milgram and Herman have used clusters in a two-dimensional histogram, whose axes are the pixel grey levels and edge magnitudes [30].

It must be remembered that many of the difficulties of segmentation are due to inherent properties of images, hence are ~~common to all~~ segmentation methods. The desired object boundaries are often essentially three-dimensional depth discontinuities and do not always ~~correspond~~ to image attribute discontinuities. Conversely, some discontinuities in images, such as painted patterns on a surface, do not correspond to object boundaries. Thus, "perfect" segmentation may be unachievable in general, and the higher level must deal with such imperfections.

8.3 TEXTURE ANALYSIS

Surfaces of natural objects are not always homogeneous in a local attribute, such as intensity or color, as has been implicitly assumed in the previously described segmentation techniques. Frequently the desired surfaces have a more or less uniform observed pattern, called the visual texture, possibly generated by the physical texture, as in a rough wall surface, or simply the markings on a surface, as in a wallpaper. In some cases it is natural to view a collection of objects as a single entity—for example, a grass field or a wall of bricks. In these cases, the pattern of individual objects determines the texture of the collection.

Ability to detect and describe surface textures is useful as an important clue in recognition of objects, and also for scene segmentation. However, the texture boundaries need not always correspond to physical object boundaries. Under certain assumptions, the surface texture can also be used to determine the three-dimensional surface orientations, as will be seen in Chapter 9.

For simplicity, first consider the textures contained in the synthetic image of Fig. 8-9. Two distinct regions are seen, and the textures of each can be considered to be characterized by repetition of a primitive element in a certain pattern. The textures differ in having different primitives or a different pattern, or both.

However, natural textures cannot be characterized so easily. Figure 8-10 shows examples of some common patterns. (A photo album by Brodatz contains many examples of natural textures [31].) These patterns have neither an obvious fixed primitive element nor a fixed pattern of repetition, even though each is perceived to be uniform in some sense, and different from the others. Such observations have led to study of texture as a property of the pattern that is uniform in a

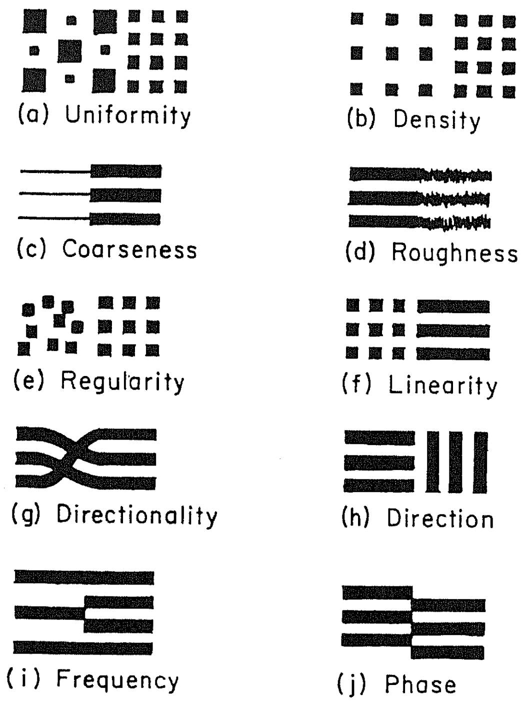


Figure 8-9: Examples of synthetic texture pairs and their distinguishing characteristics (from Laws [34])

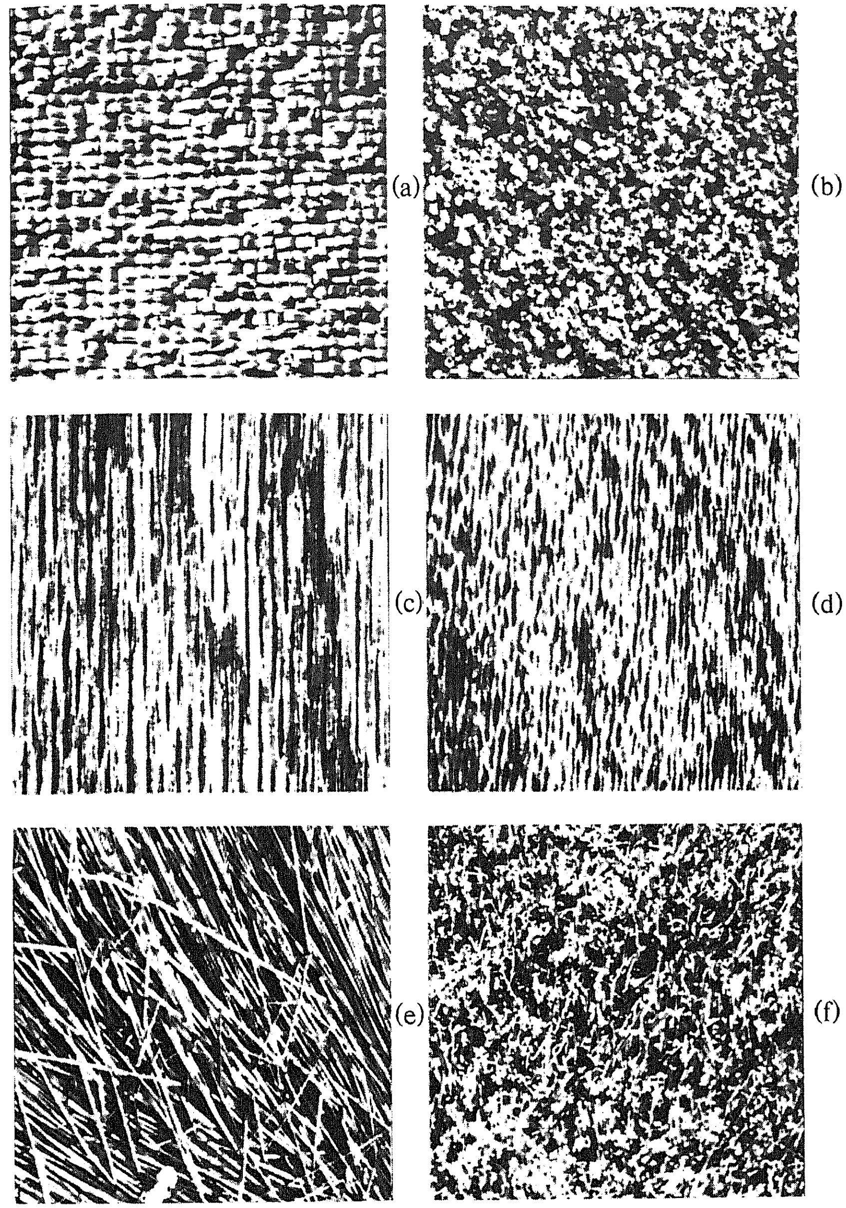


Figure 8-10: Some examples of natural textures: (a) raffia, (b) sand, (c) wood, (d) water, (e) straw, (f) grass

statistical sense, in contrast to a *structural* approach, where the texture primitives and their relationships are described explicitly. The two approaches are not mutually exclusive, as some structural properties can be inferred from statistical analysis, and some structural properties may need to be described statistically for natural textures.

8.3.1 Statistical Texture Measures

The statistical measures have been motivated by a lack of simply described patterns in natural textures. Further, these measures have been supported by an important conjecture, due to Julesz [32], that humans are unable to distinguish textures that have the same second and lower order statistics, but differ in one or more higher-order statistics. (*n*th-order statistics are determined by the joint probability distributions of *n* pixels at a time).

Figure 8-11(a) shows two easily discriminable random textures. The two textures have the same first-order but different second-order statistics. In Fig. 8-11(b), however, the top and bottom textures have the same first- and second-order statistics but differ in the third-order statistics and are not immediately distinguished. Julesz conjecture is supported by a large number of examples, however, several counterexamples have been found [33-38]. These counterexamples have patterns that have some discernible micropatterns and the local second-order statistics differ from the global ones.

Statistical measures are based on the average properties assumed to be invariant over an entire region. The number of suggested statistical measures is large, and only the commonly cited and used approaches will be described here.

First order measures. The simplest measures are based on first-order statistics—that is, probability distributions of single-pixel attributes. Some examples are mean and variance of intensity. More sophisticated first-order measures are based on histograms of the individual pixel attributes. These measures are not strictly texture measures, as they are not even dependent on the spatial distribution of the pixel attributes, but are still useful for many naturally occurring textures.

"Texture energy" measures. An improvement over the first-order measures using pixel attributes is to detect the presence of certain features in the texture and then to compute the first-order statistics of these features. The density of edges, detected by a local edge detector, is commonly used to distinguish between "coarse" and "fine" textures.

Laws [39] has generalized this concept to determining a variety of features by convolving the image with a variety of filter templates, F_1, F_2, ..., F_n, as shown in Fig. 8-12 and then measuring the "energy" of the outputs (E_1, E_2, ..., E_n). The filters consist of 3-by-3 or 5-by-5 masks and detect the presence of edges and lines in various orientations, and cornerlike features. The filters were determined empirically by testing with some natural textures. The energy measures compute properties over a larger window (15-by-15). Simple measures are average and variance of the filtered outputs. As the various measures are not independent, the resulting features can be combined into a smaller set, C_1, C_2, ..., C_n.

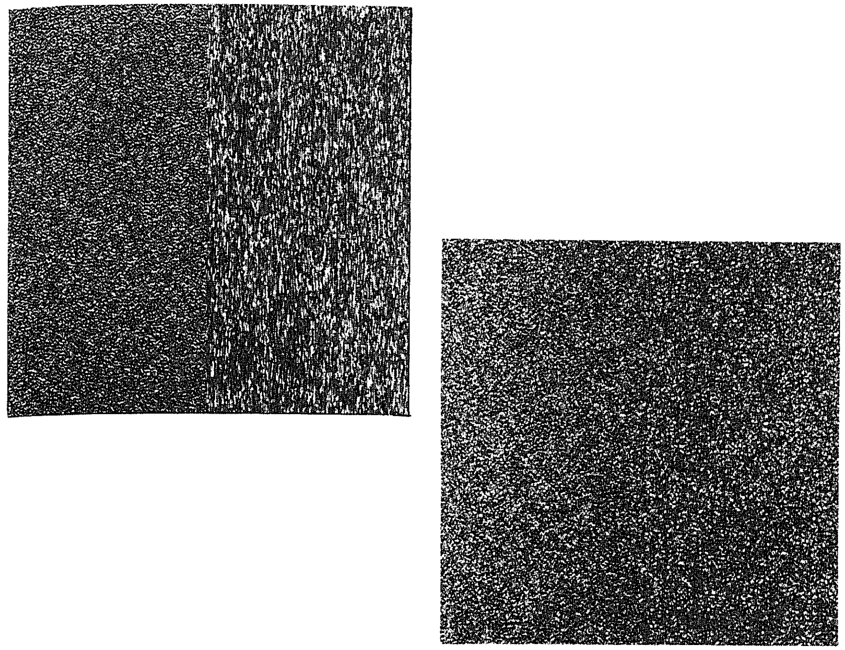


Figure 8-11: Two pairs of synthetic textures (courtesy of Dr. D. D. Garber)

Fourier measures. As textures are viewed to be at least semiperiodic, the Fourier transform of an image window can be expected to have distinct peaks useful for texture discrimination. Bajcsy used filters in the Fourier domain, consisting of annular rings and strips in different orientations [40]. The outputs of these filters was used to generate symbolic descriptions such as bloblike, homogeneous, random, monodirectional, and bidirectional. This technique was applied successfully to natural scenes containing textures such as grass, water, sand, and trees.

A disadvantage of the Fourier approach is that, except for perfectly periodic textures, the energy in the frequency domain is

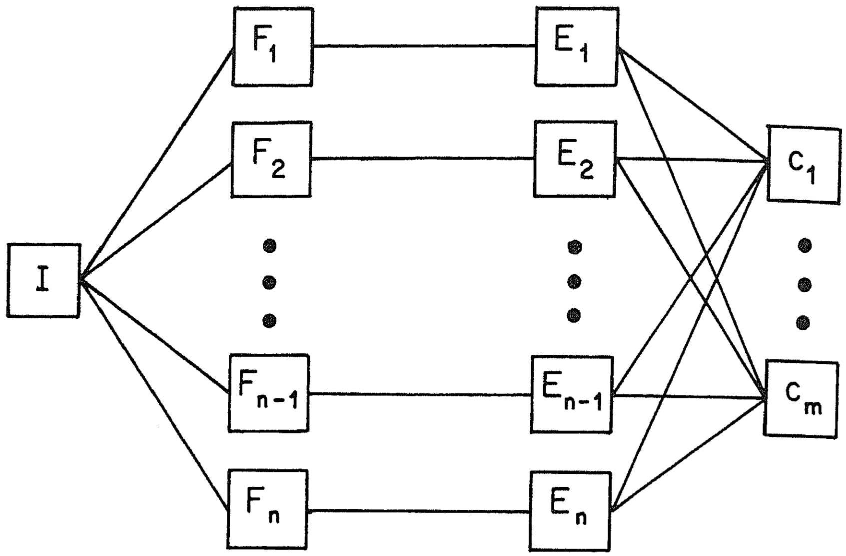


Figure 8-12: Texture energy measurement

scattered, and similar peaks may be caused by a nearly periodic texture and a single strong edge.

Second order measures. Haralick and others suggested a scheme for estimating second-order joint probability densities and devised measures based on them [41]. The second-order statistics are given by $P(i, j, d, \theta)$, the probability of a pair of pixels separated by a distance $|d|$ in direction θ having the intensity values of i and j. These statistics can be computed and stored in the form of *co-occurrence matrices*, one for each value of (d, θ). An element (i, j) of this matrix is a count of the number of pixel pairs with intensities i and j, for the given d and θ values. Note that these matrices are symmetric, as only the absolute value of d is used. Figure 8-13(a) shows a grey-level image window, with intensity values in the range of 0-3, and Fig. 8-13(b) is the corresponding co-occurrence matrix for the horizontal direction and distance of one pixel (that is, $\theta = 0$ degrees and $d = 1$ pixel). Note that computation of complete second order statistics requires co-occurrence matrices for all pairs of values of d and θ.

Haralick and others suggested various features to be computed from a co-occurrence matrix. Three of them are as follows:

$$f_1 = \sum_{i=1}^{N} \sum_{j=1}^{N} P(i, j)^2 \tag{8-1}$$

0	0	2	2
1	1	2	2
0	3	3	3
0	0	1	1

(a)

	0	1	2	3
0	4	1	1	1
1	1	4	1	0
2	1	1	4	0
3	1	0	0	4

(b)

Figure 8-13: An example of a co-occurrence matrix: (a) image grey levels, (b) co-occurrence matrix in horizontal direction (d=1, θ=0 degrees)

$$f_2 = \sum_{n=0}^{N-1} n^2 \sum_{|i-j|=n} P(i, j) \tag{8-2}$$

$$f_3 = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} [ijP(i, j) - \mu_x \mu_y]}{\sigma_x \sigma_y} \tag{8-3}$$

where $P(i, j)$ are the normalized values in the co-occurrence matrix of size N-by-N, μ_x and σ_x are the average and standard deviation along the rows, and μ_y and σ_y along the columns.

f_1 is a measure of uniformity or homogeneity of a region. For a uniform region, the co-occurrence matrix contains a small number of large-valued elements, hence the sum of squares is higher than it would be if all transitions were equally likely. f_2 is a measure of "contrast" and f_3 of the correlation of the intensities. Fourteen measures including the above are given in [41].

These measures were successfully used for classification of different textures such as wood, corn, grass, and water (using four co-occurrence matrices with $d = 1$ and $\theta = 0$, 45, 90, and 135 degrees). However, this technique has many shortcomings. First, if the number of grey levels is large, say 256, the co-occurrence matrix has 256 rows and 256 columns, and a large region is required for useful estimation of the statistics. The number of grey levels can be reduced by compressing the range, but this may introduce texture artifacts. Sometimes only the difference of grey levels is used in computing the co-occurrence matrices. It is also necessary to limit the number of values of d and θ, and methods for automatic choice of these parameters are unclear.

Other features. Faugeras and Pratt have described a different approach to estimation of texture statistics [42]. Their model assumes that underlying texture is generated as an independent, identically distributed array, say $W(j, k)$, and the observed texture, say $F(j, k)$ is obtained by a spatial operator, applied to the array $W(j, k)$. It is possible to estimate $W(j, k)$ from observed $F(j, k)$, by a so-called *whitening* transformation. The optimal linear whitening transformation can be estimated from the autocorrelation function of the observed texture. Pratt and Faugeras also demonstrated that autocorrelation alone is not sufficient for human texture discrimination and that the Julesz conjecture holds for correlated as well as uncorrelated texture fields. In their system, final texture features are derived from measurements on the decorrelated array and from the autocorrelation function.

Davis, Johns and Aggarwal have introduced a concept of *generalized co-occurrence matrices* [43]. These matrices measure the co-occurrence of some selected features in the image, and the co-occurrence is defined by satisfying a selected predicate relationship between the features. Thus, if the selected features are grey levels and the co-occurrence property is that of equality, we get the usual grey-level co-occurrence matrices. However, if the selected features are binary edges and the co-occurrence predicate requires certain angular relations between the edges (such as equality or orthogonality), other properties of texture are measured. Measures similar to those used by Haralick and others for grey-level co-occurrence matrices are used for the generalized co-occurrence matrices.

A variety of other measures have been suggested. An excellent survey can be found in [44]. Use of Markov models or time-series analysis is described in [45, 46]. Other measures using estimation theory methods and models for random placement of elements may be found in [47-50].

8.3.2 Structural Texture Descriptions

The structural approaches to texture attempt to isolate the primitives that form a texture and describe the relations between these primitives in the texture pattern. Such descriptions should distinguish between the two brick patterns shown in Fig. 8-14. However, structural descriptions may be difficult to compute for natural textures, as frequently neither the primitives nor their patterns are completely uniform and regular. Further, texture patterns may be hierarchical; that is, a particular texture pattern, may repeat to form a large texture pattern, and so on. Structural texture description techniques were suggested in early work (see, for example, [51, 52]), but not implemented until recently.

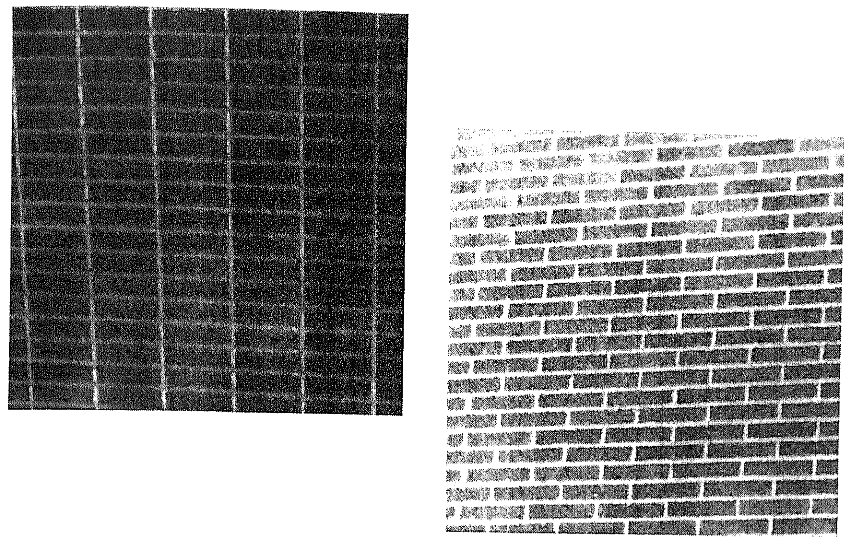


Figure 8-14: Two brick patterns

The structural view of texture is like the viewing of sentences in a language as consisting of certain primitives related by allowed rules of a grammar. Zucker has postulated that natural textures may be viewed as being generated from a two-step process [53]. In the first step, structured patterns, for example a rectangular grid, are generated according to certain rules. In the next step these patterns undergo a transformation that introduces irregularity, in either a deterministic or a random way, to yield natural textures. However, such models are only partially successful in simulating natural textures.

An important step in generating structural descriptions is to find the appropriate primitives of a given texture. These primitives can be expected to correspond to physical objects; however, choice of such primitives requires good segmentation at a detailed level. Since such capabilities do not exist in current systems for complex images, simpler primitives have been used.

Several researchers have used regions of uniform, or near uniform, intensity as primitive texture elements[54, 56]. These elements are computed by the usual region segmentation techniques described earlier. Descriptions of texture elements consist of region properties such as intensity, size, shape, and direction of elongation [54, 55]. Some textures may have random distribution of these element properties, whereas others have elements uniform in one or more of these properties. Further distinctions between textures are based on the relations between the primitives. In the work of Maleson and others [54] the relations used were the collinearity and parallelism of the axes of ellipses approximating the regions. Others have used statistics of *relative vectors* between texture elements [55, 56]. A relative vector is given by the relative coordinates of the line joining the centroids of two texture elements. Statistics of relative vectors have also also been used for synthesis of textures. Nagao and Matsuyama used them to describe artificial, regular textures in a hierarchical pattern [56].

Davis has presented a technique for describing texture pattern of dot textures. The dot patterns themselves may represent objects of interest in a real image; in the example given in this work, they are at the center of the trees in an aerial orchard image [57]. He used peaks in the histogram of the directions of line joining a point to its k nearest neighbors to detect regularity of textures; a square pattern, for example, should have to peaks separated by 90 degrees.

A technique that does not require successful segmentation first is to use line segments or more simply, the local edges. If a texture is periodic we can expect the boundaries of primitives and the local edges to occur at periodic intervals. Some simple properties of textures can be inferred by computing "edge co-occurrence" measures defined to be the number of edges that occur at a distance d in a given direction θ. The edges contributing to the matrix are required to be normal to the direction θ.

Figure 8-15 shows a simple periodic texture and the edges that may be detected in it. The elements bound by edges of opposite contrast can be viewed as comprising the texture elements. In such cases, an edge co-occurrence measure between edges of same orientations can be expected to be periodic with the distance, d, between two edge elements, and a measure between edges of opposite contrast should give the size, s, of the texture elements. d-s

Vilnrotter, Nevatia, and Price have used such plots for analysis of

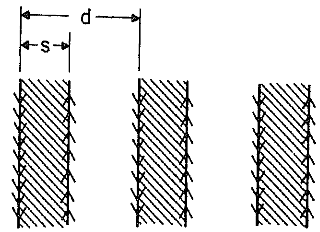


Figure 8-15: Sample texture pattern with edge directions indicated

natural scenes, and their technique seems to be useful for distinguishing between highly periodic, random, and semiperiodic textures [58, 59]. For periodic natural textures, this technique finds the period and the width of the elements, in one or more directions, using the edge co-occurrence plots described above. For non-periodic textures, a primitive element width can still be found for some textures such as grass, water, and sand. The width of the primitives is then used to actually isolate the primitives in the edge image, by searching for edges of opposite contrast separated by the known width. Other properties of primitives, such as length and area, can now be computed. Descriptions of the primitives and their arrangements are used for recognition of the textures and the recognition accuracies are claimed to be quite high, with confusion between only those textures that differ in small detail only. These descriptions have also been used for reconstruction of regular, homogeneous textures.

Measures for inferring properties of textures from edge analysis that correspond to human observations are also described in [60]. Also, Marr has suggested using elements of the primal sketch for texture description but has not specified the grouping properties to be used [61]. Some of the structural properties can also be inferred from an analysis of the grey-level co-occurrence matrices [62].

8.3.3 Comparison of Texture Features

Comparison of various texture features is complicated by the number of parameters to be considered. A large number of texture types are possible. Moreover, different samples of similar objects, such as grass and fields, may have similar but different textures. Ideally, texture features should be invariant to changes within a class, but different from other texture classes. Owing to these difficulties and the large number of suggested techniques, no authoritative comparisons have been reported.

Weszka and Rosenfeld have reported a limited study and conclude that the measures based on co-occurrence matrices are better than Fourier features [63]. Laws claims performance superior to that of co-occurrence features [39]. However, these conclusions are based on the use of a limited set of textures in testing.

Conners and Harlow give a theoretical analysis of the performance of some texture measures for certain types of textures [64]. Not surprisingly, the second order measures are concluded to be superior.

Zobrist and Thompson have studied the use of a linearly weighted combination of a number of texture features [65]. The weights were determined to agree with human judgments about the degree of dissimiliarities between given textures on a training set. Among the Haralick features, f_1, f_2, and f_3 defined in Eqs. (8-1), (8-2), and (8-3) above were determined to be most heavily weighted.

8.3.4 Texture Segmentation

One or more of the texture descriptors, constituting a feature vector, can be used to classify regions into one of the known types. These features can also be used for segmentation by edge or region techniques, analogous to other multidimensional features, such as color. Given two texture-feature vectors T_1 and T_2, we need to decide if they both belong to the same surface. However, texture is not a property of a single pixel, but of a region around it. Thus, if texture is measured by centering a window of a fixed size around each pixel, this window will encompass more than one texture near the boundary. Such techniques may lead to poorly defined boundaries, with possible uncertainty in position equal to half the window size. Another difficulty is in the choice of appropriate window size.

A model of variation in measured texture properties near the edge would be helpful in determining the appropriate window size and precise edge location (see [66]). Owing to the lack of suitable models, the common approach is to assume that the measured texture properties change smoothly from values for one texture to that of another. In this case, the edge detection corresponds to detecting a smooth ramp edge in an intensity image. Using an operator that measures gradient and choosing the gradient peak (that is, nonmaxima suppression), as described in Chapter 7, should lead to correct edge location. Some experiments are described in [66, 67].

The major application of texture analysis has been for images taken from airplanes or satellites. Such images usually have large, highly

textured areas, such as forests and mountains. A common use is for classification of agricultural crops from LANDSAT satellite images.

8.4 SUMMARY

In this chapter we have surveyed region segmentation techniques using single pixel properties and analysis of the texture properties of regions. These segmentation techniques are complementary to the edge and line techniques of the previous chapter. Segmentation of an arbitrary, general scene is difficult, but useful segmentation can be achieved in limited domains, even for complex images. Texture-analysis techniques also are not adequate for all textures, but useful descriptions can be obtained for a small class of known textures. Segmentation using range information is discussed in the next chapter.

REFERENCES

[1] J. M. S. Prewitt, "Object Enhancement and Extraction," in *Picture Processing and Psychopictorics*, B. S. Lipkin and A. Rosenfeld (eds.), Academic Press, New York, 1970.

[2] J. S. Weszka, "A Survey of Threshold Selection Techniques," *Computer Graphics and Image Processing*, Vol. 7, 1978, pp. 259-265.

[3] C. K. Chow and T. Kaneko, "Boundary Detection of Radiographic Images by a Threshold Method," in *Frontiers of Pattern Recognition*, S. Watanabe (ed.), Academic Press, New York, 1972, pp. 61-82.

[4] R. Ohlander, "Analysis of Natural Scenes," Computer Science Department Report (Ph.D. thesis), Carnegie-Mellon University, Pittsburgh, 1975.

[5] R. Ohlander, K. Price, and D. Raj Reddy, "Picture Segmentation Using a Recursive Region Splitting Method," *Computer Graphics and Image Processing*, 1978, pp. 313-333.

[6] S. Tsuji and F. Tomita, "A Structural Analyzer for a Class of Textures," *Computer Graphics and Image Processing*, Vol. 2, 1973, pp. 216-231.

[7] M. Kelley, "Edge Detection by Computer Using Planning," in *Machine Intelligence VI*, B. Meltzer and D. Michie (eds.), Edinburgh University Press, Edinburgh, 1971, pp. 397-409.

[8] L. Uhr, "Layered Recognition Cone Networks That Preprocess, Classify and Describe," *IEEE Transactions on Computers*, Vol. 21, 1972, pp. 758-768.

[9] A. R. Hanson and E. M. Riseman, "VISIONS: A Computer System for Interpreting Scenes," in *Computer Vision Systems*, A. R.

Hanson and E. M. Riseman (eds.), Academic Press, New York, 1978, pp. 303-333.

[10] A. Klinger and C. R. Dyer, "Experiments on Picture Representation Using Regular Decomposition," *Computer Graphics and Image Processing*, Vol. 5, 1976, pp. 68-105.

[11] S. Tanimoto and T. Pavlidis, "A Hierarchical Data Structure for Picture Processing," *Computer Graphics and Image Processing*, Vol. 4, 1975, pp. 104-113.

[12] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1973.

[13] G. Coleman and H. C. Andrews, "Image Segmentation and Clustering," *Proceedings of IEEE*, Vol. 67, No. 5, May 1979, pp. 773-785.

[14] C. R. Brice and C. L. Fennema, "Scene Analysis Using Regions," *Artificial Intelligence*, Vol. 1, Fall 1970, pp. 205-226.

[15] S. W. Zucker, "Region Growing: Childhood and Adolescence," *Computer Graphics and Image Processing*, Vol. 5, 1976, pp. 382-399.

[16] E. M. Riseman and M. A. Arbib, "Computational Techniques in the Visual Segmentation of Static Scenes," *Computer Graphics and Image Processing*, Vol. 6, No. 3, June 1977, pp. 221-276.

[17] R. A. Kirsch, "Computer Determination of the Constituent Structure of Biological Images," *Computers and Biomedical Research*, Vol. 4, No. 3, June 1971, pp. 315-328.

[18] L. J. Krakauer, "Computer Analysis of Visual Properties of Curved Objects," MIT Project MAC Report MAC-TR-82 (Ph.D. thesis), May 1971.

[19] E. Freuder, "Affinity: A Relative Approach to Region Growing," *Computer Graphics and Image Processing*, Vol. 5, 1976, pp. 254-264.

[20] L. S. Horowitz and T. Pavlidis, "Picture Segmentation by a Tree Traversal Algorithm," *Journal of ACM*, 1976, pp. 368-388.

[21] S. L. Tanimoto and T. Pavlidis, "The Editing of Picture Segmentations Using Local Analysis of Graphs," *Communications of ACM*, 1977, pp. 223-229.

[22] S. L. Horowitz and T. Pavlidis, "A Graph-Theoretic Approach to Picture Processing," *Computer Graphics and Image Processing*, Vol. 7, 1978, pp. 282-291.

[23] J. A. Feldman and Y. Yakimovsky, "Decision Theory and Artificial Intelligence: I. A Semantics Based Region Analyzer," *Artificial Intelligence*, Vol. 5, 1974, pp. 349-371.

[24] J. M. Tenenbaum and H. G. Barrow, "Experiments in Interpretation Guided Segmentation," *Artificial Intelligence*, Vol. 8, 1977, pp. 241-274.

[25] S. A. Dudani, "Region Extraction Using Boundary Following," *Pattern Recognition and Artificial Intelligence*, C. H. Chen (ed.), Academic Press, New York, 1976, pp. 216-232.

[26] C. R. Dyer, A. Rosenfeld and H. Samet, "Region Representation: Boundary Codes from Quadtrees," *Communications of ACM*, Vol. 23, No. 3, March 1980, pp. 171-179.

[27] H. Samet, "Region Representation: Quadtree from Boundary Codes," *Communications of ACM*, Vol. 23, No. 3, March 1980, pp. 163-170.

[28] R. Nevatia and K. Price, "Locating Structures in Aerial Images," *Proceedings of International Conference on Pattern Recognition*, Kyoto, Japan, November 1978, pp. 686-690.

[29] D. Milgram, "Region Extraction Using Convergent Evidence," *Computer Graphics and Image Processing*, Vol. 11, No. 1, 1979, pp. 1-12.

[30] D. Milgram and M. Herman, "Clustering Edge Values for Threshold Selection," *Computer Graphics and Image Processing*, Vol. 10, No. 3, June 1979, pp. 272-280.

[31] P. Brodatz, *Textures: A Photograph Album for Artists and Designers*, Dover, New York, 1956.

[32] B. Julesz, "Visual Pattern Discrimination," *IRE Transactions on Information Theory*, Vol. 8, February 1962, pp. 84-92.

[33] S. R. Purks and W. Richards, "Visual Texture Discrimination Using Random Dot Patterns," *Journal of Optical Society of America*, Vol. 67, June 1977, pp. 765-771.

[34] T. Caelli and B. Julesz, "On Perceptual Analyzers Underlying Visual Texture Discrimination: Part I," *Biological Cybernetics*, Vol. 28, 1978, pp. 167-176.

[35] T. Caelli, B. Julesz, and E. N.Gilbert, "On Perceptual Analyzers Underlying Visual Texture Discrimination: Part II," *Biological Cybernetics*, Vol. 29, No. 4, 1978, pp. 201-214.

[36] B. Julesz, E. N. Gilbert, and J. D. Victor, "Visual Discrimination of Textures with Identical Third-Order Statistics," *Biological Cybernetics*, Vol. 31, No. 3, 1978, pp. 137-140.

[37] J. D. Victor and S. Brodie, "Dircriminable Textures with Identical Buffon Needle Statistics," *Biological Cybernetics*, Vol. 31, No. 4, 1978, pp. 231-234.

[38] A. Gagalowicz, "Visual Discriminatio of Stochastic Texture Fields based upon their second Order Statistics," *Proceedings of the Fifth International Conference on Pattern Recognition*, Miami Beach, December 1980, pp.786-788.

[39] K. I. Laws, "Textured Image Segmentation," University of Southern California Report USCIPI 940 (Ph.D. thesis), January 1980.

[40] R. Bajcsy, "Computer Identification of Visual Surface," *Computer Graphics and Image Processing*, Vol. 2, 1973, pp. 118-130.

[41] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Texture Features for Image Classification," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 3, No. 6, November 1973, pp. 610-621.

[42] O. D. Faugeras and W. K. Pratt, "Decorrelation Methods of Texture Feature Extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 2, No. 4, July 1980, pp. 323-332.

[43] L. S Davis, S. A. Johns, and J. K. Aggarwal, "Texture Analysis Using Generalized Co-occurrence Matrices," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 1, No. 3, July 1979, pp. 251-259.

[44] R. M. Haralick, "Statistical and Structural Approaches to Texture," *Proceedings of IEEE*, Vol. 67, No. 5, May 1979, pp. 786-804.

[45] B. H. McCormick and S. N. Jayaramamurthy, "Time Series Model for Texture Synthesis," *International Journal of Computer and Information Science*, Vol. 3, No. 4, 1974, pp. 329-343.

[46] D. Garber, "Computational Models for Texture Analysis and Texture Synthesis," University of Southern California, USCIPI Report 1000, May 1981 (Ph.D. thesis).

[47] K. Deguchi and I. Morishita, "Texture Characterization and Texture-based Image Partitioning Using Two-dimensional Linear Estimation Techniques," *IEEE Transactions on Electronic Computers*, Vol. 27, No. 8, August 1978, pp. 739-745.

[48] B. H. McCormick and S. N. Jayaramamurthy, "A Decision Theory Method for the Analysis of Texture," *International Journal of Computer and Information Science*, Vol. 4, No. 1, 1975, pp. 1-37.

[49] J. W. Modestino, R. W. Fries, and A. L. Vickers, "Stochastic Image Models Generated by Random Tesselations of the Plane," *Computer Graphics and Image Processing*, Vol. 12, 1980, pp. 74-98.

[50] B. Schachter and N. Ahuja, "Random Pattern Generation Process," *Computer Graphics and Image Processing*, Vol. 10, 1979, pp. 95-114.

[51] R. M. Pickett, "Visual Analysis of Texture in the Detection and Recognition of Objects," in *Picture Processing and Psychopictorics*, B. S. Lipkin and A. Rosenfeld (eds.), Academic Press, New York, 1970, pp. 289-308.

[52] J. K. Hawkins, "Texture Properties for Pattern Recognition," in *Picture Processing and Psychopictorics*, B. S. Lipkin and A. Rosenfeld (eds.), Academic Press, New York, 1970, pp. 347-370.

[53] S. W. Zucker, "Toward a Model of Texture," *Computer Graphics and Image Processing*, Vol. 5, 1976, pp. 190-202.

[54] J. T. Maleson, C. M. Brown and J. A. Feldman, "Understanding Natural Texture," *Proceedings of ARPA Image Understanding Workshop*, Palo Alto, Calif., October 1977, pp. 19-27.

[55] F. Tomita, Y. Shirai, and S. Tsuji, "Description of Textures by a Structural Analyzer," *Proceedings of the International Joint Conference on Artificial Intelligence*, Tokyo, August 1979, pp. 884-889.

[56] M. Nagao and T. Matsuyama, *A structural Analysis of Complex Aerial Photographs*, Plenum Press, New York, 1980.

[57] L. S. Davis, "Computing the Spatial Structure of Cellular Textures," *Computer Graphics and Image Processing*, Vol. 11, No 2, October 1979, pp. 111-122.

[58] F. Vilnrotter, R. Nevatia, and K. Price, "Structural Description of Natural Textures," *Proceedings of the Fifth International Pattern Recognition Conference*, Miami, Vol. 2, December 1980, pp. 1142-1144

[59] F. Vilnrotter, "Structural Analysis of Natural Textures," University of Southern California Report USCISG100 (Ph.D. thesis), September 1981.

[60] H. Tamura, S. Mori, and T. Yamawaki, "Textural Features Corresponding to Visual Perception," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 8, No. 6, June 1978, pp. 460-473.

[61] D. Marr, "Early Processing of Visual Information," *Philosophical Transactions of Royal Society of London*, B275, 1976, pp. 483-524.

[62] R. W. Conners and C. A. Harlow, "Toward a Structural Textural Analyzer Based on Statistical Methods," *Computer Graphics and Image Processing*, March 1980, pp. 224-256.

[63] J. Weszka, C. R.Dyer, and A. Rosenfeld, "A Comparative Study of Texture Measures for Terrain Classification," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 6, No. 4, April 1976, pp. 269-285.

[64] R. W.Conners and C. A. Harlow, "A Theoretical Comparison of Texture Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, May 1980, pp. 204-222.

[65] A. Zobrist and W. Thompson, "Building a Distance Function for Gestalt Grouping," *IEEE Transactions on Computers*, Vol. 24, 1975, pp. 718-728.

[66] L. S. Davis and A. Mitiche, "Edge Detection in Textures," *Computer Graphics and Image Processing*, Vol. 12, No. 1, January 1980, pp. 25-39.

[67] W. Thompson, "Textural Boundary Analysis," *IEEE Transactions on Computers*, Vol. 26, 1977, pp. 272-276.