Pattern Recognition Vol. 13, pp. 3 16.
Pergamon Press Ltd. 1981. Printed in Great Britain.

0031-3203/81/0101 0003$02.00/0

A SURVEY ON IMAGE SEGMENTATION*

K. S. FU† and J. K. MUI‡

†School of Electrical Engineering, Purdue University, West Lafayette, Indiana 47907, U.S.A.,
‡Bell Telephone Laboratories, Naperville, Illinois 60540, U.S.A.

(*Received* 31 *October* 1979; *in revised form* 9 *January* 1980; *received for publication* 3 *June* 1980)

Abstract—For the past decade, many image segmentation techniques have been proposed. These segmentation techniques can be categorized into three classes, (1) characteristic feature thresholding or clustering, (2) edge detection, and (3) region extraction. This survey summarizes some of these techniques. In the area of biomedical image segmentation, most proposed techniques fall into the categories of characteristic feature thresholding or clustering and edge detection.

Boundary formation Clustering Edge detection Gradient operator Region extraction Segmentation Thresholding

INTRODUCTION

One of the approaches of automated quantitative cytology is using digital image processing. This approach not only mimics to some extent the human recognition process but also quantify the parameters (for example nucleus area, integrated nucleus density) which are not easily measurable with the human vision system. Image segmentation is the division of an image into different regions, each having certain properties. It is the first step of image analysis which aims at either a description of an image or a classification of the image if a class label is meaningful. An example of the former is the description of an office scene. An example of the latter is the classification of the image of a cancerous cell. Image segmentation is a critical component of an image recognition system because errors in segmentation might propagate to feature extraction and classification.

The applications of image segmentation are numerous[1–9]. Image segmentation has been used in biomedical areas such as in the identification of lung diseases,[10] in automated classification of white blood cells,[11] in detection of cancerous cells[12] and in chromosome karyotyping.[13]

During the past decade, many image segmentation techniques have been proposed.[1–9] These segmentation techniques can be categorized into three classes, (1) characteristic feature thresholding or clustering, (2) edge detection, and (3) region extraction. This survey is performed from the view-point of cytology image processing and is by no means exhaustive. For a more complete bibliography of image segmentation and processing, the reader is referred to Rosenfield's survey papers.[2–9]

One way to define image segmentation is as follows.[14, 15]

*This work was supported by the National Science Foundation Grant ENG 78-16970.

Definition of uniform predicate

Let X denote the grid of sample points of a picture, i.e. the set of pairs

$$\{i,j\}\ i = 1,2,\ldots,N,\ j = 1,2\ldots,M$$

where N and M are the number of pixels in the x and y direction respectively. Let Y be an nonempty subset of X consisting of contiguous picture points. Then a uniform predicate $P(Y)$ is one which assigns the value true or false to Y, depending only on properties related to the brightness matrix $f(i,j)$ for the points of Y. Furthermore, P has the property that if Z is a nonempty subset of Y, then $P(Y) =$ true implies always $P(Z) =$ true.

Definition of a segmentation

A segmentation of the grid X for a uniformity predicate P is a partition of X into disjoint nonempty subsets $X_1, X_2, \ldots, X_N$ such that:

$$\bigcup_{i=1}^{N} X_i = X \quad \text{(i)}$$

$$X_i,\quad i = 1,2,\ldots,N \text{ is connected} \quad \text{(ii)}$$

$$P(X_i) = \text{TRUE} \quad \text{for} \quad i = 1,2,\ldots,N \quad \text{(iii)}$$

$$P(X_i \cup X_j) = \text{FALSE} \quad \text{for} \quad i \neq j \quad \text{(iv)}$$

where X_i and X_j are adjacent.

Zucker[16] summarized the above conditions as follows: the first condition implies that every picture point must be in a region. This means that the segmentation algorithm should not terminate until every point is processed. The second condition implies that regions must be connected, i.e. composed of contiguous lattice points. The third condition determines what kind of properties the segmented regions should have, for example, uniform gray levels. The fourth condition expresses the maximality of each region in the segmentation.

Almost all image segmentation techniques proposed so far are *ad hoc* in nature.[1-9] There are no general algorithms which will work for all images. One of the reasons that we do not have a general image understanding system is that a two dimensional image can represent a potentially infinite number of possibilities. To build a general image understanding system would require the representation and storage of a vast amount of knowledge. Pavlidis[15] commented that an image segmentation problem is basically one of psychophysical perception, and therefore not susceptible to a purely analytical solution. Any mathematical algorithms must be supplemented by heuristics, usually involving semantics about the class of pictures under consideration. Quite often, one must go beyond simple heuristics, and the introduction of *a priori* knowledge about the picture is essential. Pavlidis then quoted the example of the dalmatian dog picture. Without the *a priori* knowledge that the picture consists of a dalmatian dog, most human observers perceive the picture as pure noise. However, when the human observers are told that the picture consists of a dalmatian dog, most can identify it in the picture. Almost all segmentation algorithms are either based on the concepts of similarity (e.g. characteristic feature clustering algorithms) or discontinuity (e.g. edge detection algorithms). Despite the large amount of research effort devoted to image segmentation algorithms, very little is known about how to measure segmentation error besides the simple criteria of the percentage of pixels misclassified.[17] As a consequence, it is still very difficult to answer the question 'how good is a given algorithm?' Therefore, it is not easy to compare different image segmentation algorithms. Further compounding the evaluation process, different authors generally use different data and few authors process more than several hundred images. Unless one specifically implements a given segmentation algorithm and tries it out on one's data, it is very difficult to evaluate from the published results how well it will work for a given set of data. For these reasons, except in very special cases, the authors will not comment exactly how well a given algorithm will work although qualitative statements on the advantages and disadvantages of the approach can be made.

1. CHARACTERISTIC FEATURE THRESHOLDING OR CLUSTERING

1.1 *Thresholding*

(*A*) *Statistical.* Characteristic feature thresholding is a technique widely used in image segmentation. Weszka[18] recently surveyed a number of threshold selection techniques. In its most general form, thresholding is described mathematically as:

$$S(x, y) = k \text{ if } T_{k-1} \leqslant f(x, y) < T_k \qquad k = 0,1,2,\ldots,m \quad (1)$$

where (x, y) is the x and y co-ordinate of a pixel; $S(x, y), f(x, y)$ are the segmented and the characteristic feature (e.g. gray level) functions of (x, y) respectively; $T_0, \ldots, T_m$ are threshold values with T_0 equal to the minimum and T_m the maximum; m is the number of distinct labels assigned to the segmented image. A threshold operator T can be viewed as a test involving a function T of the form

$$T(x, y, N(x, y), f(x, y))$$

where $N(x, y)$ denotes some local property of the point (x, y), e.g. the average gray level over some neighbourhood. Weszka[18] divided thresholding into three types depending on the functional dependencies of the threshold operator T. When T depends only on $f(x, y)$, the threshold is called global. If T depends on both $f(x, y)$ and $N(x, y)$, then it is called a local threshold. If T depends on the coordinate values x, y as well as on $f(x, y$ and $N(x, y)$, then it is called a dynamic threshold.

There are a number of global threshold selection schemes. Some are based on the characteristic feature (e.g. gray level) histogram, others are based on local properties such as the gradient or Laplacian of an image. For an image consisting of object and background where the percent of the object area is known, Doyle[19] suggested the 'p-tile' method which chooses as a threshold the gray level which most closely corresponds to mapping at least $(1\text{-}p)\%$ of the gray levels into the object. If, for example, dark objects occupy 20% of the picture area, then the image should be thresholded at the 80th percentile, or, more precisely, at the largest gray level allowing at least 20% of the picture points to be mapped into the object. This method is not applicable if the object area is unknown or varies from picture to picture.

For segmenting images of white blood cells, Prewitt and Mendelsohn[20] chose thresholds at the valleys on the gray level histogram. This technique, called the mode method, involves smoothing of the histogram into a predetermined number of peaks (modes) and placing thresholds at the valleys between peaks. The mode method has the advantage that it minimizes the probability of misclassifying an object point as background or vice versa[1] (assuming bi-modal distribution, both background and object are Gaussian distributed). In general, if we threshold at the bottom of a valley in the smoothed histogram, the results are relatively insensitive to the exact choice of the threshold level since the gray levels at a valley bottom are relatively unpopulated. However, there are a number of disadvantages. First, the knowledge of the class of pictures we are dealing with is assumed to be known since many pictures may have the same histogram and thresholding the pictures may give results that are not necessary meaningful. Second, no spatial information is used to arrive at the thresholds which means that there is no guarantee that the segmented regions are contiguous. Third, the method of finding minima between modes by smoothing, depending on the smoothing method used, may smooth out small modes. Nevertheless, for a wide class of images, this method works reasonably well (for example, when there are isolated white blood cells in the image, see Fig. 1). However, in many cases, even when the histogram is bimodal, it may be difficult to accurately locate the valley bottom, since the valley may be broad and flat. Weszka *et al.* proposed ways to sharpen the valley between the two modes, by histogramming the gray values of points whose Laplacians[21] or gradients[22] are in the p-tile range. Rosenfeld *et al.*[23] proposed using iterative histogram modification to sharpen the peaks in enhancing images and their histograms.

Local properties can be used to select a global threshold. Watanabe *et al.*[24] used the gradient approach to determine a global threshold. For each gray level z, compute

$$d = \sum_{(x,y)\in S_z} p'(x, y) \tag{2}$$

where S_z is the set of points having gray level z; $p'(x, y)$ is the magnitude of the gradient at the point (x, y). The threshold is then chosen at the level z for which d_z is the highest. Since this level has a high proportion of large difference points, it should occur just at the borders between objects and background. Intuitively, this method would not work well on some images because equation (2) is not normalized by the number of pixels in S_z. So if S_z consists of a large number of pixels, although $p'(x, y)$ may be small for every (x, y) in S_z, their sum is not necessarily small. Watanabe achieved good results on a data base of cervical smear images, but poor results were obtained by Weszka *et al.*[25] on images of chromosomes, handwriting, and cloud cover and also by Aggarwal *et al.*[26] on pap smear images.

Ohlander[29] used nine one-dimensional histograms of features such as color intensity for red, green and blue, overall intensity (the average of the three colors at each pixel), hue, etc. to segment natural scenes. Thresholding on values corresponding to valleys bounding sharpely defined peaks in a histogram furnished clusters of points which were uniform for the given feature. These regions were then thresholded. The procedure was repeated until all pixels were segmented.

Geometric information can be used to refine the result of segmentation. Brenner *et al.*[35] segmented white cells in bone marrow images by first thresholding the image and then examining the shape of the resulting boundary. Using a graph of the Gallus eight-chain of the boundary, Brenner *et al.* successfully distinguished between "notches" which were the cusp formed when two convex objects touched each other and the vacuoles which were near the boundaries of white cell. By joining appropriate pairs of 'notches', they were able to isolate the white cell of interest. Similarly, irregular boundary produced by vacuoles near the cell boundary could be repaired. Shrinking and expansion (or sometimes called contraction and dilation) were used by a number of researchers (e.g.[36]) to perform segmentation.

Young and Paskowitz,[32] and Ingram and Preston[34] segmented blood cell images by logically combining three threshold images taken at three different wave lengths. Recently Weszka *et al.*[26] proposed two threshold evaluation techniques. One is based on gray level co-occurrence matrix, the other is based on percentage of misclassification. For descriptions of local and dynamic threshold selections, see Weszka.[18]

(B) *Structural.* Tomita *et al.*[30] and Tsuji *et al.*[31] described a method for detecting texturally homogeneous regions based upon uniform values of some local picture property. In their approach, pictures were first segmented into "elements" or "atomic regions" e.g. connected components of constant gray level. A set of properties such as shape, size, position and density was measured for each atomic region. For each property, a histogram was constructed. When the histogram consisted of a small number of distinct modes, it was plausible that there existed a partition of the picture into regions whose elements had similar property values. These modes were then separated by establishing thresholds in the valleys between the peaks. Elements whose property values belong to a given mode were tagged and this gave rise to clusters of similarly tagged elements in the picture. Next, some heuristics were applied to connect similarly labelled elements into regions. The above procedure was then applied to newly formed regions until the histograms of the region descriptors did not show any valleys. The approach was tested out on a few highly artificial pictures (e.g. a white cube with black dots on its surfaces) and no real data example was given.

Keng and Fu[33] and Keng[37] used syntactic techniques to recognize highways, rivers, bridges and commercial/industrial areas from LANDSAT images. They used different LANDSAT bands for recognition of concrete-like objects and water-like objects. The primitives were obtained by thresholding the different bands. The images consisting of these primitives were then smoothed. A finite state automaton or a set of templates was used to perform the recognition of line objects such as highways, rivers and bridges; commercial/industrial areas were obtained by subtracting highway from the thresholded "concrete" image. In Keng,[37] a tree automaton was used to process textural primitives of terrain and tactical targets from LANDSAT and infrared images respectively.

1.2 *Clustering*

Clustering of characteristic features applied to image segmentation is the multidimensional extension of the concept of thresholding. Typically, two or more characteristic features are used and each class of regions is assumed to form a distinct cluster in the space of these characteristic features. A clustering method is used to group the points in the character-

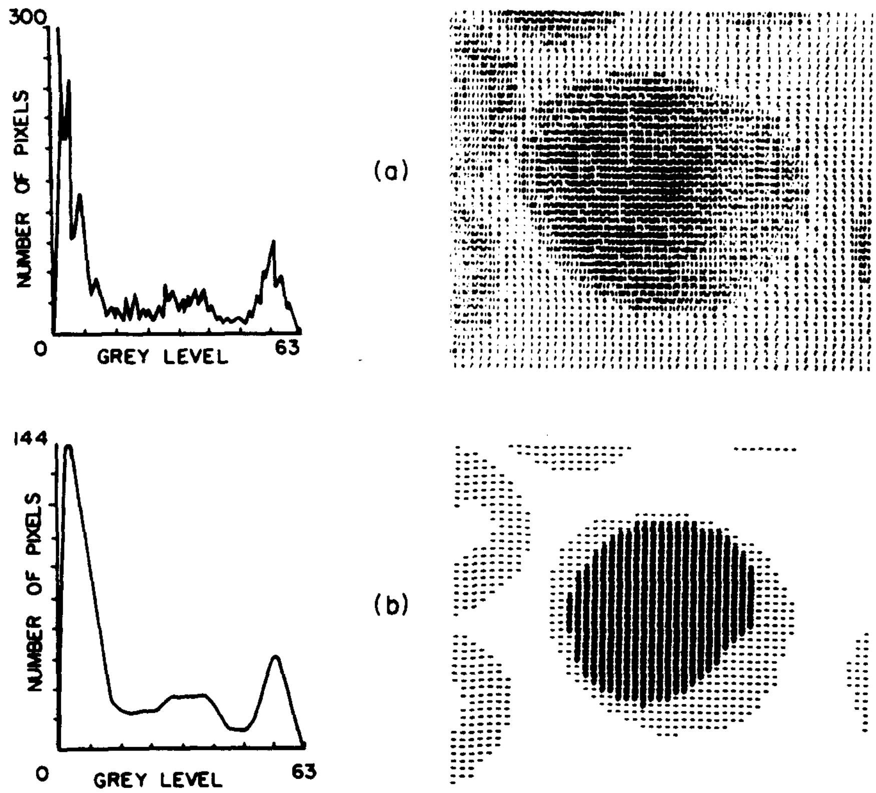

Fig. 1. A successful example of the segmentation technique of smoothing a monochromatic optical density histogram into three peaks and two valleys. (a) Original histogram and computer printout of image, (b) smoothed histogram and computer printout of segmented image.

istic feature space into clusters. These clusters are then mapped back to the original spatial domain to produce a segmentation of an image. The characteristic features that are commonly used in image segmentation by clustering not only include gray values through different filters as in white blood cell image segmentation, it may include any feature that one thinks is helpful to this segmentation problem; for example, texture measures defined on a local neighborhood[46, 49, 50] may be used. The reason one wants to use two or more characteristic features to perform image segmentation is that sometimes there are problems not resolvable with one feature that can be resolved with 2 or more features. Figure 2 illustrates a case where in two dimension feature space (x, y) the clusters can be easily separated, but in each of the one dimensional projections, there is a high degree of overlap of the two distributions corresponding to the two classes of regions such that no valley may exist between the modes of the distributions. Conversely, one may easily resolve two modes in one dimension feature space. This is equivalent to a decision boundary which is a straight line in two dimension space. Figure 3 illustrates this idea.

The use of clustering techniques to perform image segmentation goes back to as early as 1969. Wacker[38] used a clustering algorithm proposed by Swain and Fu[39] to find boundaries in remote sensing data which had up to 12 channels of information. He divided the image into windows of size about 20 × 20 called "boundary cells". For each "boundary cell", the boundary finding algorithm was applied. His boundary finding algorithm consisted of a clustering algorithm followed by an edge finding algorithm. The input to the clustering algorithm consisted of M_m, the maximum number of modes permitted and a threshold T which determined if two modes were distinct or not. The clustering algorithm first established M_m initial mode center $\mathbf{m}_k$, $k = 1, 2, \ldots M_m$ where

$$\mathbf{m}_k = \boldsymbol{\mu} + \boldsymbol{\sigma}\left(\frac{2(k-1)}{(M_m - 1)} - 1\right) \tag{3}$$

such that $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_L)$ and $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_L)$ were the sample mean and variance for each of the L dimensions respectively. (L was the number of channels used.)

Each vector of a "clustering cell" (a "clustering cell" is a window slightly larger than a "boundary cell") was classified to the nearest mode center using a minimum Euclidean distance rule. After all the vectors of a "clustering cell" were classified, a new mode center for each mode was computed and this classification process iterated until there was no change in mode

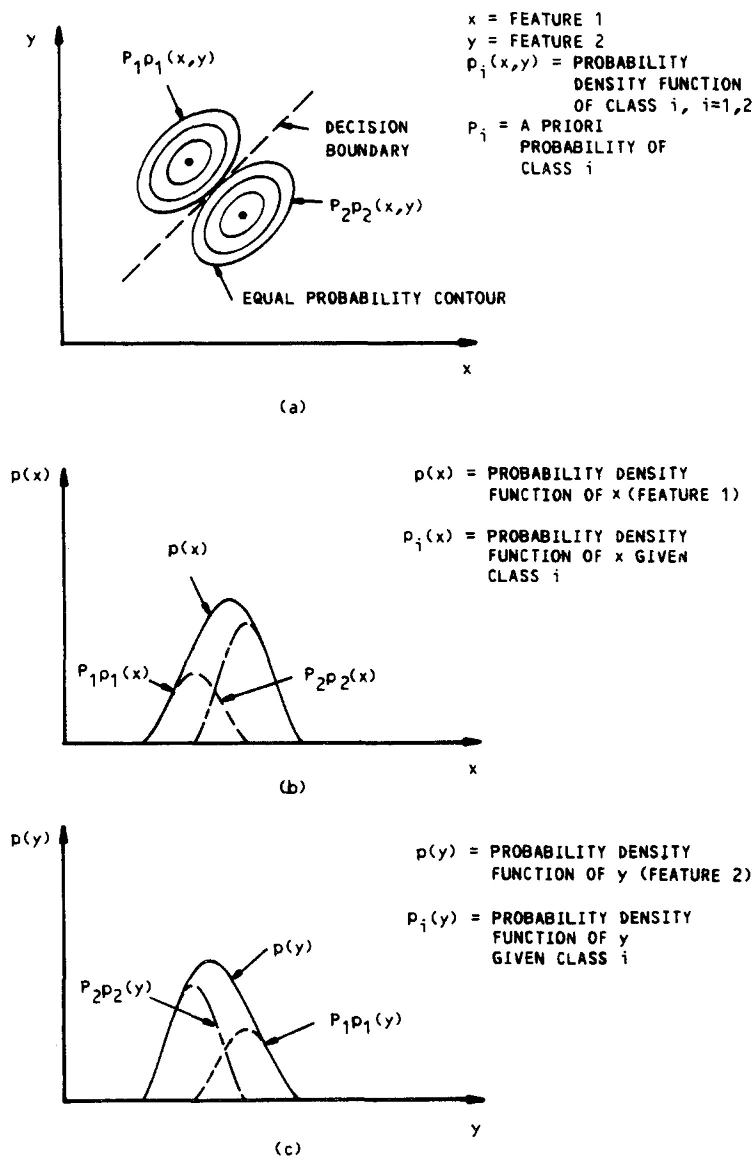

Fig. 2. Illustration of an example where one can easily resolve the two clusters in two dimension feature space (a), but not in each of the one dimension feature spaces (b) and (c).

assignment of the vectors. A pairwise measure of separation was then computed for each pair of modes. Let S_{ij} be this measure for modes i and j. If $S_{ij} \geqslant T$ for all i, $j = 1,2,\ldots, M$ (where M was the present number of modes), then the M modes were taken to be distinct and the clustering procedure was terminated. If one or more $S_{ij} < T$, then the two modes corresponding to the smallest S_{ij} were merged yielding $M - 1$ modes. The clustering procedure was then repeated with $M - 1$ modes until all the modes were distinct. The results of clustering were then mapped back to the original spatial domain and a procedure called Edge Finding Algorithm was initiated to find the boundaries between segmented regions. A correlation distance k was introduced to smooth out some noisy results. A vertical boundary existed between pixel (i, j) and $(i, j + 1)$ if the first $(k - 1)$ pixels to the left of (i, j) all belong to one mode, and $(i, j + 1)$ and the first $(k - 1)$ pixels to the right of $(i, j + 1)$ all belong to another mode. Horizontal boundaries were found in an analogous manner. Since no knowledge of the scene was assumed in Wacker's approach,[38] a great deal of computation time was spent in determining the number of clusters. Also, the iterative assignment of pixels until there is no change in mode assignments is a costly computation process. Further reduction in computation time can be achieved by dealing with the histogram rather than with the pixels individually.

Haralick and Dinstein[51] proposed a spatial clustering technique applicable to multi-spectral image data. The procedure thresholded the gradient image,

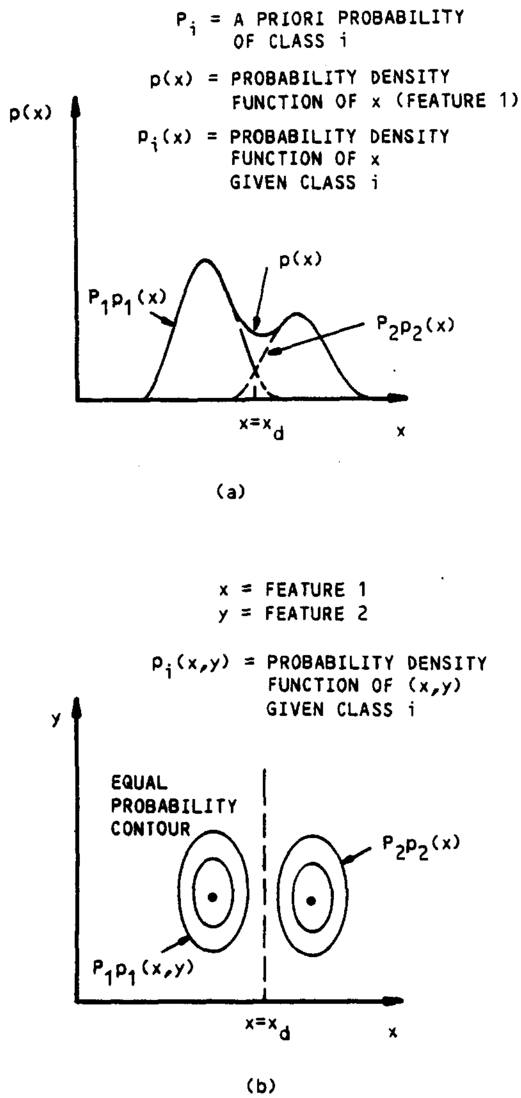

Fig. 3. Illustration of the idea that distributions that can be resolved in one dimension feature space can also be resolved in two dimension feature space. In (a) $x = x_d$ is the decision threshold. This is equivalent to a decision line $x = x_d$ in the two dimension feature space.

cleaned the threshold image, labeled the connected regions in the cleaned image and clustered the labeled regions. Each labeled connected region was assigned a class code by the clustering algorithm. One of the critical problems of the above approach is the problem of distinct spatial clusters merging because of a low gradient bridge between them.

Schachter *et al.*[43, 44] used clustering to perform image segmentation on multispectral remote sensor imagery. In Schachter *et al.*,[43] they used the gray level values of several channels as features in the feature space. In Schachter *et al.*,[44] they tried instead to use just one monochromatic image, using features such as mean gray level, median filtered minimum total variation, and mean typicality computed over a 3 × 3 local neighbourhood to do clustering. They concluded that the results of using one monochromatic image with locally computed features were not as good as those obtained with gray level values of several channels as features.

Carlton and Mitchell[46] used texture and gray level information for image segmentation. They used a texture measure that counted the number of local extrema in a window centered at each pixel. Using three thresholds called "low", "medium" and "high", they produced three intermediate "gray level" pictures whose values are the number of local extrema (averaged over a window) produced by that threshold. These intermediate pictures were used to derive the number of segments in which to divide the original image. The segmentation was then performed by assigning each pixel in the original image to a region by using a four-dimensional distance measure on the intermediate pictures, comparing each pixel to each selected segment. This process was then repeated in a hierarchical structure using decreasing window sizes so that smaller regions within the larger ones were defined. Only one example, an aerial scene of a military simulation area was given. There are several parameters which are critical to the success of this approach, namely extreme size, window sizes, and distance similarity criteria. These parameters are not easily determined and they may vary from picture to picture depending on the details one wants to segment.

We proposed to segment blood cell neutrophil images using iterative segmentations.[52] With some basic assumptions on the image such as that its intensity histogram generally has three peaks and two valleys, we obtained an initial segmentation. Based on this initial segmentation, we determined some critical information from the image such as the presence or absence of points of maximal concavity which would help to determine the number of clusters. The initial segmentation also helped in determining the initial locations of cluster centres of the clusters in the bivariate "color-density" histogram. Segmentation was achieved by dividing the "color-density" histogram into a number of clusters and these clusters were mapped back to the spatial domain. Of the neutrophil images, 97% were correctly segmented based on a data set of 378 images.

Aggarwal *et al.*[26] used a combination of thresholding and clustering techniques to segment cervical smear images. Thresholding at a pre-set gray level was used to extract the nucleus. A ceiling lowering clustering technique on the bivariate histogram was used to segment the cytoplasm. A success rate of 87.6% in extracting the nucleus and a success rate of 88.1% in isolating cytoplasm based on 233 scenes were achieved.

Cahn *et al.*[97] separated the cytoplasm from background in cervical cell images by thresholding the images based on the stability of the perimeter of the cell as the threshold was varied. Once the cytoplasm threshold was determined, cytoplasm and nucleus were segmented by clustering into three classes, namely cytoplasm, folded cytoplasm and nucleus. Evaluation of the proposed technique was based on the results of classifications using the automated segmentation technique described above vs manual segmentation. Manually thresholded cells were classified correctly 66.0% of the time for the 13-class problem and 95.2% of the time on the two-class (normal-abnormal) problem. The automated technique was 52.9% and 90.0% correct, respectively based on 1500 cervical cells that belong to one of eight normal classes and five abnormal classes.

Goldberg and Shlein[45] proposed the idea of clustering on the histogram of 4 bands of multispectral images. The scheme initially identified the most separable clusters in the data. It then ran on an interactive basis allowing the user to split specific clusters into subclusters at the expense of less separability.

Coleman[49] proposed a bottom up procedure for image segmentation using clustering. A number of features such as gray level values through the red, green and blue filters, texture features and nonlinearly filtered features were used. The features were decorrelated using Karhunen–Loeve rotation. The basic procedure was a K-means clustering algorithm which converged to a local minimum in the average squared inter-cluster distance for a specified number of clusters. The algorithm iterated on the number of clusters, evaluating the clustering based on a parameter of clustering quality.

Recently Yoo and Huang[50] proposed an image segmentation algorithm based on graph theoretic clustering. They used the gray level histogram and three different feature pairs which were mean–standard deviation, local minimum–local maximum and an eigenvalue pair of the local characteristic matrix. These feature pairs were extracted from gray levels within a 3×3 local window. An unsupervised, non-iterative and non-parametric clustering technique based on graph theory[92] was employed to group the features into clusters. The results of clustering were mapped back to the original spatial domain. Segmentation results based on feature pairs were more homogeneous than those based on histograms. A number of examples were given such as 'the girl', forward looking infrared images of tank and armored personnel carriers, a military testing site, etc. The results produced by this technique were better than Carlton and Mitchell's[46] on the image of a military testing site. There are a number of questions still unanswered: not all feature pairs would work properly in segmenting a given image and no automatic technique is available to determine which feature pairs should be used. The choice of clustering parameters also poses a problem.

Since one of the shortcomings of using characteristic feature clustering as an approach to image segmentation is that no spatial information is used in performing the cluster analysis, sometimes the extent of clusters in feature space may be ambiguous. To correct these shortcomings, Nagin *et al.*[53] proposed a relaxation scheme in which the probability of a pixel belonging to a number of classes iterated using a compatibility relation. No results were published. The computation time is exorbitant because it takes a number of iterations for the relaxation scheme to converge.

1.3 *Comments on characteristic feature thresholding or clustering approaches to image segmentation*

The philosophy of this approach is basically a global one because some aggregate properties of the features are used. The similarity of feature values of each class of a segmented region form a 'mode' in the feature space. This technique is more immune to noise than, for example, edge detection techniques. Also, this technique gives closed boundaries although sometimes it is necessary to smooth out some of the noisy boundaries.[38, 50] Since this approach is based on the assumption that different classes of segments of an image are represented by distinct "modes" in the distribution of suitably chosen features extracted from the image, the technique will fail if this assumption is not true. Another drawback is that because, in general, the number of segments is not known, an unsupervised clustering scheme may not produce the right number of segments. Besides gray level values, other features are generally image dependent and it is not clear how these other features should be defined in such a way as to produce good segmentation results. Furthermore, most researchers who used this approach generally did not use the spatial information inherent in a image. Although attempts have been made to utilize such information[51,53] the results so far are no better than those that do not use spatial information.[50,52]

2. EDGE DETECTION

Edge detection is a picture segmentation technique based on the detection of discontinuity. An edge or boundary is the place where there is a more or less abrupt change in gray level. Some of the motivating factors of this approach are: (1) most of the information of an image lies on the boundaries between different regions,[1] and (2) biological visual systems appear to make use of edge detection, but not of thresholding.[1] Davis,[42] Riseman *et al.*,[54] Rosenfeld *et al.*[1] and Pavlidis[15] surveyed a number of edge detection techniques. Davis[42] categorized edge detection techniques into two categories, parallel and sequential. By a parallel solution to the edge detection problem, it is meant that the decision of whether or not a set of points is on an edge is made on the basis of the gray level of the set and some set of its neighbours; but the decision is not dependent on first deciding if other sets of points lie on an edge. So the edge detection operator in principle can be applied simultaneously everywhere in the picture. By a sequential solution to the edge detection problem, it is meant that the result at a point is contingent upon the results of the operator at previously examined points. It should be noted that these definitions are with respect to edge element extraction. To produce a closed edge or boundary, the edge elements extracted have to be connected together to form closed curves. Although the method of generating edge elements is parallel in nature (e.g. gradient operator), the method of connecting these extracted edge elements could be sequential in nature (e.g. heuristic search tree) depending on the method used; this method will be referred to as a parallel scheme. Parallel edge detection schemes can be broken down generally in two steps (A) edge element extraction and (B) edge element combination (or called "streak" boundary formation). Edge element extraction methods can be categorized as (A.1) high-emphasis spatial frequency filtering, (A.2) gradient operators, and (A.3) functional approximations. Edge element combination consists of eliminating false edge elements and merging the edge elements into longer edge segments called streaks, eliminating false streaks, combining the streaks into boundaries and eliminating false boundaries. Some techniques (e.g.[40]) require the thinning or skeletonizing of the edge elements before they are combined together. Edge element combination is generally carried out by three classes of techniques (B.1) heuristic search and dynamic programming, (B.2) relaxation, and (B.3) line and curve fitting. Most sequential techniques incorporate edge element extraction as part of the process of boundary detection[67–69] so there is no need for a separate edge element extraction process. However, sequential techniques such as heuristic search[67,68] may also be used for boundary formation.

2.1 *Parallel techniques*

(A) *Edge element extraction*

(A.1) *High-emphasis spatial frequency filtering.* Since high spatial frequencies are associated with sharp changes in intensity, so one can enhance or extract edges by performing high-pass filtering: i.e. take the Fourier transform of the picture, say $F(f(x, y)) = F(u, v)$ where $f(x, y)$ and $F(u, v)$ are the original gray level function and its Fourier transform respectively, F is the Fourier operator. Multiply F by the linear spatial filter H: $E(u, v) = F(u, v) \cdot H(u, v)$ and take the inverse transform $e(x, y) = F^{-1}(E(u, v))$ where $e(x, y)$, is the filtered picture of $f(x, y)$ and $E(u, v)$ its Fourier transform and F^{-1} is the inverse Fourier transform operator. The problem here is filter design.

(A.2) *Gradient operators.* The gradient operator is defined as

$$\nabla f(x, y) = \frac{\partial f}{\partial x}\mathbf{i} + \frac{\partial f}{\partial y}\mathbf{j} \tag{4}$$

where

$$|\nabla f(x, y)| = \left(\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right)^{1/2} \tag{5}$$

and the direction of $\nabla f(x, y)$ is

$$\tan^{-1} \frac{\left(\dfrac{\partial f}{\partial y}\right)}{\left(\dfrac{\partial f}{\partial x}\right)} \tag{6}$$

where f is the original gray level function; $\mathbf{i}$ and $\mathbf{j}$ are unit vectors in the positive x and y directions respectively.

Quite a few proposed edge detection techniques[55–58, 93–95] are based on the digital approximations on variations of equation (4) which will produce a high magnitude where there is an abrupt change in gray level and a low magnitude where there is little change in gray level. Roberts' cross operator[93] is based on a 2 × 2 window

$$\begin{aligned} g(i,j) = [(f(i,j) - f(i+1,j+1))^2 \\ + (f(i+1,j) \\ - f(i,j+1))^2] \end{aligned} \tag{7}$$

where $f(i,j)$ and $g(i,j)$ are the gray level function and magnitude of gradient of point (i,j) respectively.

The operator requires that there is a distinct change in intensity between two adjacent points in the gray value function, so only very sharp edges with high contrast between the surfaces which form the edges will be detected. This method cannot detect ill defined edges (edges which are formed by a gradual change in intensity across the edge). Since the computation is based on a small window, the result is quite susceptible to noise. Kirsch's,[56] Sobel's,[57] and Prewitt's[55] operators are based on a 3 × 3 neighborhood. The main difference between these operators are the weights assigned to each element of the 3 × 3 template.

An adaptive local operator was proposed by Rosenfeld *et al.*[94,95] The procedure involved taking averages over neighbourhoods of sizes $2^k \times 2^k$ at every point (x, y) in the image. For each size neighbourhood, at each point (x, y), differences between pairs of averages corresponding to pairs of nonoverlapping neighbourhoods just on opposite sides of the point in 4 directions (horizonal, vertical and the left and right diagonals) were computed. Of the 4 differences corresponding to the 4 directions, the one that gave the highest absolute difference was selected. At each point, a best size neighbourhood which was defined to be the largest size neighbourhood for which the next smaller size did not give a significantly higher absolute difference was chosen. Specifically, if $E_k(x, y)$ was the best of the E's of size $2^k \times 2^k$ neighbourhood in four directions at the point (x, y), and the sizes used were 1×1, $2 \times 2, \ldots 2^L \times 2^L$, the "best size" was the largest K such that

$$E_L < \lambda E_{L-1} < \lambda^2 E_{L-2} < \ldots < \lambda^{L-K} E_K \quad \text{but } E_K \geqslant \lambda E_{k-1} \tag{8}$$

where $\lambda = \frac{3}{4}$.

The value at a point was erased if there was a higher value at any point within a distance of half the best size in a direction perpendicular to the best orientation at that point. This approach has the advantage of being able to detect a large variety of edges and also may be able to detect edges of texture regions if the average characteristic feature values (e.g. gray value) over the neighborhoods are different. However, the choices of the different sizes of neighborhood and λ are critical to the success of the algorithm.

Wechsler and Kidode[58] proposed an edge detection algorithm based on finite differences. For each of the four quadrants about the pixel in question, they computed finite differences up to the 3rd order and the central difference. Then the best "edge" element of pixel (x, y) was the one that gave the minimum finite difference. The results were reported to be comparable to those obtained by Sakai[59] and by Kasvand.[60]

(A.3) *Functional approximations*. Edge detection can be considered as an approximation problem. For every point (x', y') in an image, Hueckel[61] used a circular neighbourhood D about (x', y') and asked the question 'Are the intensities (x, y) in D the noisy form of an ideal edge which is characterized by a step function?' Let

$$F(x, y, c, s, p, b, d) = \begin{cases} b & \text{if} \quad cx + sy \le p \\ b + d & \text{if} \quad cx + sy > p \end{cases}, \tag{9}$$

where the x–y co-ordinate system has its origin at the center of the circular region; F is the step function. The task of the operator is to best approximate a given empirical edge element whose gray values are $f(x, y)$ by an ideal edge element characterized by a step function F. As a measure of closeness, E (the square of the Hilbert distances between f and F) was chosen.

$$E = \int_D [f(x, y) - F(x, y, c, s, p, b, d)]^2 \, \mathrm{d}x \, \mathrm{d}y. \tag{10}$$

Hueckel's operator is an efficient solution to the minimization of E. The minimization procedure was facilitated by choosing orthonormal functions (e.g. Fourier functions) over D. The results of the minimization were the best edge and a measure of the goodness of the edge. This technique was later extended to detect lines.[62] Bullock[64] qualitatively evaluated six edge detection techniques applied to the detection of textured edges of outdoor scenes. These six edge detection operators were (1) Robert's cross,[93] (2) high pass filter,[1] (3) Laplacian,[1] (4) Sobel,[57] (5) Kirsch[56] and (6) Hueckel.[61,62] He ranked Hueckel operator first although the complexity of this operator was the highest of these operators. He justified using Hueckel operator in[64] because it performed better than the other five operators particularly on low contrast edges. He ranked the other operators in the following order: second Kirsch; third, Sobel; fourth, Robert's cross, high pass filter, and Laplacian (the last three operators shared the same rank). Fram and Deutsch[65] evaluated three edge detection schemes quantitatively. They found Hueckel's operator performed the poorest. Operators due to Macleod[66] and Rosenfeld *et al.*[94,95] had similar performances. However only two sets of test pictures were used. The first set consisted of five pictures artificially generated with various amounts of nominal contrast. The second set consisted of four pictures of a slanted edge taken from an ERTS photograph with various amounts of artificially generated random noise added to it.

Persoon's[63] operator was defined over a window of size 5×5 pixels and the two columns to the left and to the right of the central one were approximated by linear functions. Deviations from the actual gray levels for the left and right linear function were computed and the right gradient (0°) was defined as a function of the two deviations and the average gray values corresponding to the left and right two columns. The picture was then rotated 7 times through 45° and seven additional gradients were computed. The maximum value of the 8 gradients was taken as an indication of the goodness of the edge which was perpendicular to the direction of the gradient. The technique was applied to rib outlining in chest X-rays. It was reported that this technique gave significantly better results than gradient type operators. This edge detector solves some of the problems related to edge direction and noise but takes more computation time than some simpler edge operators.[55–58]

(B) *Edge element combination (streak or boundary formation)*

(B.1) *Heuristic search and dynamic programming*. Heuristic search is a technique using state space search methods where heuristic information is used to limit the space to be searched. Martelli[67,68] formulated the edge detection problem as a heuristic search for the shortest path on a graph. The graph nodes (or states) were edge elements defined by two neighbouring pixels, e.g. the points $A = (i, j)$, $B = (i, j + 1)$ defined the directed edge element AB. The direction of the edge was obtained with the convention of moving clockwise around the first pixel. He then stated that an edge was a sequence of adjacent edge elements that started in the top row (his arbitrary starting point), ended in the bottom row (his arbitrary ending point), contained no loops and had no element whose direction was "up". So an edge was a path in the graph that represented the state space and the problem of finding the best edge in a picture reduced to the problem of finding an optimal path in the graph. He then embedded properties of edges into an evaluation function and the edge which minimized this function was sought. Only two examples which were artifically generated with various amount of random noise were reported. Some of the drawbacks of this approach are that the algorithm is sequential in nature and the proposed approach does not provide for backtracking, so that once a mistake is made in the midst of the edge the detected edge could be far off from the desired edge. The construction of a proper evaluation function is another problem.

Lester *et al.*[96] applied heuristic search technique and a simpler scheme, the least maximum cost technique, to white blood cell image segmentation. They incorporated both threshold and gradient information in the cost function which guided the search. Both heuristic search and the least maximum cost technique were applied to 50 examples of touching white cells. They reported that heuristic search technique produced more acceptable boundaries than the least maximum cost technique.

Montanari[69] proposed using dynamic programming techniques to perform edge detection. A figure of merit representing the heuristic information was ued to determine the relative value of different paths but was not used to guide the search as in the heuristic search case mentioned above. This figure of merit determined the best path once they had all been enumerated. Montanari discussed finding a smooth, dark curve of fixed length. The curve was embedded in a noisy background, but since the merit function did not guide the search, the computation time was independent of the noise level (which would not be the case if the merit function guided the search as in heuristic search).

The figure of merit of a path $z_1, \ldots z_n$ was defined as

$$h(z_1, \ldots, z_n) = \sum_{i=1}^{n} f(z_i) - q \sum_{i=2}^{n-1} (d(z_{i+1}, z_i) - d(z_i, z_{i-1})) \bmod 8 \quad (11)$$

where $f(z_i)$ is the gray level at $z_i = (x_i, y_i)$, and $d(z_i, z_j)$ is the slope between adjacent points z_i and z_j, so the second term is proportional to curvature. The following constraints were placed on the solution

$$\max(|x_{i+1} - x_i|, |y_{i+1} - y_i|) = 1 \quad (12)$$

$$(d(z_{i+1}, z_i) - d(z_i, z_{i-1})) \bmod 8 \leqslant 1. \quad (13)$$

Montanari then used dynamic programming techniques to arrive at an optimum solution. An artificially generated picture with various amounts of randomly generated noise was processed. Some of the criticisms of heuristic search can also be applied here. The procedure also requires high execution time and large memory.

(B.2) *Relaxation.* Rosenfeld[70] and Riseman *et al.*[54] used a relaxation technique to connect edge elements. The technique is an iterative process where the probability that a canditate edge element is a true edge element is re-estimated at each iteration. Some of the advantages of this approach are that it is a parallel process and it utilizes spatial information. Some of the disadvantages are that the construction of the compatibility function which updates the probabilities of edge elements is not trivial and the convergence rate of the process is often slow.

(B.3) *Line and curve fitting.* Another technique of connecting edge elements together is to fit lines[71] or curves[72] through the edge elements. Duda and Hart[71] proposed an efficient solution to the Hough transform which is an ingenious way of detecting colinear points. Suppose we have a set of n points $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ and we want to find a set of straight lines that fit them. We transform the points (x_i, y_i) into the sinusoidal curves in the θ–ρ plane defined by

$$\rho = x_i \cos\theta + y_i \sin\theta.$$

It is obvious that curves corresponding to colinear points have a common point of intersection. This point in the θ–ρ space, say (θ_0, ρ_0), defines the line passing through the colinear points. The implementation is to quantize the θ–ρ space into an array of cells and plot these sinusoidal curves on this array of cells. The number of curves that pass through every cell in the array is recorded. If the count in a given cell (θ_i, ρ_i) is k, then precisely k figure points lie (to within quantization error) along the line whose normal parameters are (θ_i, ρ_i). The Hough transform concept can be extended to curves.[72,73] Some of the limitations are that the results are sensitive to the quantization of both θ and ρ, and the technique finds colinear points without regard to contiguity. Thus the position of a best-fit line can be distorted by the presence of unrelated points in another part of the picture.

There are other techniques of edge detection such as template matching[1] which can be applied not only in edge detection but in other areas as well, e.g. object extraction.[1] Template matching works well in a very constrained environment but fails where there is great variation of the patterns to be matched.

2.2 *Sequential techniques*

In sequential edge detection, the result at a point is contingent upon the results of the operator at previously examined points. The major components of a sequential edge detection procedure are:

(1) The picking of a good initial point: the performance of the entire procedure will depend upon the choice of a good starting point.

(2) The dependence structure: how do the results obtained at previously examined points affect both the choice of the next point to be examined and the result at the next point?

(3) A termination criterion: there must be a way for the procedure to determine that it is finished.

There are a number of sequential techniques such as using heuristic search and dynamic programming (both are discussed in B.1 of Section 2.1). A guided search technique was used by Kelly.[75] Chien and Fu[74] used guided search with an evaluation function to detect cardiac and lung boundaries in chest X-ray images.

2.3 *Comments on the edge detection approach to image segmentation.*

The problems with edge detection techniques are that sometimes edges are detected which are not the transition from one region to another and detected edges often have gaps in them at places where the transitions between regions are not abrupt enough. So detected edges may not necessarily form a set of closed connected curves that surround connected regions. As far as the applicability of edge detection techniques to cell image segmentation is concerned, for example, there are more than six types of white blood cell and each type has its own characteristic color and texture of the nucleus and cytoplasm. For neutrophils, the shape of the nucleus varies widely. For basophils, there are heavy cytoplasm granules that are on top of the nucleus. So it is very difficult to connect detected edge elements to form closed curves because the information used to connect these edge elements may vary for different cell types. Besides, it is computationally very expensive. Prewitt[55] applied an edge detection technique to neutrophil segmentation. However, her technique has not been proven to be robust.

3. REGION EXTRACTION

Another way of doing image segmentation instead of finding boundaries of regions, is to divide the image into regions. Zucker[16] wrote an excellent survey on region extraction methods. Region extraction techniques can be broken down into three categories, (1) region merging, (2) region dividing, and (3) a combination of region merging and dividing.

3.1 *Region merging*

Muerle and Allen[76] used regional neighbour search method to merge regions of similar properties. Brice and Fennema[77] formed connected components of equal intensity, refined with phagocyte and weakness heuristics. Pavlidis[90] partitioned the image into a collection of one-dimensional strips, divided the strips into segments and merged the segments with similar approximation coefficients. Feldman and Yakimovsky[78,84] used semantics to do region merging. They tried to maximize the probability that all regions and boarders were correctly interpreted. Rosenfeld *et al.*[88] used a relaxation approach, also called iterative probabilistic process to do scene labelling.

Tenenbaum and Barrow[89] proposed IGS (Interpretation Guided Segmentation) as an approach to region merging. The program iteratively processed the scene until its components were semantically consistent. For example: a picture hung on a wall, a telephone on top of a desk. Gupta and Wintz[79,80] used a minimum distance classifier which interpreted each initial region as belonging to one of a small predetermined number of different classes such as corn, soybean, forest, water, etc. Neighboring regions were merged based on their class membership. Jarvis[91] used a shared near neighbour clustering technique to do region merging. Tsuji and Fujiwara[83] used linguistic techniques to perform region merging.

3.2 *Region dividing*

One way of doing image segmentation by region extraction is the region dividing approach. Robertson *et al.*[85] used a mean vector of gray levels of multispectral image to perform region dividing. Klinger[86,87] proposed to use regular decomposition for image segmentation.

3.3 *Region merging and dividing*

Horowitz and Pavlidis[81,82] approached the problem using a "split and merge" principle. Regions were described in terms of an approximating function. They merged adjacent regions having similar approximations and split those regions that had large approximating errors.

3.4 *Comments on the region extraction approach to image segmentation*

One of the disadvantages of the region merging processes is their inherently sequential nature. The regions produced depend on the order in which regions are merged together. Almost all region extraction algorithms use local information heavily. There is no simple way to incorporate global information into the model unless we severely restrict the class of pictures we are dealing with. All of the region extraction techniques process the pictures in an iterative manner and usually involve a great expenditure in computation time and memory. Thus, they are not suitable for application in cell image segmentation. Up to the present moment, there is no published information on successfully applying region extraction techniques to a practical cytology pattern recognition system.

CONCLUSIONS

This paper surveys various existing approaches to image segmentation. So far, image segmentation techniques are strongly application dependent. For example, edge detection techniques are favoured by most researchers in chest X-ray image segmentation whereas thresholding and clustering techniques are widely used by researchers in cell image segmentation. Semantic and *a priori* information about the type of images are critical to the solution of the segmentation problem. One of the fruitful areas of reaearch is to combine spatial and semantic information with edge detection and thresholding or clustering techniques to perform image segmentation.

REFERENCES

1. A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, Academic Press, New York (1976).

2. A. Rosenfeld, Picture processing by computer, *Computing Surveys* **1,** 147–176 (1969).
3. A. Rosenfeld, Progress in picture processsing: 1969–71, *Computings Surveys* **5,** 81–108 (1973).
4. A. Rosenfeld, Picture processing: 1972 *Computer Graphics and Image Processing,* **1,** 394–416 (1972).
5. A. Rosenfeld, Picture processing: 1973, *Computer Graphics and Image Processing,* **3,** 178–194 (1974).
6. A. Rosenfeld, Picture processing: 1974, *Computer Graphics and Image Processing,* **4,** 133–155 (1975).
7. A. Rosenfeld, Picture processing: 1975, *Computer Graphics and Image Processing,* **5,** 215–237 (1976).
8. A. Rosenfeld, Picture processing: 1976, *Computer Graphics and Image Processing,* **6,** 157–183 (1977).
9. A. Rosenfeld, Picture processing: 1977, *Computer Graphics and Image Processing,* **7,** 211–242 (1978).
10. K. P. Kruger, W. B. Thompson and A. F. Turner, Computer diagnosis of pneumoconiosis, *IEEE Trans. Systems, Man Cybernet.* **SMC-4,** 40–49 (1974).
11. J. R. Green, Parallel processing in a pattern recognition based image processing systems: The Abbott ADC-500 differential counter, *Proc. IEEE Conf. Pattern Recognition and Image Processing,* 31 May–2 June, Chicago, pp. 492–498 (1978).
12. P. H. Bartels and G. L. Wied, High resolution prescreening systems for cervical cancer, *The Automation of Uterine Cancer Cytology,* p. 144. Tutorials of Cytology, Chicago (1976).
13. K. R. Castleman and J. H. Melnyk, An automated system for chromosome analysis, Final report Jet Propulsion Laboratory 5040-30 Pasadena, 4 July (1976).
14. S. L. Horowitz and T. Pavlidis, Picture segmentation by a directed split-and-merge procedure, *Proc. 2nd Int. Joint Conf. Pattern Recognition,* pp. 424–433. (1974).
15. T. Pavlidis, *Structural Pattern Recognition.* Springer, New York (1977).
16. S. W. Zucker, Region growing: childhood and adolescence, *Computer Graphics and Image Processing,* **5,** 382–399 (1976).
17. W. A. Yasnoff, J. K. Mui and J. W. Bacus, Error measures for scene segmentation, *Pattern Recognition* **9,** 217–231 (1977).
18. J. S. Weszka, A survey of threshold selection techniques, *Computer Graphics and Image Processing* **7,** 259–265 (1978).
19. W. Doyle, Operations useful for similarity-invariant pattern recognition, *J. Ass. comput. Mach.* **9,** 259–267 (1962).
20. J. M. S. Prewitt and M. L. Mendelsohn, The analysis of cell images, *Trans. N.Y. Acad. Sci.* **128,** 1035–1053 (1966).
21. J. S. Weszka, R. N. Nagel and A. Rosenfeld, A threshold selection technique *IEEE Trans. Comput.* **C-23,** 1322–1326 (1974).
22. J. S. Weszka and A. Rosenfeld, Threshold Selection Techniques 5, University of Maryland, Computer Science Center, TR-349, Feb. (1975).
23. A. Rosenfeld and L. S. Davis, Iterative histogram modification, University of Maryland, Computer Science Center, TR-519, April (1977).
24. S. Watanabe *et al.*, An automated apparatus for cancer prescreening: CYBEST, *Computer Graphics and Image Processing* **3,** 350–358 (1974).
25. J. S. Weszka, J. A. Verson and A. Rosenfeld, Threshold selection techniques 2, University of Maryland Computer Science Center, TR-260, August (1973).
26. R. K. Aggarwal and J. W. Bacus, A multi-spectral approach for scene analysis of cervical cytology smears *J. Histochem. Cytochem.* **25,** 668–680 (1977).
27. L. S. Davis, A. Rosenfeld and J. S. Weszka, Region extraction by averaging and thresholding *IEEE Trans. Systems, Man Cybernet.* **5,** 383–388 (1975).
28. S. W. Zucker, A. Rosenfeld and L. S. Davis, Picture segmentation by texture discrimination, *IEEE Trans. Comput.* **24,** 1228–1233 (1975).
29. R. B. Ohlander, Analysis of natural scenes, Ph.D. dissertation, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, April (1975).
30. F. Tomita, M. Yachida and S. Tsuji, Detection of homogeneous regions by structural analysis, *Proc. 3rd IJCAI,* pp. 564–571 (1973).
31. S. Tsuji and F. Tomita, A structural analyzer for a class of textures *Computer Graphics and Image Processing* **2,** 216–231 (1973).
32. J. Keng and K. S. Fu, A syntax-directed method for land-use classification of LANDSAT images, *Proc. Symp. Current Mathematical Problems in Image Science,* Moneterey, CA, pp. 261–265, 10–12 November (1976).
33. I. T. Young and I. L. Paskowitz, Localization of cellular structures, *IEEE Trans. biomed. Engng* **BME-22,** 35–40 (1975).
34. M. Ingram and K. Preston Jr., Automatic analysis of blood cells, *Sci. Am.* **223,** 72–82 (1970).
35. J. F. Brenner *et al.*, Scene segmentation techniques for the analysis of routine bone marrow smears from acute lymphoblastic leukemia patients *J. Histochem. Cytochem.* **25,** 601–613 (1977).
36. A. Rosenfeld and J. S. Weszka, Picture recognition and scene analysis, *Computer* 23–38 (1976).
37. J. Keng, Syntactic algorithms for image segmentation and a special computer architecture for image processing, Ph.D. thesis, Purdue University, West Lafayette, Indiana, December (1977).
38. A. G. Wacker, A cluster approach to finding spatial boundaries in multispectral imagery, Laboratory for Applications of Remote Sensing Information Note 122969, Purdue University (1969).
39. P. H. Swain and K. S. Fu, On the applications of non-parametric techniques to crop classification problems *National Electronics Conf. Proc.* **24,** 14–19 (1968).
40. K. S. Fu, Y. P. Chien and E. Persoon, Computer systems for the analysis of chest X-ray's, *Proc. EASCON '75,* Washington, D.C., pp. 72-A–72-P (1975).
41. H. Wechsler and J. Sklansky, Finding the rib cage in chest radiographs, *Pattern Recognition,* **9,** 21–30 (1976).
42. L. S. Davis A survey of edge detection techniques, *Computer Graphics and Image Processing* **4,** 248–270 (1975).
43. B. J. Schachter, L. S. Davis and A. Rosenfeld, Scene segmentation by cluster detection in color space, TR-424, Computer Science Center, University of Maryland, Nov. (1975).
44. B. J. Schachter, L. S. Davis and A. Rosenfeld, Some experiments in image segmentation by clustering of local feature values, TR-510, Computer Science Center, University of Maryland, March (1977).
45. M. Goldberg and S. Shlien, A cluster scheme for multispectral images *IEEE Trans. Systems, Man Cybernet.* **SMC-8,** 86–92 (1978).
46. S. G. Carlton and O. R. Mitchell, Image segmentation using texture and gray level, *Proc. IEEE Conf. Pattern Recognition and Image Processing,* Troy, New York, pp. 387–391, 6–8 June (1977).
47. T. Ito, Pattern classification by color effect method, *Proc 3rd Int. Joint Conf. on Pattern Recognition,* Coronado, pp. 26–30 (1976).
48. T. Ito and M. Fukushima, Computer analysis of color information with applications to picture processing, *Proc. Int. Joint Conf. Pattern Recognition,* Coronado, pp. 833–837 (1976).
49. G. B. Coleman, Image segmentation by clustering, Report 750, University of Southern California Image Processing Institute, July (1977).
50. J. R. Yoo and T. S. Huang, Image segmentatiqn by unsupervised clustering and its applications, TR-EE

78-19, Purdue University, West Lafayette, Indiana (1978).

51. R. M. Haralick and I. Dinstein, A spatial clustering procedure for multi-image data, *IEEE Trans. Circuits Systems* **CAS-22**, 440–450 (1975).
52. J. K. Mui, J. W. Bacus and K. S. Fu, A scene segmentation technique for microscopic cell images J. Sklansky (Ed.) *Proc. Symp. Computer Aided Diagnosis of Medical Images*, San Diego, CA, *IEEE Publ.* No. 76 CH1170-OC, pp. 99–106 (1976).
53. P. Nagin, A. Hanson and E. Riseman, Relaxation-based segmentation based on spatial context and feature space cluster labels, *Proc. IEEE Conf. Pattern Recognition and Image Processing*, Chicago, Illinois, p. 421, May (1978).
54. E. M. Riseman and M. A. Arbib, Computational techniques in the visual segmentation of static scenes, *Computer Graphics and Image Processing* **6**, 221–276 (1977).
55. J. M. S. Prewitt, Object enhancement and extraction. In *Picture Processing and Psychopictorics*, B. S. Lipkin and A. Rosenfeld (Eds.), pp. 75–149, Academic Press, New York (1970).
56. R. Kirsch, Computer determination of the constituent structure of biological images, *Computer biomed. Res.* **4**, 315–328. (1971).
57. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis.* Wiley, New York (1973).
58. H. Wechsler and M. Kikode, A new edge detection technique and its implementation, *IEEE Trans Systems, Man Cybernet.* **SMC-7**, 827–836 (1977).
59. T. Sakai, M.. Nagao and M. Kidode, Processing of multilevel pictures by computer—the case of photographs of human face, *Systems, Computers Controls* **2**, (3) 47–54 (1971).
60. T. Kasvand, Iterative edge detection, *Computer Graphics and Image Processing* **4**, 279–286 (1975).
61. M. Hueckel, An operator which locates edges in digital pictures *J. Ass. comput. Mach.* **18**, 113–125 (1971).
62. M. Hueckel, A local operator which recognizes edges and lines, *J. Ass. comput. Mach.* **20**, 634–647 (1973).
63. E. Persoon, A new edge detection algorithm and its applications in picture processing, TR-EE 75-38 Purdue University, W. Lafayette, Indiana, Oct. (1975).
64. B. L. Bullock, Finding structure in outdoor scenes. In *Pattern Recognition and Artificial Intelligence*, C. H. Chen (Ed.), Academic Press, New York (1976).
65. J. R. Fram and E. S. Deutsch, On the quantitative evaluation of edge detection schemes and their comparison with human performance. *IEEE Trans. Comput.* **C-24**, 616–628 (1975).
66. I. D. G. Macleod, On finding structures in pictures. In *Picture Language Machines*, S. Kaneff (Ed.), Academic Press, New York (1970).
67. A. Martelli, Edge detection using heuristic search methods. *Computer Graphics and Image Processing* **1**, 169–182 (1972).
68. A. Martelli, An application of heuristic search methods to edge and contour detection, *Communs ACM* **19**, 73–83 (1976).
69. U. Montanari, On the optimal detection of curves in noisy pictures. *Communs ACM* **14**, 335–345 (1971).
70. A. Rosenfeld, Iterative methods in image analysis, *Proc. IEEE Conf. Pattern Recognition and Image Processing*. Troy, New York, pp. 14–20, 6–8 June (1977).
71. R. O. Duda and P. E. Hart, Use of the Hough transformation to detect lines and curves in pictures, *Communs ACM* **15**, 11–15 (1972).
72. S. D. Shapiro, Transformations for the computer detection of curves in noisy pictures, *Computer Graphics and Image Processing*, **4**, 328–338 (1975).
73. C. Kimme, D. H. Ballard and J. Sklansky, Finding circles by an array of accumulators, *Communs ACM* **18**, 120–122 (1975).
74. Y. P. Chien and K. S. Fu, Preprocessing and feature extraction of picture patterns, TR-EE 74-20, Purdue University, West Lafayette, Indiana (1974).
75. M. Kelly, Edge detection by computer using planning. In *Machine Intelligence* VI, pp. 397–409, Edinburgh University Press, Edinburgh (1971).
76. J. L. Muerle and D. C. Allen, Experimental evaluation of techniques for automatic segmentation of objects in a complex scene. In *Pictorial Pattern Recognition*, C. C. Cheng *et al.* (Eds.), pp. 3–13, Thompson, Washington, (1968).
77. C. Brice and C. Fennema, Scene analysis using regions, *Artificial Intell.* **1**, 205–226 (1970).
78. J. A. Feldman and Y. Yakimovsky, Decision theory and artificial intelligence, I. A semantics based region analyzer *Artificial Intell.* **5**, 349–371 (1974).
79. J. N. Gupta and P. A. Wintz, Computer processing algorithm for locating boundaries in digital pictures, *Proc. Int. Joint Conf. Pattern Recognition*, pp. 155–156 (1974).
80. J. N. Gupta and P. A. Wintz, Multi-image modeling, Technical Report TR-EE 74-24, School of Electrical Engineering, Purdue University, September (1974).
81. S. L. Horowitz and T. Pavlidis, Picture segmentation by a directed split-and-merge procedure, *Proc. Int. Joint Conf. Pattern Recognition*, pp. 424–433 (1974).
82. S. L. Horowitz and T. Pavlidis, Picture segmentation by a tree traversal algorithm, *J. Ass. comput. Mach.* 368–388 (1976).
83. S. Tsuji and R. Fujiwara, Linguistic segmentation of scenes into regions, *Proc. 2nd Int. Joint Conf. Pattern Recognition*, pp. 104–108 (1974).
84. Y. Yakimovsky and J. Feldman, A semantics-based decision theory region analyzer, *Proc. Int. Joint Conf. Artificial Intelligence*, Stanford, California, pp. 580–588, August (1973).
85. T. V. Robertson, K. S. Fu and P. H. Swain, Multispectral image partitioning, Technical Report TR-EE73-26, School of Electrical Engineering, Purdue University, W. Lafayette, Indiana, August (1973).
86. A. Klinger, Data structures and pattern recognition, *Proc. 1st Int. Joint Conf. Pattern Recognition*, pp. 497–498 (1973).
87. A. Klinger and C. R. Dyer, Experiments on picture representation using regular decomposition, *Computer Graphics and Image Processing* **4**, 68–105 (1975).
88. A. Rosenfeld, R. Hummel and S. W. Zucker, Scene labeling by relaxation operations, *IEEE Trans. Systems, Man, Cybernet.* **SMC-6**, 420–433 (1976).
89. J. M. Tenenbaum and H. G. Barrow, IGS: a paradigm for integrating image segmentation and interpretation *Proc. 3rd Int. Joint Conf. Pattern Recognition*, pp. 504–513, November (1976).
90. T. Pavlidis, Segmentation of pictures and maps through functional approximation, *Computer Graphics and Image Processing*. **1**, 360–372 (1972).
91. R. A. Jarvis, Region based image segmentation using shared near neighbor clustering, Seventh International Conference on Cybernetics and Society, Washington, D.C., 19–21 September (1977).
92. W. Koontz, P. M. Narendra and K. Fukunaga, A graph theoretic approach to nonparametric cluster analysis, *IEEE Trans. Comput.* **C-25**, 936–944 (1976).
93. L. G. Roberts, Machine perception of three dimensional solids. In *Optical and Electro-Optical Information Processing*. J. Tippett, D. Berkowitz, L. Clapp, C. Koester, A. Vanderburg (Eds), pp. 159–197. M.I.T. Press (1965).
94. A. Rosenfeld and M. Thurston, Edge and curve detection for visual scene analysis, *IEEE Trans. Comput.* **C-20**, 562–569 (1971).

95. A. Rosenfeld, M. Thurston and Y. Lee, Edge and curve detection further experiments, *IEEE Trans. Comput.* **C-21**, 677–715 (1972).
96. L. M. Lester, H. A. Williams, B. A. Weintraub and J. F. Brenner, Two graph searching techniques for boundary finding in white blood cell images, *Comput. Biol. Med.* **8**, 293–308 (1978).
97. R. L. Cahn, R. S. Poulsen and G. Toussaint, Segmentation of cervical cell images, *J. Histochem. Cytochem.* **25**, 681–688 (1977).

About the Author—KING-SUN FU received the Ph.D. degree in Electrical Engineering from the University of Illinois, Urbana, in 1959. He is presently Goss Distinguished Professor of Engineering and Professor of Electrical Engineering at Purdue University.

He is a Fellow of the Institute of Electrical and Electronic Engineers, a member of the National Academy of Engineering and Academia Sinica, and a Guggenheim Fellow. He received the Herbert N. McCoy Award for contributions to Science in 1976 and the Outstanding Paper Award of the IEEE Computer Society in 1977. He is the Senior Editor of *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and an Associate Editor of *IEEE Transactions on Systems, Man and Cybernetics, Pattern Recognition, Journal of Cybernetics*, and *Information Sciences*, a member of the editorial board of the *International Journal of Computer and Information Sciences*, and a member of the editorial advisory committee of the *Journal of Information Processing*. He is presently the chairman of the Executive Committee of International Association for Pattern Recognition (IAPR). He is the author of the books, *Sequential Methods in Pattern Recognition* and *Syntactic Methods in Pattern Recognition*, published by Academic Press in 1968 and 1974, respectively.

About the Author—JACK KIN YEE MUI was born in Toyshan, Kwangtung, China on February 1, 1952. He received the B.S. degree with highest distinction and M.S. degree, both in electrical engineering from Northwestern University, Evanston, Illinois in 1972 and 1974 respectively.

During 1972–1973, he was the recipient of Walter J. Murphy Fellowship. During 1973–1974, he was a teaching assistant at the department of electrical engineering and computer science at Northwestern University. Since August 1974, he has been a graduate research assistant in the Advanced Automation Research Laboratory at Purdue University. From May 1975 to April 1977, he was on leave to perform research on automated blood cell classification at the Medical Automation Research Unit of Rush Presbyterian St. Luke's Medical Center in Chicago, Illinois. His research interests include image processing, pattern recognition, computer graphics and software engineering. Since December 1978 he has been with Bell Telephone Laboratories, Naperville, Illinois.

Mr. Mui is a member of Eta Kappa Nu, Phi Kappa Phi, Tau Beta Pi, IEEE, ACM and the Pattern Recognition Society.