AD-A070 928 MASSACHUSETTS UNIV AMHERST DEPT OF COMPUTER AND INF--ETC F/G 9/2
AN IMAGE SEGMENTATION SYSTEM BASED ON THRESHOLDING.(U)
DEC 78 R KOHLER N00014-75-C-0459
UNCLASSIFIED COINS-TR-78-23 NL

END
DATE FILMED
8-79
DDC

LEVEL II

12

AD A070928

Computer and Information Science

University of Massachusetts at Amherst

Computers

Theory of Computation

Cybernetics

LEVEL II

12

See 1473 in back.

AN IMAGE SEGMENTATION SYSTEM BASED ON THRESHOLDING

Ralf Kohler

COINS Technical Report 78-23

December 1978

This research has been supported by the Office of Naval Research under Grant N00014-75-C-0459.

Abstract.

The segmentation algorithm proposed in this paper is a complex form of thresholding which utilizes multiple thresholds. The algorithm consists of two major components: a threshold selection component and a relaxation component.

The threshold selection component automatically selects a threshold so as to maximize the global average contrast of edges detected by the threshold across the image. This algorithm for threshold selection compares favorably with other methods for automatic threshold selection. The threshold selection algorithm can be applied recursively to select additional thresholds by ignoring any edges which have already been detected by previously selected thresholds.

The relaxation component utilizes the immediate spatial context of each pixel to update both the label at the pixel and the feature measurement at the pixel. The update function proposes a new feature value at the pixel defined by a weighted average of the central pixel and all of its neighbors. The weight associated with each pixel (with respect to the pixel being updated) is proportional to the spatial distance between the pixels, the probability that the two pixels are correctly labeled, and the probability that the two pixels belong to the same region. The update function then replaces the feature value at the pixel with a value somewhere between the current value and the proposed value. When the local evidence for shifting the feature value is consistent then the value adopted will be close to the proposed value; however, when the local evidence is inconsistent the value adopted will be close to the original value.

The relaxation is independently performed for each threshold selected. The resulting binary images are intersected to produce the final segmentation. This algorithm works well not only for simple images but also produces reasonable segmentations for complex images.

A SEGMENTATION ALGORITHM BASED ON THRESHOLDING

1.0 Introduction.

This paper develops a segmentation algorithm based on multiple thresholds. An image segmentation algorithm partitions an image into disjoint sets of spatially contiguous pixels (referred to as regions). The goal of image segmentation algorithms is to produce segmentations for which there is a high corellation between the entities of the real world (objects, surfaces, and parts of objects) and the regions of the segmentation. Existing image segmentation algorithms can be divided into two broad classes. The first class attempts to build regions in the image based on the similarities of some characteristics (or features) of the picture elements (pixels) in the image. The second class of algorithms attempts to locate those edges in the image which correspond to object or surface discontinuities [fn. 1] based on differences between pixel characteristics.

The segmentation algorithm described here does not strictly lie in either of these two classes. The algorithm attempts to utilize both pixel similarity and pixel difference information to arrive at a meaningful

---

[1] Henceforth, both object and surface boundaries will be referred to as object boundaries. While there are particular considerations relevant to the detection of surface boundaries, they are not the focus of this treatment.

segmentation. It is hoped that the quality of segmentations based on both types of knowledge will be superior to segmentaions based only on pixel similarities or pixel differences.

The basic mechanism used to generate the segmentation is a complex form of thresholding. Let us consider some of the issues in the simple thresholding of images. The image is partitioned by assigning one label to pixels with feature values which are above some threshold T, and another label for pixels with feature values not above T. For some images, such as chromosome images or hand printed characters, where a clear foreground-background (figure-ground) relationship exists, a single threshold will be able to detect all or most of the object boundaries at the object-background discontinuity. However, those boundaries which correspond to object-object discontinuities, or internal structure of the object may not be detected by that threshold. Furthermore, in more complex images which do not exhibit a clear foreground-background distinction (such as images of natural outdoor scenes), one cannot expect a single threshold to detect all or even most of the object boundaries in the scene. In order to detect most of the interesting boundaries within an image, our

segmentation algorithm will be defined in terms of a set of n thresholds rather than a single threshold. These n thresholds will partition the feature space into n+1 possible classes.

Section 2.0 below classifies the possible segmentation errors assuming a 'correct' segmentation is known. These error classes are an oversimplification since there are no accepted criteria defining a 'correct' segmentation; however, they aid in the analysis of the segmention processes.

Figure 1 shows a block diagram of the structure of the proposed segmentation algorithm. The algorithm selects a set of n thresholds, independently applies a relaxation correction procedure to each binary segmentation defined by the set of thresholds, and combines the resulting n segmentations into a single segmentation. The threshold selection component automatically selects a set of thresholds so as to maximize the global average contrast of edges detected by the threshold across the image. The development of the algorithm and a comparison to other threshold selection algorithms is presented in sections 3.0 to 3.5 below. The thresholder in figure 1 simply generates a binary labeling of the image pixels for

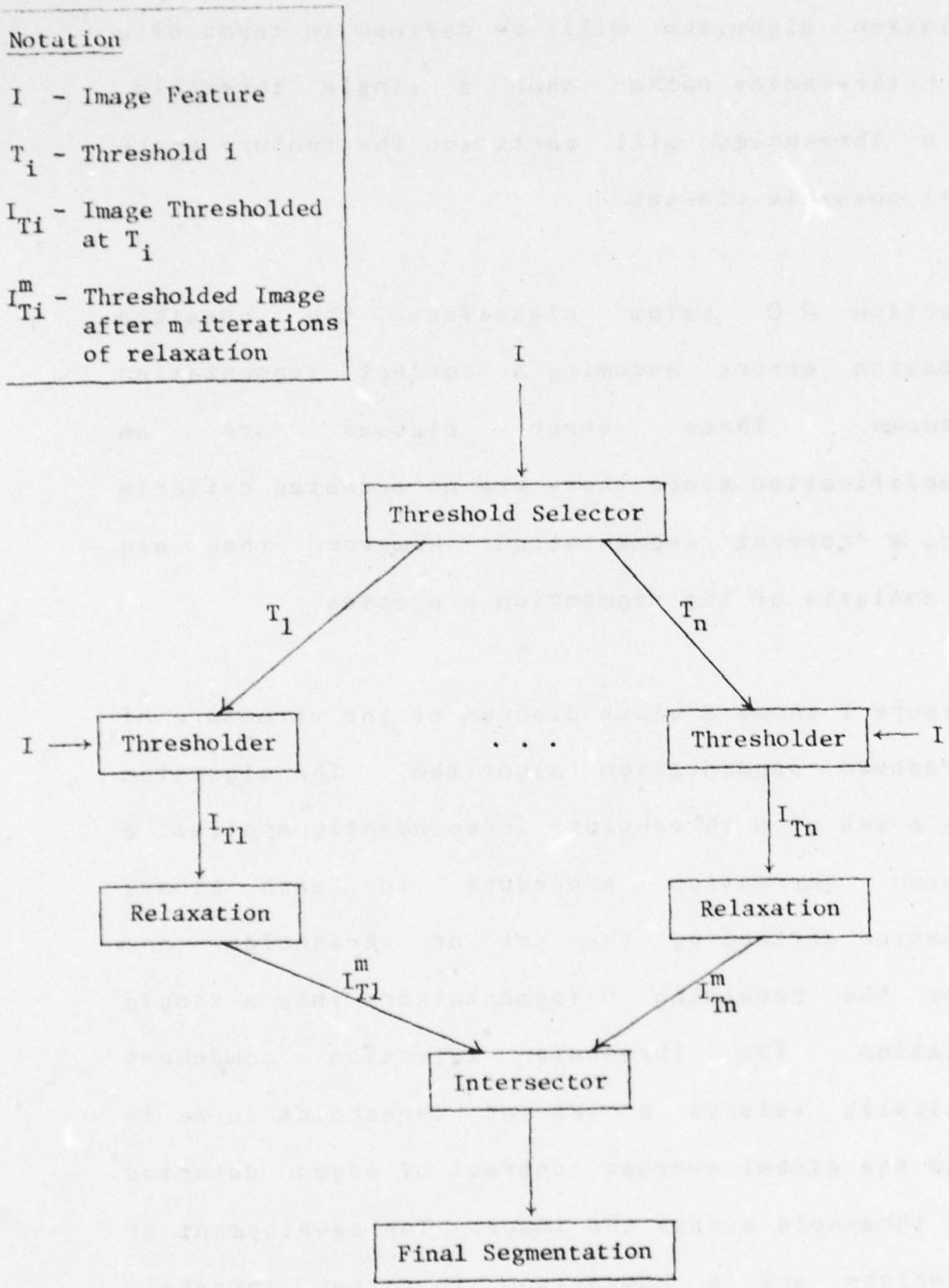


Figure 1. The Segmentation Algorithm.

each selected threshold. The relaxation process utilizes the immediate spatial context of each pixel to update both the label and the feature measurement at the pixel. The relaxation process can be viewed as an interpolation between the central pixel and the set of pixels in the spatial context of the central pixel. The development of the relaxation process is described in detail in sections 4.0 to 4.7 below. To obtain the final segmentation, the intersector process combines the set of binary segmentations by simply overlaying them and defining a new region label for each distinct combination of n labels in the set of binary images.

2.0 Segmentation Errors.

Given any resultant segmentation and a corresponding 'correct' segmentation [fn. 2] of an image, one can distinguish two primary types of errors. The segmentation can contain boundaries which do not exist in the 'correct' goal segmentation and, therefore, do not correspond to any real object discontinuity in the image (i.e. a false alarm). Alternatively, the segmentation can miss edges which

---

[2] Note that there are no well defined criteria for 'correct' segmentation of an image. Furthermore, the definition of a 'correct' segmentation is inherently ambiguous since the 'correct' segmentation is a function of the goals of the segmentation system.

appear in the 'correct' segmentation. These will be referred to as type one and type two errors respectively. Examples of these errors are provided in figure 2. This figure also shows a third type of error which can be viewed as a compound type one and type two error. For this type of error the boundary in the 'correct' segmentation is correctly detected by the segmentation algorithm, but not in the proper location. These errors are often due to inaccuracies in the transformation between the spatially continuous image and its discrete representation. It should be much easier to recover from these errors than from type one or type two errors.

3.0 Threshold Selection.

The threshold selection process should select that threshold which minimizes some measure of the expected segmentation errors. The proposed threshold selection algorithm is based on estimating the expected number of type one and type two errors using either edge information or pixel feature differences. This algorithm will be compared to several other algorithms, most of which attempt to minimize the number of pixels misclassified using pixel similarity information. Since the edge and region approaches are based on totally different models of image information, it is

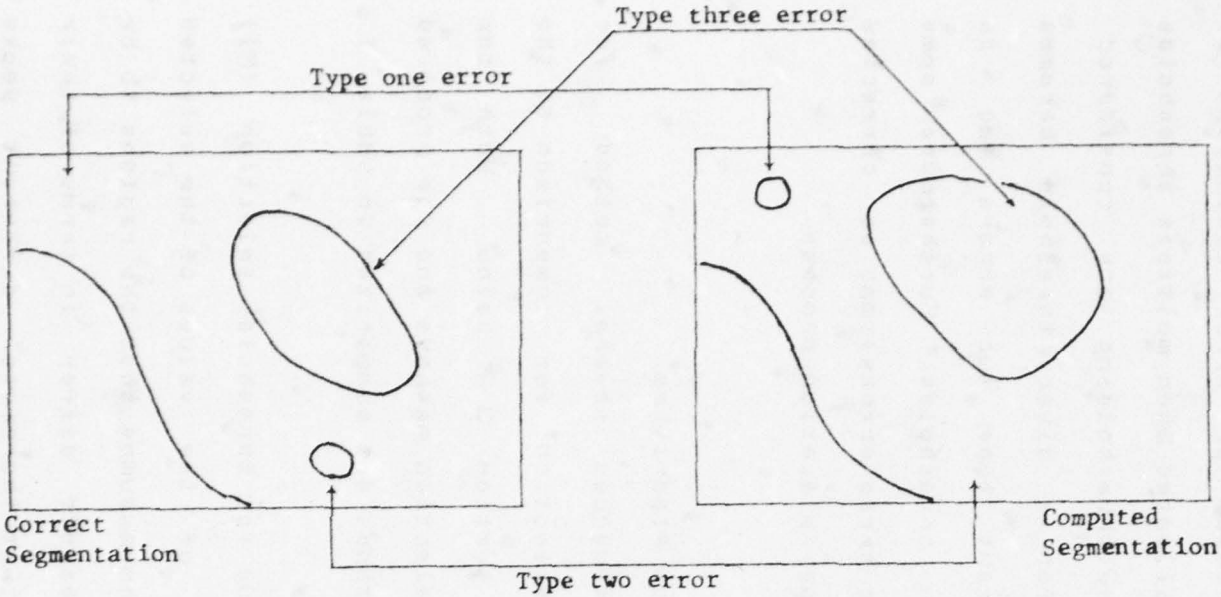


Figure 2. Categorization of Segmentation Error Types.

Type one error - Inserted edges

Type two error - Missing edges

Type three error - Incorrectly placed edges

argued that the range of images to which automatic threshold selection is applicable is extended.

Note that the problem of error estimation is considerably more complicated when multiple thresholds and processing after the thresholding are considered. Estimating errors for a given threshold becomes difficult since apparent type two errors may be detected by secondary thresholds. Furthermore, some errors (especially type three errors) may be corrected by the post-thresholding relaxation process.

3.1 Threshold Selection Algorithms.

This section introduces several methods for automatic threshold selection for comparison to the methods proposed in section 3.2 below. Both the comparison threshold selection methods and the proposed threshold selection methods are summarized in table 1.

One standard method for threshold selection (M1) utilizes a histogram of the values of the selected features. The algorithm assumes that the regions to be detected by the threshold differ in terms of their feature activity, and that therefore, different peaks in the feature histogram correspond to different image regions. This assumption is often violated, especially

Table 1. Summary of Threshold Selection Algorithms.

M1. Standard Feature Histogram Method.

This method selects a valley between two peaks in the feature histogram.

M2. Gradient Weighted Feature Histogram Method.

This method is identical to method M1 except that the contribution of the pixel to the feature histogram is weighted toward pixels of low gradient.

M3. High Gradient Pixel Histogram Method.

This method is identical to method M1 except that only image pixels exhibiting high gradient are considered in the histogram.

M4. Total Gradient Histogram Method.

This method selects the threshold at the largest peak in a total-gradient histogram. The histogram is computed as for method M1 except that the contribution of a pixel to the feature histogram is directly proportional to the gradient magnitude at the pixel.

M5. High Gradient Pixel Average Method.

This Method computes the threshold by simply averaging the feature values of all high gradient pixels.

M6. Uniform Average Contrast Method.

This method selects the threshold which generates the highest average contrast of detected edges over the image using the expected contrast histogram.

M7. Relative Contrast Method.

This method is identical to method M6 except that the contrast of detected edges used in computing the histogram is defined relative to the threshold rather than using the simple edge strength as the measure of contrast.

when high levels of texture are present in the image. Nevertheless, let us consider the case for which the assumption is valid and the peaks in the feature histogram correspond to overlapping normal distributions of feature activity. With two normal distributions the Bayesian minimum error decision would place the decision boundary (i.e. the threshold) at the minimum between the histogram peaks. The correct application of such statistical methods is virtually impossible in practice due to the difficulty in estimating the underlying distributions from the histogram; especially when the number of underlying distributions is unknown or when the types of the underlying distributions are unknown. Therefore, the simple heuristic of threshold placement at the minimum between histogram peaks is typically adopted [fn. 3]. This method of threshold selection is referred to as the standard histogram method. There are two difficulties in threshold selection using this method:

a) the valleys between histogram peaks are long and flat making threshold selection difficult; and

b) the edge information in the image is not utilized.

---

[3] Although automatic methods which precisely define this selection criterium have been developed [NAG78], the results presented below are based on local minima which were manually selected by the author.

In order to overcome these difficulties, Weszka [WES74] proposed adding gradient information into the histogram by reducing the relative weight of histogram entries which exhibit high gradient magnitudes. It was hypothesized that pixels at edges, where the gradient magnitude is large, have feature activity which is between the peaks of feature activity associated with the regions bounded by the edge. It was hoped that the lower weighting of these pixels would sharpen the peaks and valleys of the histogram. This method is referred to as the gradient-weighted histogram method (M2).

Another method proposed by Weszka [WES75], referred to as the high-gradient histogram method (M3), is based on a complementary point of view. It considers only pixels of high gradient magnitude in the computation of the histogram. Presumably, these pixels are more critical in the selection of the threshold since they bound an edge which should be detected by the threshold selected. This method is similar to the methods of Katz [KAT65] which also utilizes only the high gradient pixels. The Katz method computes the threshold simply by averaging all of the high-gradient pixels and is the only method discussed which does not utillize a histogram. The Katz method (M5) is referred to as the high-gradient pixel average method.

Another method for selecting a threshold utillizing gradient information was proposed by Watanabe [WAT74]. This method selects the threshold at that intensity value for which the total gradient magnitude is largest and is therefore referred to as the total gradient histogram method (M4). This method and the modification proposed by Weszka [WES73] represent the conceptual starting point for the threshold selection method proposed below. The Watanabe method estimates the expected total contrast of all edges detected by each threshold and the Weszka method normalizes for the number of edges detected. This is the only method discussed so far which selects a threshold at a peak in the histogram rather than at a valley. Note that all of these methods, except the standard histogram method, attempt to take gradient information into consideration.

3.2 The Proposed Algorithm.

The threshold selection method proposed in this paper attempts to incorporate the gradient information much more directly. Unlike the previous methods, the algorithm does not directly depend on the assumption that regions generate peaks in the histogram of feature values. Instead, the algorithm is based on the obvious heuristic that edges which correspond to real region

discontinuities in the image tend to have high contrast, while edges of low contrast usually do not correspond to real region boundaries. This is the heuristic on which most of the edge segmentation algorithms are based. Clearly, this heuristic is not valid in some cases -- especially when high contrast texture is present -- but we will not be concerned with these cases in this treatment. This heuristic, translated into a threshold selection mechanism, might be stated as follows:

> The optimum threshold for segmentation of the image is that threshold which detects more high contrast edges and fewer low contrast edges than any other threshold.

Under the assumption above, detecting more high contrast edges implies a reduction of type two errors (missing edges), while not detecting low contrast edges implies a reduction of type one errors (inserted edges).

One possible, easily computed, function which serves as a measure of the criterion stated above is the average contrast of all edges detected by a particular threshold. The more low contrast edges a threshold detects, the smaller the average contrast becomes. The more high contrast edges a threshold

detects, the larger the average contrast becomes. If one creates a histogram of the average contrast for each possible threshold then the highest peak in this histogram corresponds to the optimum threshold.

The use of the average-contrast-histogram was first proposed by Weszka, Vernon, and Rosenfeld [WES73] as a possible modification of the total-contrast-histogram method of Watanabe [WAT74]. Each class of the average-contrast-histogram was defined as the ratio of the corresponding classes of the total-contrast histogram of method M4 and the feature value frequency histogram of method M1. This method was rejected by Weszka (as an improvement to the Watanabe method) since the resulting thresholds were inconsistent over several different gradient magnitude measures.

The proposed method defines the average-contrast histogram differently from Weszka [WES73] as follows. Consider an edge between adjacent pixels a and b where the feature value at a is I[a] and the feature value at b is I[b] and (without loss of generality) $I[a] \leq I[b]$. We define the edge between pixels a and b to be detected by threshold T if and only if $I[a] \leq T < I[b]$. We can then define the number of edges detected by

threshold T as:

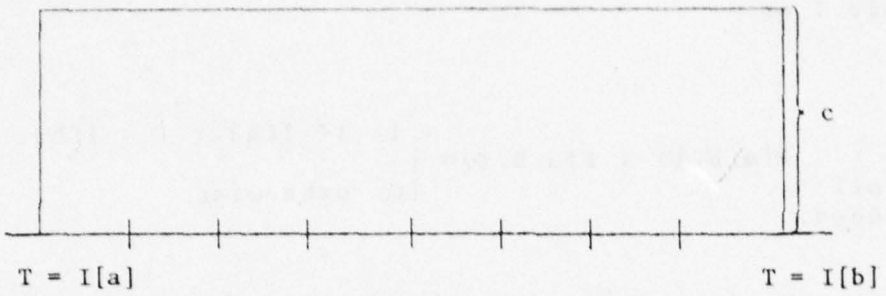
$$N(T) = \sum_{\substack{\text{all}\\ \text{edges}}} f(a,b,t) \;|\; f(a,b,t) = \begin{cases} 1, & \text{if } I[a] \le T < I[b] \\ 0, & \text{otherwise} \end{cases}$$

Similarly, if the contrast of the edge between pixels a and b is given by $c(a,b) = |\, I[a] - I[b] \,|$ then the total contrast detected by threshold T can be defined as:

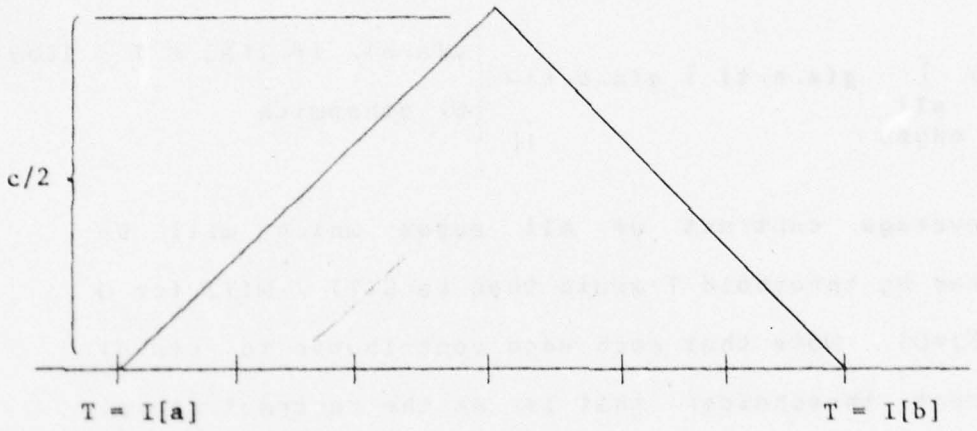
$$C(T) = \sum_{\substack{\text{all}\\ \text{edges}}} g(a,b,t) \;|\; g(a,b,t) = \begin{cases} c(a,b), & \text{if } I[a] \le T < I[b] \\ 0, & \text{otherwise} \end{cases}$$

The average contrast of all edges which will be detected by threshold T would then be C(T) / N(T) (or 0 if N(T)=0). Note that each edge contributes to c(a,b) different thresholds; that is, as the contrast c(a,b) increases, the set of possible thresholds which detect the edge also increases.

Figure 3a shows how a particular edge updates C(T) across the set of possible thresholds. Since for each threshold which detects the edge, the contribution to C(T) is c(a,b) this method is referred to as the uniform contrast method (M6). This method implies that any of the thresholds in the interval (I[a],I[b]) are equally acceptable. If the thresholding step was the



a) uniform contrast method



b) relative contrast method

Figure 3. C(T) Update Functions for the Edge between Pixels a and b.

last step in the segmentation process, then there would be no reason to differentiate between the thresholds in the interval (I[a],I[b]); however, the thresholding process is followed by an error correcting relaxation process discussed in sections 4.0 to 4.7 below. Let us assume that the edge between pixels a and b matches the correct segmentation; that is, failure to detect the edge represents a type two error. (Later we will define a relaxation process which operates by shifting the feature measurement I[x] to increase local consistency.) Therefore the possible thresholds in (I[a],I[b]) are not equally acceptable, because a shift of I[a] toward I[b], implies that a threshold near I[a] would result in a type two error and similarly for a shift in I[b] toward I[a]. Therefore, the threshold furthest from I[a] and I[b] should be favored because it will tolerate larger changes in I[a] and I[b] before it fails to detect the edge. We satisfy this criterion using a definition of contrast relative to the threshold :

$$c(a,b) = \mathrm{MIN}(\mid I[a]-T \mid , \mid I[b]-T \mid) .$$

Figure 3b shows how a particular edge would update C(T) for this definition of contrast. Since the contrast is defined to be relative to the threshold, this method is referred to as the relative contrast method (M7).

3.3 Multiple Threshold Selection.

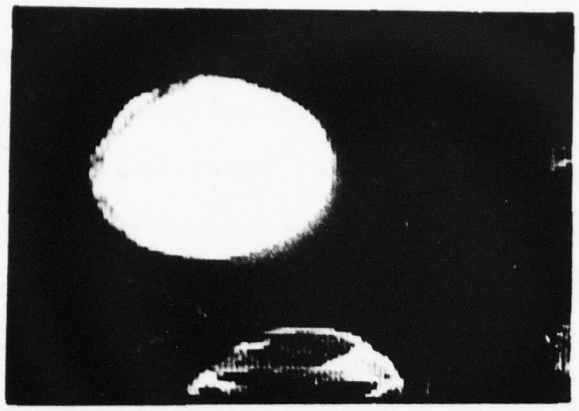
The proposed threshold selection algorithm facilitates the selection of additional thresholds. For any initial threshold T[0], all edges which provided a non-zero contribution to the histogram class corresponding to T[0] are detected by thresholding the image at T[0]. We then repeat the calculation of the histogram as before except that any edge already detected by T[0] (or in general, any previous threshold) will not contribute to the histogram. Let us assume that this new histogram has a peak at threshold T[1]. The average contast of edges detected by T[1] can be no greater than the average contrast of edges detected by any previous threshold such as T[0]. Clearly, one could continue to select thresholds until the maximum average contrast for any threshold fell below some threshold $\theta > 1$. This θ is referred to as the minimum average contrast criterion for threshold selection.

3.4 Threshold Selection Results.

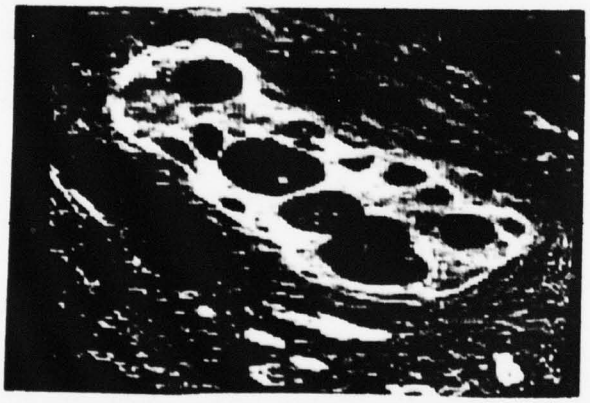
The seven different algorithms for threshold selection listed in Table 1 were applied to three different images of widely varying semantic content and complexity. The simplest image shows a white blood cell on a dark background. This image contains

relatively little texture. The white and red blood cells in the image do contain internal structure. This image is typical of the kind of images for which threshold segmentation methods have been successfully utilized. The second image shows a photo-micrograph of a breast duct exhibiting cribriform morphology (an abnormality often indicative of carcinoma). This image contains much more texture than the first image and the foreground/background relationship is much less distinct. The third image is a complex outdoor scene of a house with bushes and trees. This image contains a variety of objects and textures with no straightforward foreground/background distinction. These three images are shown in Figure 4.

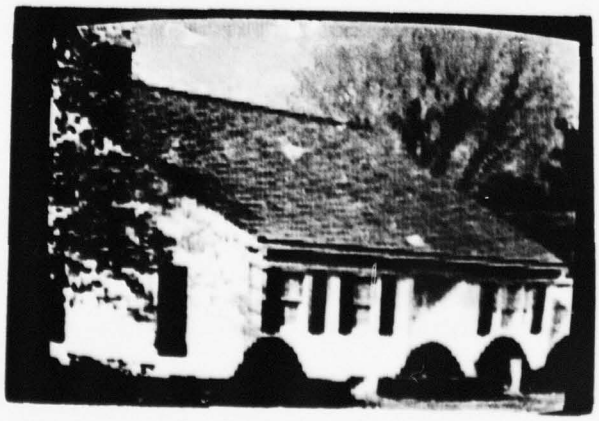
For the simple blood cell image, all of the methods except methods M4 and M5 select a threshold which detects the boundary of the nucleus of the white blood cell (see figure 5). The threshold selected by method M5 detects the cytoplasm boundary of the white blood cell and some of the red blood cell boundaries. The threshold selected by method M4 is too low and fails to detect almost all of the important edges in the image. The method fails since the histogram is dominated by the numerous weak texture edges found in the large background region. The histograms for method



a) white blood cell



b) cribriform breast duct



c) house scene

Figure 4. Three Sample Images.

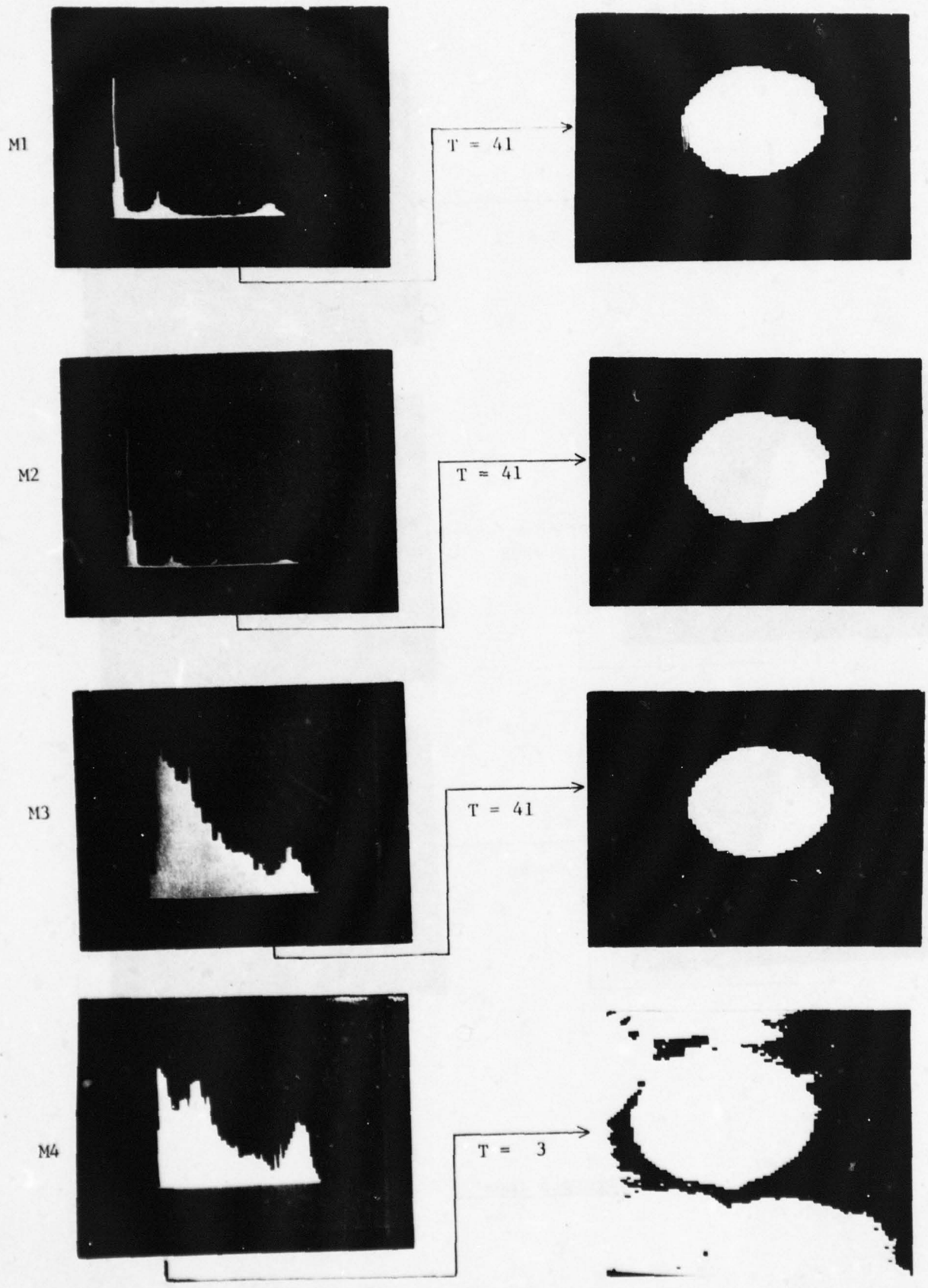


Figure 5. Blood Cell; Threshold Selection Methods.

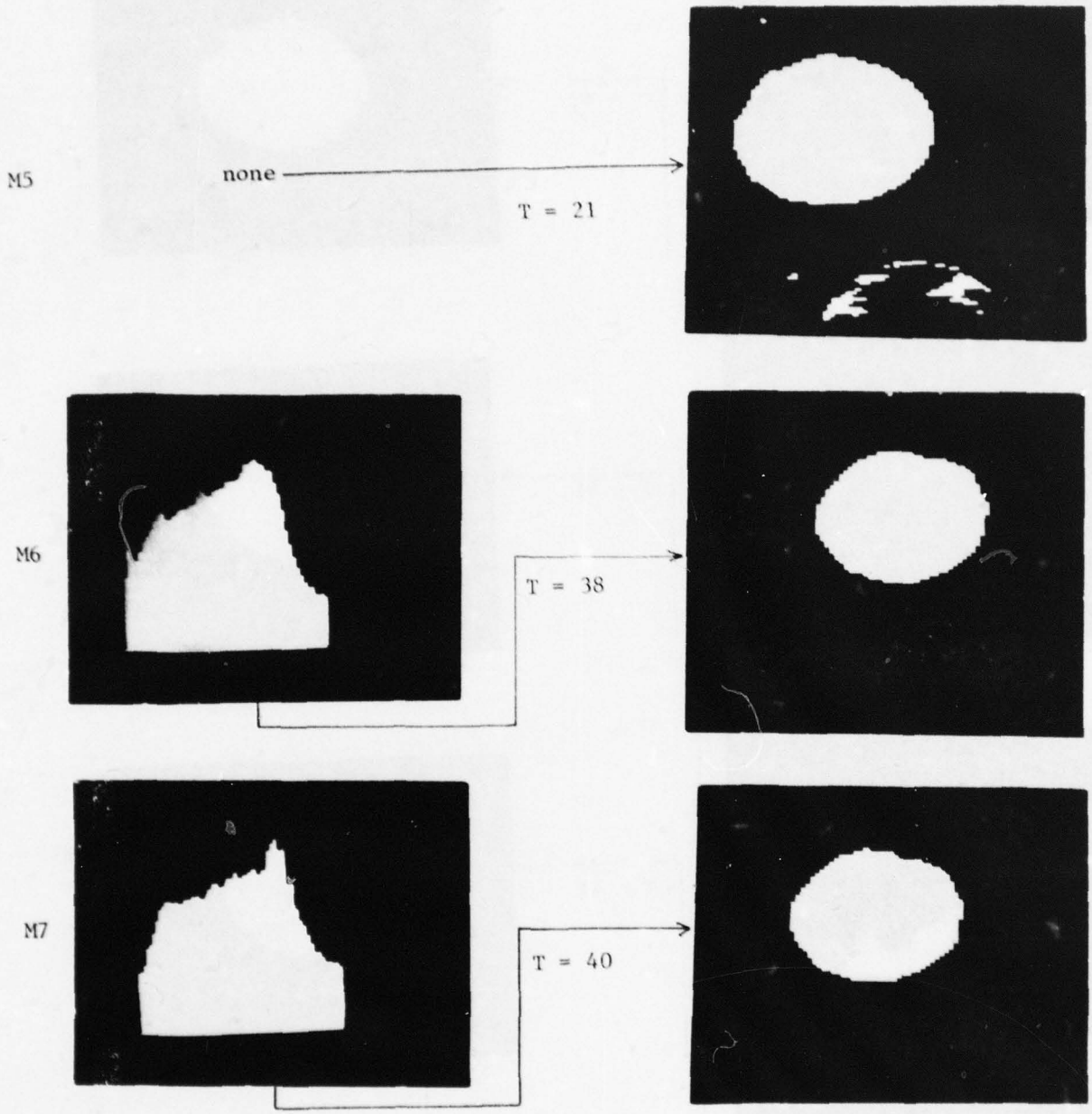


Figure 5 (cont.)

M1 and M2 each contain two distinct valleys; however, the valleys are long and flat making selection of a threshold difficult. The multitude of local maxima and minima make threshold selection more difficult for method M3. The histograms for methods M6 and M7 are quite smooth and selection of the threshold is clear.

Methods M1 and M2 do not generate good thresholds for the breast duct image (figure 6). The algorithms fail since the assumption that image regions are uniform and therefore generate discrete peaks in the histogram is violated for this image. Note that the histograms for methods M1 and M2 are virtually without any valleys, making threshold selection almost arbitrary. The methods M3, M5, M6, and M7 all select thresholds which lead to reasonable segmentations. All of the thresholded images exhibit a plethora of texture edges which are not part of the object boundary.

All of the algorithms select thresholds which result in reasonable segmentations for the house image (see figure 7). Each method detects some of the region boundaries in the image, however, none of the algorithms detect a high percentage of the semantically important boundaries. Several different thresholds could have been selected from the histograms generated

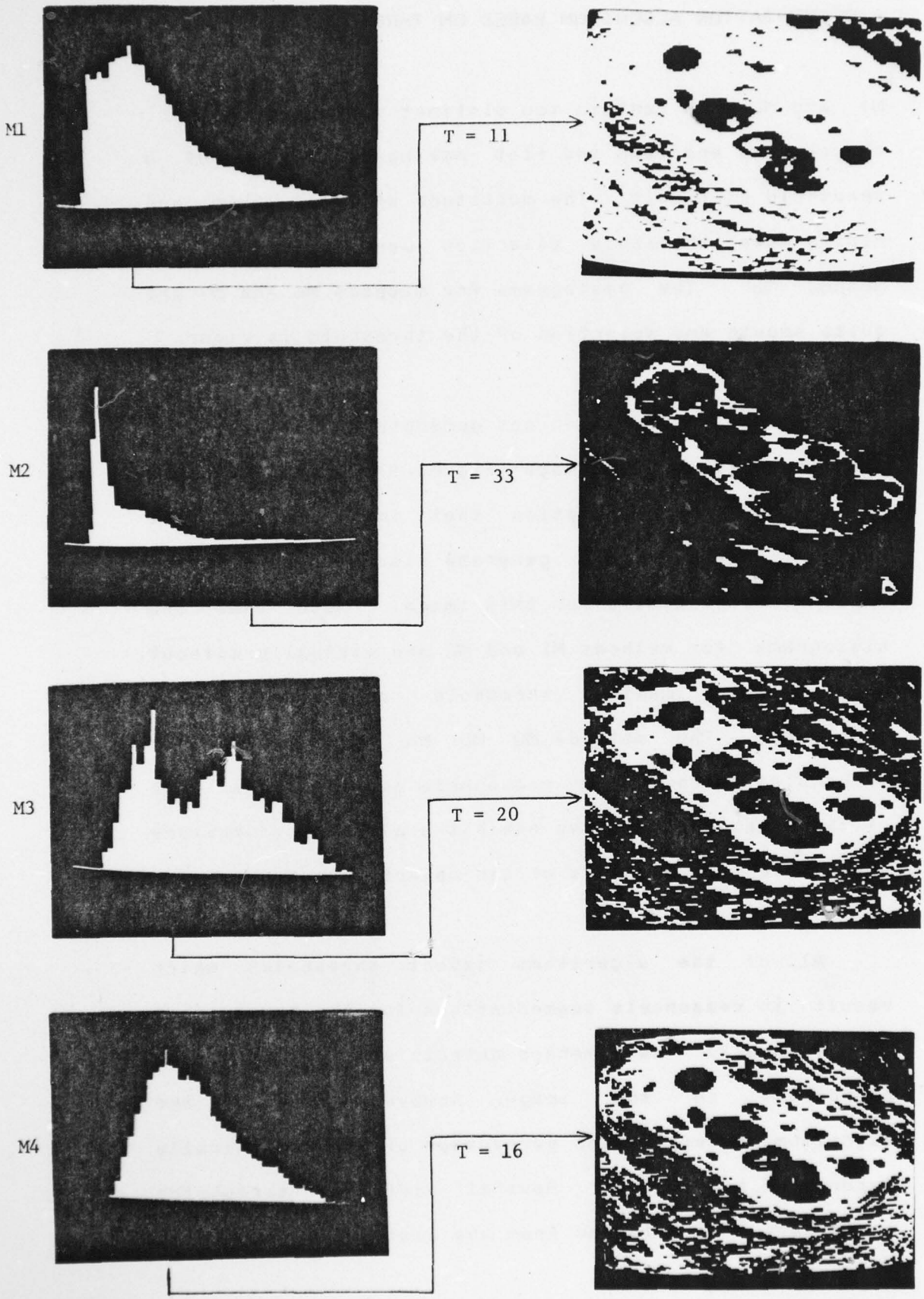


Figure 6. Breast Duct; Threshold Selection Methods.

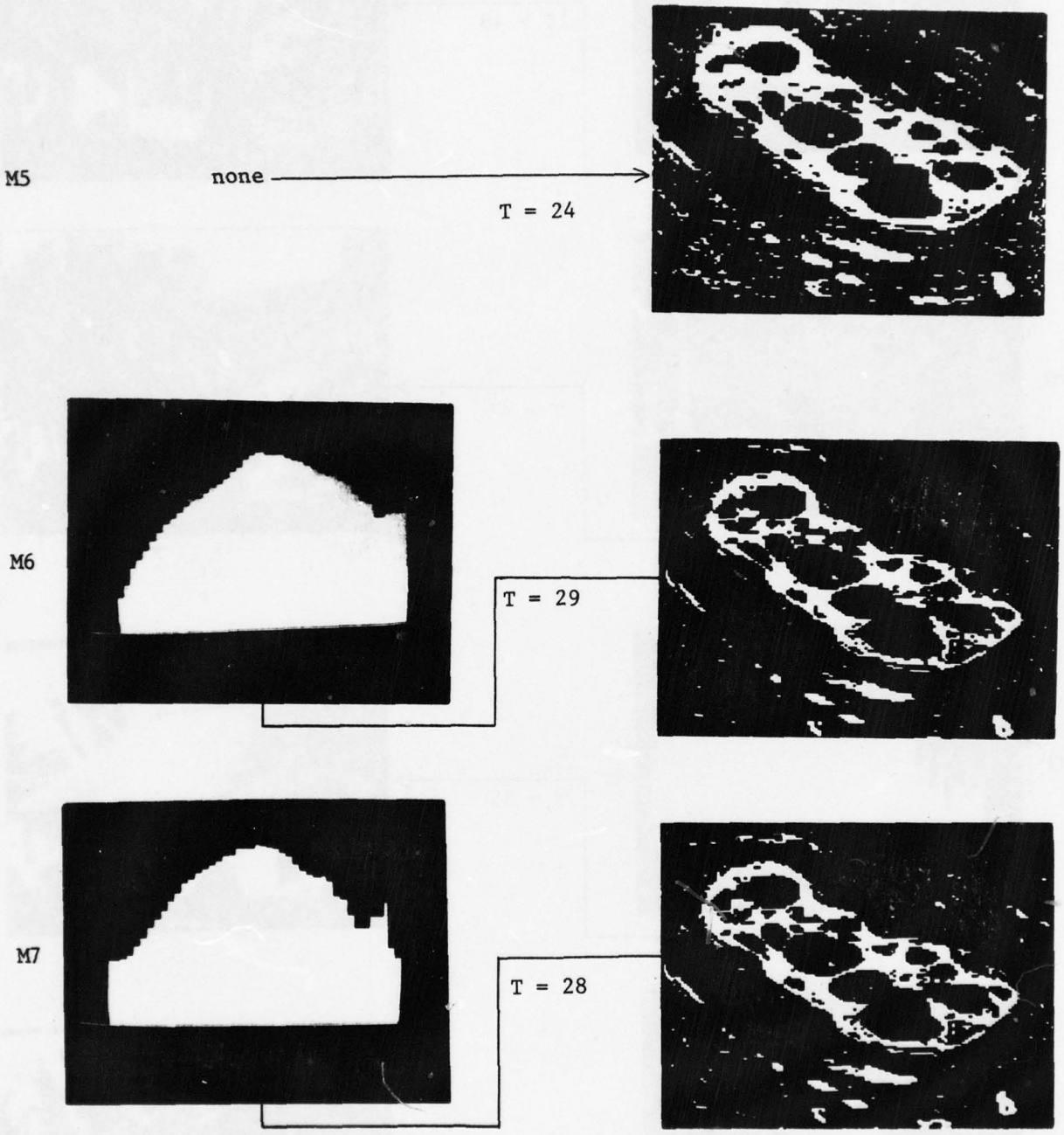


Figure 6 (cont.)

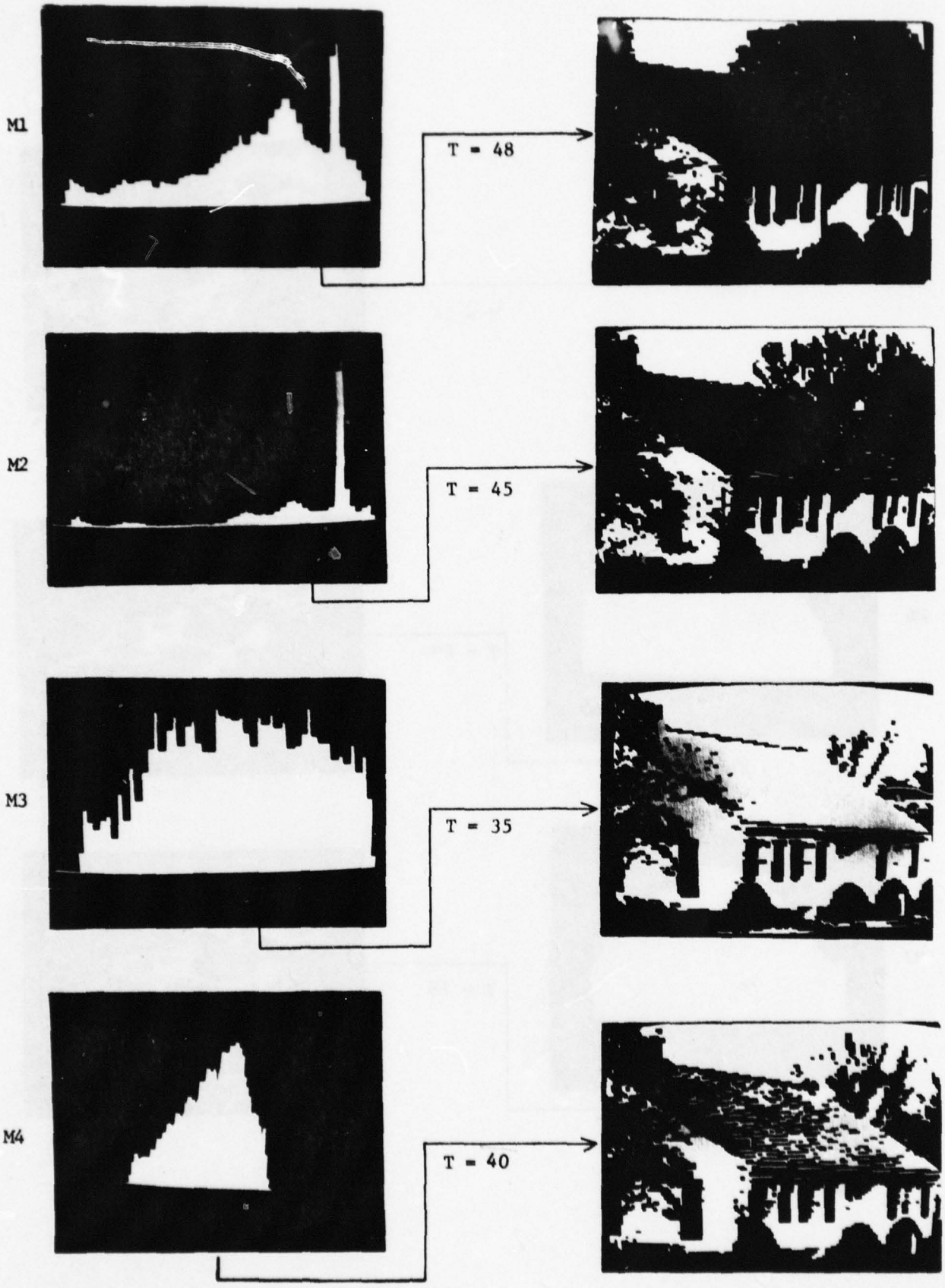


Figure 7. House; Threshold Selection Methods.

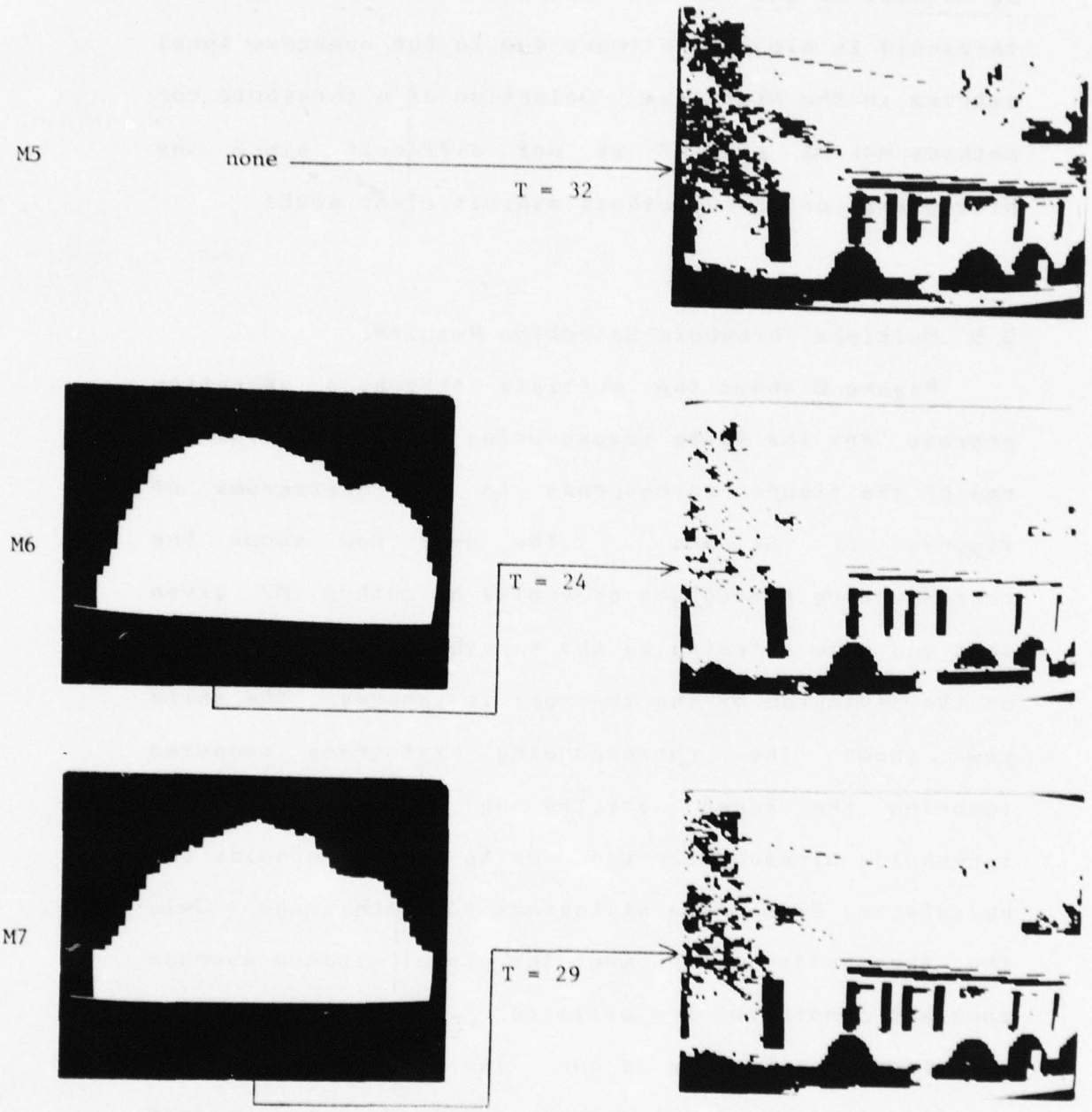


Figure 7 (cont.)

by methods M1 and M2. For method M3 the selection of a threshold is almost arbitrary due to the numerous local extrema in the histogram. Selection of a threshold for methods M4, M6 and M7 is not difficult since the histograms for these methods exhibit clear peaks.

3.5 Multiple Threshold Selection Results.

Figure 8 shows the multiple threshold selection process for the three images using method M7. The top row of the figure corresponds to the histograms of figures 5, 6, and 7. The next row shows the corresponding histograms generated by method M7 given that any edge detected by the threshold selected (based on the histogram of the top row) is ignored. The third row shows the corresponding histograms computed ignoring the edges detected by any of the three thresholds already selected. Up to five thresholds can be selected from these histograms for each image. Only the thresholds which meet the global minimum average contrast constraint are utilized. Selection of this criterion is currently ad hoc. The constraint utilized in the following experiment required an average contrast greater than 2.0. The bottom row shows the resulting region labels encoded as gray levels for the three images thresholded at all of the selected thresholds. These images preserve most of the

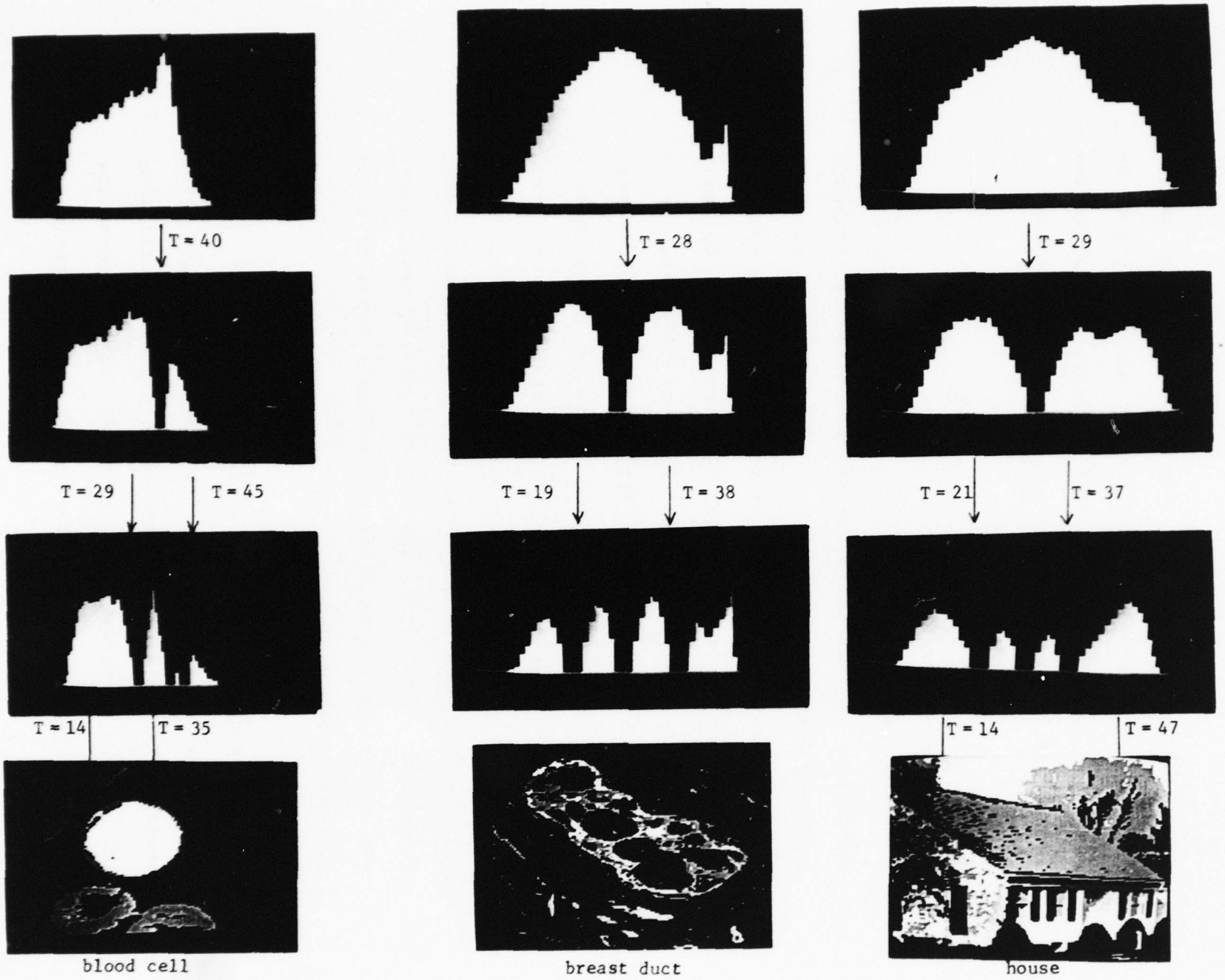


Figure 8. Multiple Selection Results Using Relative Contrast Method.

important boundaries of the image. These images represent a reduction from the original intensity of 64 distinct labels to 6, 4 and 6 distinct labels for the blood cell, breast duct, or house image respectively.

4.0 Threshold Relaxation.

All of the threshold selection algorithms discussed in the previous section select a threshold based on some global measure. Since the measure is global across the image, it is probable that subareas exist for which the threshold selected is not optimal. Furthermore, the measured value of the feature is subject to errors due to digitization and discrete representation of the image. Together these factors imply that regardless of the threshold selection algorithm utilized, the resulting segmentation often will contain many of the three error types previously discussed. The type two errors (missing real edges) are reduced by the selection of multiple thresholds as discussed previously.

The purpose of the relaxation procedure is to reduce the frequency of the type one and type three errors. The relaxation achieves this goal by utilizing the local context of each pixel to determine the relative likelihood that the pixel is correctly or

incorrectly labeled. For the purposes of this paper, the local context of a pixel is the neighborhood of eight immediately adjacent pixels. The process operates by updating in parallel both the label of the pixel and the feature measurement at the pixel so as to reduce local inconsistency. The process terminates when no more changes are possible.

The relaxation is simply an interpolation process between the feature measurement at pixel x and the spatial context of x. The relaxation is based on the premise that pixel x and the pixels in the spatial context of x probably belong to the same region in the image and therefore, the feature mesurement at x and its neighbors should be on the same side of the threshold. The mechanism for reducing local inconsistency then shifts the feature measurement at x toward the measurement at the neighbors in order to increase the likelihood that x and its neighbors lie on the same side of T. This interpolation is controlled based on the following additional premises:

1. The likelihood that pixel x and pixel y are part of the same region is inversely proportional to the distance between x and y.

2. The likelihood that pixel x and pixel y belong to the same region is inversely proportional to the likelihood of an edge existing somewhere between pixel x and pixel y.

3. Finally, the shift of the feature measurement

at x should be inversely proportional to the ambiguity of the neighborhood.

The third premise allows the relaxation to converge quickly when reliable information is available in a subarea of the image. These areas form "islands of reliablility" [LES77] from which information propagates to the adjoining image areas. The relaxation does not change the value of the feature measurement until adequate information is available at the pixel.

Table 2 below summarizes much of the notation. The neighborhood of pixel x is denoted N[x]. The label of the pixel L[x] is an element of $\mathcal{L}=\{0,1\}$, with L[x]=0 whenever the feature measurement I[x] is less than or equal to the threshold T, and L[x] = 1 otherwise.

The neighborhood elements are partitioned into subsets N[1,x] and N[0,x]. N[1,x] consists of those elements of the neighborhood labeled 1 while N[0,x] consists of those elements of the neighborhood labeled 0. The pixels in N[1,x] are considered to be consistent with the labeling L[x]=1 and inconsistent with the labeling L[x]=0. Likewise the elements of N[0,x] are considered to support the labeling L[x]=0 and to contradict the labeling L[x]=1.

Table 2. Threshold Relaxation Notation.

1. The set of labels $\mathcal{L}$ = { 0, 1 }.

2. The threshold used is T.

3. The feature measurement at pixel x is I[x].

4. The label at pixel x is L[x], where
   L[x] = 0 if I[x] ≤ T and
   L[x] = 1 if I[x] > T.

5. The neighborhood about pixel x is N[x]. Note that $x \in N[x]$.

6. The set N[0, x] = { y | $y \in N[x]$ and L[y] = 0 }
   The set N[1, x] = { y | $y \in N[x]$ and L[y] = 1 }

7. Imin is the minimum measurement of feature I anywhere in the image.
   Imax is the maximum measurement of feature I anywhere in the image.

4.1 Estimation of Contextual Support.

It is, of course, necessary to quantify the degree to which N[1,x] or N[0,x] supports or contradicts L[x] given the feature measurements at x and for each y in N[x]. The first step is to define a measure of the likelihood that a pixel in the neighborhood is correctly labeled based only on the feature measurement at the pixel. The measure used in this paper is the relative distance of the feature measurement from the threshold. For an element $y \in N[0,x]$ the relative distance is D[0,y] = (T-I[y])/(T-Imin). For an element $y \in N[1,x]$ the relative distance is given by D[1,y] = (I[y]-T)/(Imax-T). Note that the relative distance is 1 when the feature measurement is as far from the threshold as possible and 0 when the feature measurement is at T. One can then define measures of the net contextual support S[0,x] and S[1,x] over the neighborhood as

$$S[0,x] = \sum_{y \in N[0,x]} D[0,y]\, W[x,y]\, G[x,y]$$

$$S[1,x] = \sum_{y \in N[1,x]} D[1,y]\, W[x,y]\, G[x,y]$$

This measure represents the summation of support for L[x]=0 and L[x]=1, respectively, over the entire neighborhood. The W[x,y] term is an inverse function

of the distance between pixels x and y. The specific function W[x,y] utilized for the nine pixel neighborhood is discussed in more detail in section 4.5 below. This term is based upon the assumption that x and y are less likely to belong to the same region as the distance between x and y increases. The G[x,y] function is used to consider the effect of boundaries on the net contextual support measure. This term accounts for the assumption that the larger the intensity or feature gradient between pixels x and y, the more likely it is that the hypothesis that x and y belong to different regions is valid. The gradient measure G[x,y] is discussed in more detail in section 4.6 below.

4.2 Estimation of Relative Confidences for L[x].

The net contextual support measures S[0] and S[1] can easily be converted to the relative confidences of labels L[x]=1 and L[x]=0 by normalization.

$$P(L[x]=1) = S[1,x] / (S[1,x]+S[0,x])$$

$$P(L[x]=0) = S[0,x] / (S[1,x]+S[0,x])$$

These confidences represent the likelihood that x is correctly labeled L[x]=1 or L[x]=0 based on the

information in the neighborhood of x.

4.3 Proposing an Update of I[x].

If x is correctly labeled L[x]=i, then x should be "close" (in feature value I[x]) to all of its neighbors which are also labeled i. We therefore propose a new value I'[x] for I[x] based on the gradient and distance weighted average of the consistently labeled neighbors.

$$AVE[0,x] = \frac{\sum_{y \in N[0,x]} I[y]\, G[x,y]\, W[x,y]}{\sum_{y \in N[0,x]} G[x,y]\, W[x,y]}$$

$$AVE[1,x] = \frac{\sum_{y \in N[1,x]} I[y]\, G[x,y]\, W[x,y]}{\sum_{y \in N[1,x]} G[x,y]\, W[x,y]}$$

If i=1 then the proposed value I'[x]=AVE[1,x] would necessarily exceed T and x would be correctly labeled. Likewise for i=0, the proposed value of I'[x]=AVE[0,x] would be less than T and again x would be correctly labeled. Since we do not know apriori what the "correct" label at x is, we use the relative confidences P(L[x]=1) and P(L[x]=0) to estimate the confidence of a correct label of 1 and 0 at x based on information in the neighborhood of x. This suggests a value of I'[x] based on an interpolation between

AVE[0,x] and AVE[1,x] utilizing the relative confidences as follows:

$$I'[x] = P(L[x]=1)\ AVE[1,x] + P(L[x]=0)\ AVE[0,x].$$

Note that when P(L[x]=i) approaches one, the value of I'[x] is dominated by AVE[i,x] and therefore, I'[x] will be on the correct side of T. However, if P(L[x]=1) = P(L[x]=0) then I'[x] may not be meaningful since it is based on contradictory evidence.

4.4 Using Local Ambiguity to Control the Update of I[x].

We argue that the magnitude of the change in I[x] should be a function of the ambiguity in the neighborhood. High ambiguity exists when the difference between P(l[x]=1) and P(L[x]=0) is small. When the ambiguity is large there is not enough information in the neighborhood to be sure of what the best final labeling of the pixel should be. The strategy that we employed will shift the feature value by only a small amount for ambiguous pixels. This effectively suppresses changes at a pixel until enough information has propagated into the neighborhood of the pixel to make the decision unambiguously. On the other hand, when the relative magnitude of either label is large, the relaxation should quickly converge to the

correct label and appropriate feature value.

The ambiguity measure used is referred to as the confidence of sufficient information and is given by

$$CSI[x] = | P(L[x]=1) - P(L[x]=0) | .$$

The compliment of this measure, the degree of ambiguity or the confidence of insufficient information is given by

$$CII[x] = 1.0 - CSI[x]$$

The feature update function can now be expressed as follows:

$$I[x] <= CII[x]\ I[x] + CSI[x]\ I'[x].$$

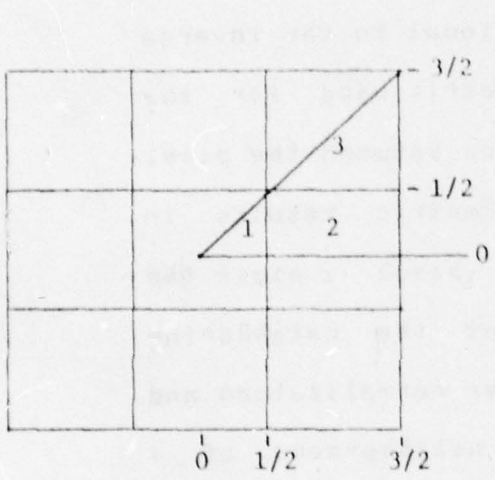
The ambiguity factors CII[x] and CSI[x] balance the effect of the current and proposed values of I[x]. Note that by the definition of the ambiguity terms the change in I[x] can be large only when either P(L[x]=1) or P(L[x]=0) predominates. The net effect of the ambiguity process is to shift the feature value slowly when there is uncertainty and more rapidly as either label begins to clearly dominate.

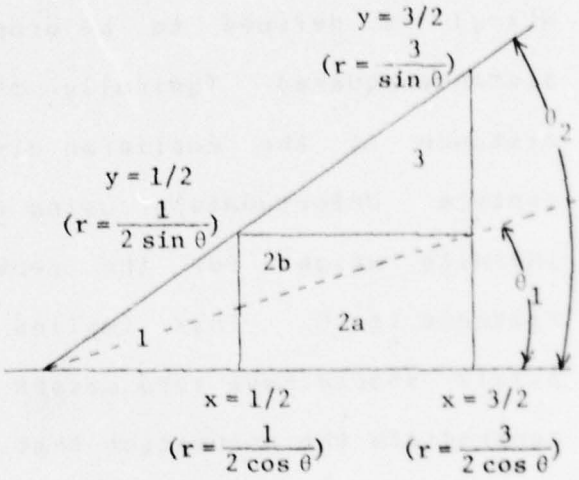
4.5 The Distance Weighting Function W[x,y].

The function W[x,y] represents a spatial weighting over the neighborhood about x based on the distance

between pixel x and pixel y in N[x]. The value of W[x,y] is defined to be proportional to the inverse distance squared. Typically, the metric used for the distance is the Euclidian distance between the pixel centers. Unfortunately, using this metric results in infinite weight for the central pixel x since the distance is 0. This implies that the neighboring pixels should have zero weight after normalization and contradicts the assumption that the neighborhood of x contains useful information.

The digitization process does not represent a point measure at the center of a pixel, but an average measure of the feature integrated across the spatial context of the pixel. The proposed distance metric will be based upon the average distance of the spatial area from the center point of the center pixel (see figure 9). This measure utilizes the measure of distance across the entire area for which the feature value was computed. The simple integration assumes that the feature measurement is based on a uniformly weighted average across the pixel. With additional information about the sampling characteristics of the digitization device, more precise coefficients W[x,y] could be computed.





1) The average distance from the origin for the center pixel is

$$8 \cdot \int_{\theta=0}^{\theta=\theta_2} \int_{r=0}^{r=\frac{1}{2\cos\theta}} r\,dr\,d\theta = 1$$

2) The average distance for the side adjacent pixel is

$$2 \cdot \int_{\theta=0}^{\theta=\theta_1} \int_{r=\frac{1}{2\cos\theta}}^{r=\frac{3}{2\cos\theta}} r\,dr\,d\theta + 2 \cdot \int_{\theta=\theta_1}^{\theta=\theta_2} \int_{r=\frac{1}{2\cos\theta}}^{r=\frac{1}{2\sin\theta}} r\,dr\,d\theta = 5/3$$

3) The average distance for the corner adjacent pixel is

$$2 \cdot \int_{\theta=\theta_1}^{\theta=\theta_2} \int_{r=\frac{1}{2\sin\theta}}^{r=\frac{3}{2\cos\theta}} r\,dr\,d\theta = 5/2$$

Figure 9. Average Distance from Neighborhood Center.

For the neighborhood consisting of the pixel and its eight adjacent neighbors, there are three cases to be consisdered:

1) the central pixel, 2) the edge adjacent pixels, and

3)the corner adjacent pixels. Assuming W[x,y] varies inversely with the square of the distance, the normalized values for the weights W[x,y] corresponding to these three cases are simply: 1 for the central pixel, 9/25 for the edge adjacent pixels, and 4/25 for the corner adjacent pixels.

4.6 The Gradient Weighting Function G[x,y].

The degree to which pixel y in N[x] should be utilized to update x is a function of the likelihood that x and y are part of the same object. It will be assumed that the larger the magnitude of the local gradient, g[x,y], between pixels x and y, the less likely it is that x and y are part of the same region in the image, which implies that G[x,y] should decrease as g[x,y] increases. For the purposes of this paper, g[x,y] is defined as simply the feature difference of the pixels: $g[xy] = |\, I[x]-I[y] \,|$. It is also hypothesized that for some K if $g[x,y] > K$, the influence of y on x is negligible. The value of K was selected such that 95% of all gradients in the image

were smaller.

A third assumption which is embedded in the definition of G[x,y] states that any region which consists of only a single pixel is of no interest. This requires that there exists some y in N[x], y = x, such that x and y are members of the same region.

Now let us define G(x,y) in terms of g(x,y). The heuristics can be translated into the following gradient weighting function:

$$G[x,y] = \begin{cases} 1, & \text{if } x=y \\ 1, & \text{if } gmax=gmin \\ \dfrac{gmax - g[x,y]}{gmax - gmin}, & \text{otherwise} \end{cases}$$

where $g[x,y] = | I[x] - I[y] |$,

$$gmax = \max_{y \in N[x]} (K , g[x,y]) \quad , \text{ and } gmin = \min_{y \in N[x]} (g[x,y])$$

This function sets G[x,y]=1 for y=x. This formulation also guarantees that for at least one of the neighbors of x, G[x,y] will be 1, in particular that neighbor y in N[x]-x which is most like x. If all points in N[x]-x are equally like x, then all will have G[x,y]=1. The function is zero only when g[x,y] is equal to the largest local gradient and g[x,y] is at

least as large as K.

4.7 Threshold Relaxation Results.

Figure 10 shows the operation of the threshold relaxation process on one of the binary labelings of the house scene (T=29). The number of regions present is greatly reduced by the process. Isolated, unsupported pixels quickly shift accross the threshold, in fact many of these pixels are 'corrected' during the first iteration. As expected, larger regions which are strongly supported by the image data are changed very little by the process (note the leftmost bush in figure 10). However, when the exact position of the boundary for a region is not clearly determined by the threshold, the boundaries of the region may shift considerably (note the window above the leftmost bush). These changes take place much more slowly since the boundary pixels must wait for information to propagate before the decision at the pixel is unambiguous. Even after 30 iterations the process had not fully converged.

Figure 11 shows the three segmented images before and after the relaxation process. Each gray level encodes a label of the set of labels produced by intersection of the binary thresholded images. In each

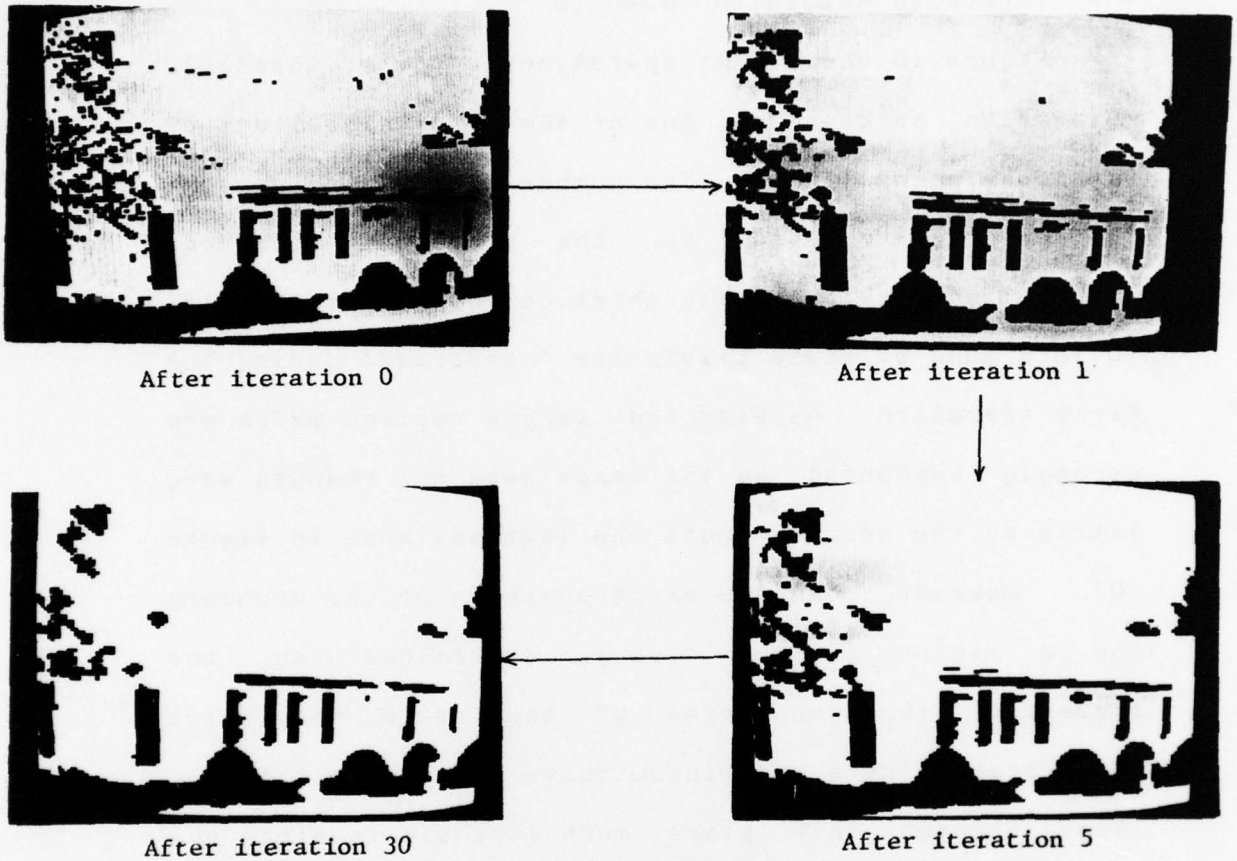
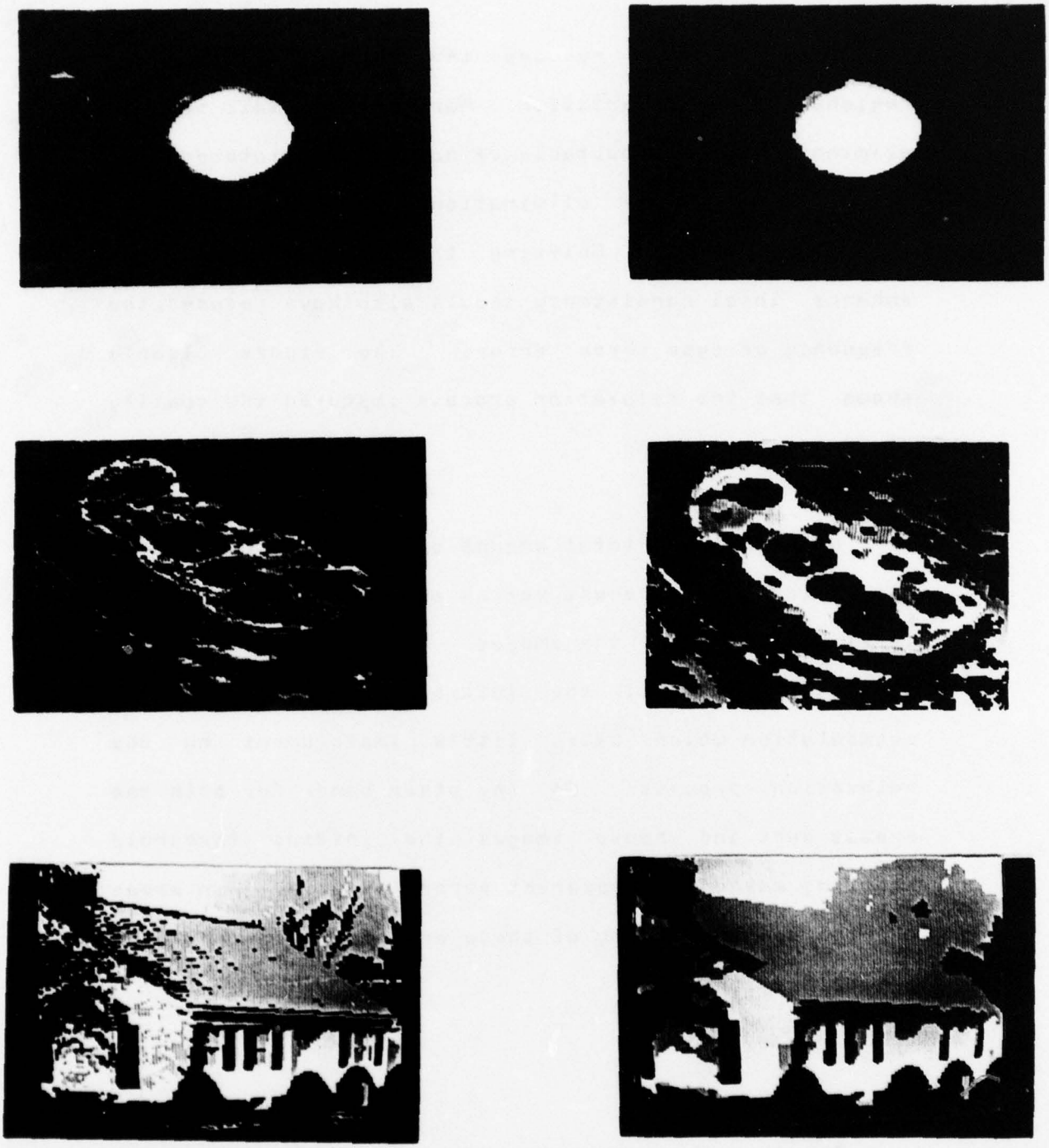


Figure 10. Relaxation on Threshold Labelling: House T = 29.



Before Relaxation

After Relaxation (30 iterations)

Figure 11. Threshold Relaxation Results.

case the relaxation reduces the number of distinct regions in the segmentation. Many of the small regions eliminated were undoubtably of no semantic interest and thus represent the elimination of type one errors in the segmentation. Shifting the edges slightly to enhance local consistency should also have reduced the frequency of type three errors. The figure clearly shows that the relaxation process improved the quality of the segmentations.

Note that the total amount of change induced by the relaxation process varies with the complexity and level of texture of the images. For the simple image of the blood cell the initial thresholds provide a segmentation which bears little improvement by the relaxation process. On the other hand, for both the breast duct and house images the initial threshold labeling makes many apparent errors especially in areas of high texture. Many of these errors are corrected by the relaxation.

5.0 Summary.

Based on the authors subjective evaluation, the proposed segmentation algorithm resulted in acceptable segmentations for the three test images of varying complexity. The first phase of the segmentation

algorithm generated an initial segmentation by selecting a set of thresholds based on local edge information in the image. The proposed threshold selection method compared favorably with alternative threshold selection methods. The initial segmentations contained many type one and type three errors. The selection of multiple thresholds helped to reduce the frequency of type two errors. The relaxation process improved the quality of the final segmentation, reducing the complexity of the segmentation by eliminating many of the type one and type three errors. The relaxation process utilized both local edge information and pixel similarity information to modify the segmentation.

Although the results have been encouraging, it is not clear to what range of images this approach is applicable. One weakness of the current implementation is that only a single image feature is used to generate the segmentation. Future work will consider the automatic threshold selection generalized over a pool of features where the image feature and threshold are selected together for each threshold. This will allow for utilization of color information now ignored by the process.

6.0 Acknowledgments.

The author would like to thank Tom Williams, who initiated this research in our laboratory and who worked closely with the author in developing the initial threshold relaxation algorithm. The author would also like to thank Allen Hanson and Edward Riseman for their efforts in reviewing earlier drafts of this technical report and Janet Turnbull for typing the drafts and preparing the figures.

7.0 References.

KAT65 Y.H. Katz, "Pattern recognition of meteorlogical satellite photography", Proc. 3rd Symp. on Remote Sensing, U. of Michigan, Feb. 1965, pp. 173-214.

LES77 V. Lesser and L. Erman, "A Retrospective View of Hearsay-II Architecture", Proc. 5th IJCAI, Cambridge, Ma., Aug 1977.

NAG78 P. Nagin, "Segmentation Using spatial Context and Feature Space Cluster Labels", COINS TR 78-8, U. of Mass., May 1978.

WAT74 S. Watanabe, "An automated cancer prescreening apparatus: CYBEST", Computer Graphics and Image processing 3, 1974.

WES73 J. S. Weszka, J. A. Vernon, and A. Rosenfeld, "Threshold Selection Techniques 2", TR-260, Computer Science Center, U. of Md., Aug. 1973.

WES74 J. S. Weszka, "Threshold Selection 4", TR-336, Computer Science Center, U. of Md., Oct. 1974.

WES75 J. S. Weszka, "Threshold Selection Techniques 5", TR-349, Computer ScienceCenter, U. of Md., Feb. 1975.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: COINS TR 78-23	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): AN IMAGE SEGMENTATION SYSTEM BASED ON THRESHOLDING	5. TYPE OF REPORT & PERIOD COVERED: INTERIM Rept.
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Ralf Kohler	8. CONTRACT OR GRANT NUMBER(s): ONR N00014-75-C-0459
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Computer and Information Science Department, University of Massachusetts, Amherst, Massachusetts 01003	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Dec 78
11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research, Arlington, Virginia 22217	12. REPORT DATE: 12/78
	13. NUMBER OF PAGES: 53
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): COINS-TR-78-23	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report): Distribution of this document is unlimited.	12. 54 pp.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)	
18. SUPPLEMENTARY NOTES	
19. KEY WORDS (Continue on reverse side if necessary and identify by block number): computer vision; image processing; segmentation; thresholding; relaxation	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The segmentation algorithm proposed in this paper is a complex form of thresholding which utilizes multiple thresholds. The algorithm consists of two major components: a threshold selection component and a relaxation component.

The threshold selection component automatically selects a threshold so as to maximize the global average contrast of edges detected by the threshold

next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

across the image. This algorithm for threshold selection compares favorably with other methods for automatic threshold selection. The threshold selection algorithm can be applied recursively to select additional thresholds by ignoring any edges which have already been detected by previously selected thresholds.

The relaxation component utilizes the immediate spatial context of each pixel to update both the label at the pixel and the feature measurement at the pixel. The update function proposes a new feature value at the pixel defined by a weighted average of the central pixel and all of its neighbors. The weight associated with each pixel (with respect to the pixel being updated) is proportional to the spatial distance between the pixels, the probability that the two pixels are correctly labeled, and the probability that the two pixels belong to the same region. The update function then replaces the feature value at the pixel with a value somewhere between the current value and the proposed value. When the local evidence for shifting the feature value is consistent, then the value adopted will be close to the proposed value; however, when the local evidence is inconsistent, the value adopted will be close to the original value.

The relaxation is independently performed for each threshold selected. The resulting binary images are intersected to produce the final segmentation. This algorithm works well not only for simple images but also produces reasonable segmentations for complex images.