Automatic Multithreshold Selection

SHYUAN WANG AND ROBERT M. HARALICK

Department of Computer Science and Electrical Engineering, Virginia Polytechnic Institute and
State University, Blacksburg, Virginia 24061

Received September 15, 1982; revised October 18, 1982

A recursive technique for multiple threshold selection on digital images is described. Pixels
are first classified as edge pixels or nonedge pixels. Edge pixels are then classified, on the basis
of their neighborhoods, as being relatively dark or relatively light. A histogram of the graytone
intensities is obtained for those pixels which are edge pixels and relatively dark and another
histogram is obtained for those pixels which are edge pixels and relatively light. A threshold is
selected corresponding to the graytone intensity value corresponding to one of the highest
peaks from the two histograms. To get multiple thresholds, the procedure may be recursively
applied first using only those pixels whose intensities are smaller than the threshold and then
only those pixels whose intensities are larger than the threshold.

1. INTRODUCTION

In image segmentation techniques, nothing is simpler than global thresholding.
Two cases are of interest. In case 1, the images simply contain objects and
backgrounds; a graytone threshold value is needed to separate them. In case 2, the
images contain three or more classes of homogeneous regions; we want to get one
"sliced" output image for each of the classes. Case 1 is a special case of case 2. If a
graytone image $P$ is defined over a set $Z_r$ of row coordinates, a set $Z_c$ of column
coordinates, and a graytone range $[g_1, g_2]$, $P: Z_r \times Z_c \to [g_1, g_2]$, then the result of
case 1 is a two-valued image $P_2$, $P_2: Z_r \times Z_c \to \{0, 1\}$, defined by

$$P_2 = \begin{cases} 1 & \text{if } P(r, c) > t \\ 0 & \text{if } P(r, c) \leq t, \end{cases}$$

where $t$ is the graytone threshold and $g_1 < t < g_2$. Similarly, the result of case 2 can
be considered as a $(k + 1)$-valued image $P_k$, $P_k: Z_r \times Z_c \to \{0, 1, \ldots, k\}$, defined by

$$P_k(r, c) = \begin{cases} k & \text{if } P(r, c) > t(k) \\ k - 1 & \text{if } t(k - 1) < P(r, c) \leq t(k) \\ \vdots & \text{if } \vdots \\ 1 & \text{if } t(2) < P(r, c) \leq t(2) \\ 0 & \text{if } P(r, c) \leq t(1), \end{cases}$$

where $t(i)$, $1 \leq i \leq k$, are thresholds and $g_1 < t(1) < t(2) < \cdots < t(k) < g_2$.

A variety of thresholds selection techniques are described in Weszka's [1] survey
article. Our method belongs to the class of global threshold selection based on local
properties. There are two ways to use local properties. First, they can be used to modify the graytone histogram. Second, the histogram of local property values can be used directly to compute a global threshold, as suggested by Watanabe et al. [2].

Chow and Kaneko’s [3] method is useful when global thresholds are inadequate. In their approach, an image is divided into overlapping windows. Local statistics are measured in each window to determine a threshold. Thresholds for every point in the image are obtained by a linear interpolation procedure. Weszka classifies this method as dynamic threshold selection. The method of this paper could be used to find thresholds within the windows thereby constituting a general segmentation technique.

Two new points of view about using local properties are presented in this paper. First, instead of modifying the graytone histograms, we decompose the histogram of local property values. Second, instead of finding a best binary threshold, we work on the multithreshold selection problem. We believe that more clues are provided in the latter case, leading to a general solution of the threshold selection problem.

In Section 2, Watanabe’s experiment is described. In Section 3, we illustrate how to improve Watanabe’s result by classifying edge pixels into three categories. In Section 4, a criterion for a good threshold is proposed. In Section 5, an algorithm for multithreshold selection is described. The final two sections are on experimental results and discussion.

2. WATANABE’S METHOD AND SOME VARIATIONS

Watanabe used the gradient approach to determine a global threshold because, at the borders between objects and background, the gradient should be high. If a gradient image is defined as $D: Z_r \times Z_c \rightarrow R^*$ where $R^*$ is the set of nonnegative real numbers, then the gradient histogram $H_1$ is defined as $H_1: [g_1, g_2] \rightarrow I^*$, where $I^*$ is the set of nonnegative integers, and $H_1(g)$ is the sum of the gradient values for all pixels having graytone $g$, i.e.,

$$H_1(g) = \sum_{(r, c) \in A_1(g)} D(r, c), \quad g \in [g_1, g_2]$$

where $A_1(g) = \{(r, c) \mid P(r, c) = g \text{ where } P \text{ is the graytone image}\}$.

Watanabe selected graytone $t$ as a threshold where $H_1(t)$ is a peak. In the following, some variations of $H_1$ are designed to see what effects they might have.

The first variation is to use the number of pixels in $A_1(g)$. Thus, histogram $H_2: [g_1, g_2] \rightarrow I^*$ is defined as $H_2(g) =$ number of pixels in $A_1, g \in [g_1, g_2]$. Another variation [4] is to use the normalized gradient histogram $H_3, H_4: [g_1, g_2] \rightarrow I^*$, which is defined by $H_3(g) = H_1(g)/H_2(g)$.

In the other variations, we do not use the gradients of all the pixels. A comprehensive comparison of Watanabe’s method and four alternatives was reported in [5]. Histograms of pixels having either low or high values were used. Katz [6] suggested producing a histogram of the gray levels of only those pixels that have high edge values. Similarly, we can calculate $H_1, H_2, H_3$ only at locations where “meaningful” edge responses are detected. To do this, we need a binary edge image $E: Z_r \times Z_c \rightarrow \{0, 1\}$, where

$$E(r, c) = \begin{cases} 1 & \text{if } (r, c) \text{ is identified as an edge pixel}, \\ 0 & \text{otherwise}. \end{cases}$$
TABLE 2.1
Design of Six Different Histograms

<table>
<thead>
<tr>
<th>Calculation</th>
<th>Pixel selection</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All pixels</td>
<td>Edge pixels only</td>
<td></td>
</tr>
<tr>
<td>Gradient</td>
<td>$H_1$</td>
<td>$H_4$</td>
<td></td>
</tr>
<tr>
<td>Number of pixels</td>
<td>$H_2$</td>
<td>$H_5$</td>
<td></td>
</tr>
<tr>
<td>Normalized gradient</td>
<td>$H_3$</td>
<td>$H_6$</td>
<td></td>
</tr>
</tbody>
</table>

Now using the edge image, $H_1$, $H_2$, $H_3$ can be modified into $H_4$, $H_5$, $H_6$, where

$$H_4(g) = \sum_{(r, c) \in A_2(g)} D(r, c)$$

$$A_2(g) = \{(r, c)|P(r, c) = g, E(r, c) = 1\},$$

$H_5(g)$ = number of pixels in $A_2(g)$,

and

$$H_6(g) = H_4(g) / H_5(g).$$

An experiment using Watanabe's method and variations then has the possibilities shown in Table 2.1. In this experiment, using a Landsat image, we create the image $E$ by Haralick's [7] zero crossing of second directional derivative edge operator which detects good step edges even in noisy pictures. A pixel is identified as an edge pixel if and only if there is some point in the pixel's area having a zero crossing of the second directional derivative taken in the direction of a non-zero gradient at the pixel's center.

Fig. 2.1. Original Landsat image.
For the Landsat image in Fig. 2.1 and the corresponding binary edge image in Fig. 2.2, the histograms $H_1 \sim H_6$ are shown in Fig. 2.3. From these histograms, it can be concluded that $H_1$, $H_2$, $H_4$, and $H_5$ are alike, and $H_3$, $H_6$ are not useful. In the following sections, we will concentrate on how to split $H_5$ to get better results.

3. CLASSIFYING EDGE PIXELS

Not all edge pixels in the binary edge image $E$ have the same roles in the relative brightness of the graytone image $P$. As in Fig. 3.1, edge pixels may be bright (upper plane), dark (lower plane), or gray (slope). Thus, information contained in the above histograms is mixed and ambiguous. One should create another classified edge image $C: Z_r \times Z_c \rightarrow \{0, 1, 2\}$ where

\[
C(r, c) = \begin{cases} 
1 & \text{if edge pixel } (r, c) \text{ is dark compared to} \\
& \text{most of its neighbors, or it can be called} \\
& \text{a dark edge pixel.} \\
2 & \text{if edge pixel } (r, c) \text{ is bright compared to} \\
& \text{most of its neighbors, or it can be called} \\
& \text{a bright edge pixel,} \\
0 & \text{otherwise.} 
\end{cases}
\]

Next, $H_5$ can be split into $H_d$ and $H_b$. $H_d$ is the histogram of the dark edge pixels with $H_d(g) = \text{number of elements in } \{(r, c)|P(r, c) = g, C(r, c) = 1\}$. Also $H_b$ is the histogram of numbers of bright edge pixels with $H_b(g) = \text{number of elements in } \{(r, c)|P(r, c) = g, C(r, c) = 2\}$. The histograms for $C(r, c) = 0$ are omitted because they are not interesting. Now, there is a graytone $t_1$ at which $H_d(t_1)$ is a peak. Also, there is a graytone $t_2$ at which $H_b(t_2)$ is a peak. It looks as though both $t_1$ and $t_2$ could be selected as thresholds, but it should be noted that $t_1$ and $t_2$ are totally different ways of getting regions. By thresholding at $t_1$, we are supposed to get dark regions, and by thresholding at $t_2$, we are supposed to get bright regions. This is the
Fig. 2.3a. Histogram $H_1$.

Fig. 2.3b. Histogram $H_2$. 
Fig. 2.3c. Histogram $H_3$. 

Fig. 2.3d. Histogram $H_4$. 

Fig. 2.3c. Histogram $H_3$.

Fig. 2.3f. Histogram $H_6$. 
key concept in this paper. We will come back to this in the next section. First, we describe a simple heuristic method to create the edge classified image.

For each pixel \((r, c)\) let \(g_{\min}(r, c)\) and \(g_{\max}(r, c)\) be the minimum and maximum graytone values in the set \(\{P(x, y)| (x, y)\text{ is one of } (r, c)'s\text{ eight neighbors}\}\).

Next, calculate
\[
\mu(r, c) = \frac{(g_{\max}(r, c) + g_{\min}(r, c))}{2},
\]
\[
\Delta(r, c) = \frac{(g_{\max}(r, c) - g_{\min}(r, c))}{2N},
\]

where \(N\) is a constant, to get different \(\Delta\) values.

The \(C\) image is defined as
\[
C(r, c) = \begin{cases} 
1 & \text{if } P(r, c) \text{ is in } [g_{\min}(r, c), \mu(r, c) - \Delta(r, c)] \text{ and } E(r, c) = 1, \\
2 & \text{if } P(r, c) \text{ is in } [\mu(r, c) + \Delta(r, c), g_{\max}(r, c)] \text{ and } E(r, c) = 1, \\
0 & \text{otherwise.}
\end{cases}
\]

If \(N\) is small and \(\Delta\) is large, then fewer pixels will be classified as dark and bright edge pixels. Conversely, if \(N\) is large and \(\Delta\) is small, then more pixels will be

![Fig. 3.2: Classified edge pixels of Fig. 2.3. Bright edge pixels are shown bright; dark edge pixels are shown dark.](image-url)
Fig. 3.3a. Histogram $H_a$.

Fig. 3.3b. Histogram $H_b$. 
classified as dark and bright edge pixels. Thus we can get different $C$ images by changing $N$. In most cases, $N$ is set as 3. The classified edge image of the Landsat image is shown in Fig. 3.2, and the $H_d$, $H_b$ histograms are shown in Fig. 3.3.

4. CRITERION FOR GOOD THRESHOLDS

The peak occurs at graytone 12 in $H_d$ and occurs at graytone 17 in $H_b$ for the Landsat image. Which one is a better threshold? Before looking at these two histograms in detail, let us study an ideal case as in Fig. 4.1, where region a is bright, region b is gray and region c is dark. Note that b is bright relative to c, but is dark relative to a. In terms of a classified edge image, b’s edge pixels at the borders between b and c are bright edge pixels and b’s edge pixels at the borders between a and b are dark edge pixels. Thus, in the graytone range of b’s pixels, both the $H_d$ and $H_b$ histograms are expected to have significant population counts. Note also that c’s edge pixels at the border between b and c are always classified as dark edge pixels, and that a’s edge pixels at the border between a and b are always classified as bright edge pixels.

These observations correspond quite well to our definition of the $P_k$ image in Section 1. The pixels with $P_k(r, c) = k$ will always be bright relative to the other pixels with $P_k \in [0, k-1]$. The pixels with $P_k(r, c) = 0$ will always be dark relative to the pixels with $P_k \in [1, k]$. Also the pixels with $P_k(r, c) = i$, $i \in [1, k-1]$ will be dark relative to pixels with $P_k \in [i+1, k]$ and bright relative to pixels with $P_k \in [0, i-1]$ at the same time.

Now we are ready to get some insight into $H_d$ and $H_b$. Scan $H_d$ and $H_b$ from left to right. In the range $[1, 8]$, $H_b$ has almost zero frequencies, but $H_d$ does not. This means that if we threshold the image using $t \in [1, 9]$ to get dark regions, they are purely dark. Starting at graytone $g = 9$, the total population counts in the range $[1, g]$ are not zero in both $H_d$ and $H_b$, which means that if we threshold the image using $t \in [10, 36]$, the dark regions we get are not so purely dark as if we use $t \in [1, 9]$. This suggests the definition of the probability of getting dark regions by

![Fig. 4.1. An ideal image: region a is bright, region b is gray, and region c is dark.](image)
Fig. 4.2a. Printout of $P_d(t)$: the probability of getting dark regions if thresholding at $t$.

Fig. 4.2b. Printout of $P_b(t)$: the probability of getting bright regions if thresholding at $t$. 

thresholding at \( t \) is

\[
P_d(t) = \frac{\text{Total population counts from } g_1 \text{ to } t \text{ for } H_d \text{ only}}{\text{Total population counts from } g_1 \text{ to } t \text{ for } H_d \text{ and } H_b} \times 100\%
\]

\[
= \frac{\sum_{i=g_1}^{t} H_d(i)}{\sum_{i=g_1}^{t} H_d(i) + \sum_{i=g_1}^{t} H_b(i)} \times 100\%
\]

Similarly, by scanning \( H_d \) and \( H_b \) from right to left, we can define the probability of

Fig. 5.1. An example of selecting two threshold values to separate three different regions.
Fig. 6.1a. $H_d$ for graytone range [13, 36].

Fig. 6.1b. $H_b$ for graytone range [13, 36].
Fig. 6.1c. $P_d$ for graytone range [13, 36].

Fig. 6.1d. $P_b$ for graytone range [13, 36].
Fig. 6.2a. $H_d$ for graytone range [13, 25].

Fig. 6.2b. $H_b$ for graytone range [13, 25].
Fig. 6.2c. $P_d$ for grayscale range [13, 25].

Fig. 6.2d. $P_a$ for grayscale range [13, 25].
getting bright regions by thresholding at \( t \) as

\[
P_b(t) = \frac{\sum_{i=t}^{g_2} H_b(i)}{\sum_{i=t}^{g_2} H_d(i) + \sum_{i=t}^{g_2} H_b(i)}.
\]

Figure 4.2 shows the printout of \( P_d \) and \( P_b \) with respect to graytone values for the Landsat image. Because \( P_d(12) \) is higher than \( P_b(17) \), 12 is selected as the threshold value. In general, the rule is

Rule. If the peak of \( H_d \) occurs at \( g_d \) and the peak of \( H_b \) occurs at \( g_b \), and \( P_d(g_d) \geq P_b(g_b) \), then \( g_d \) is selected as the threshold. Conversely, if \( P_d(g_d) < P_b(g_b) \), \( g_b \) is selected as the threshold.

5. RECURSIVE MULTITHRESHOLD SELECTION

Using the threshold value selected by the rule in the last section, we can get one image of bright regions and another image of dark regions. Repeating the steps for
Fig. 6.5. (a) Perfect checkerboard for graytone range [10, 15]. (b) Edge image. (c) Sliced binary image for graytone range [10, 10].

Fig. 6.6. (a) Noisy checkerboard for graytone range [10, 15]. (b) Edge image. (c) Sliced binary image for graytone range [10, 10]. (d) Sliced binary image for graytone range [10, 12].
these two new images, we can get other images of even brighter or darker regions. This is exactly the same idea Ohlander [8] used in his recursive segmentation method. A similar method was devised by Milgram and Kahl [9] who used the concept of convergent evidence to accept or reject object regions in recursive region extraction. Milgram and Herman [10] discuss convergent evidence from clustering thinned edge points in a 2-D histogram using gray level value and edge value as axes. An example of the recursive thresholding method is shown in Figure 5.1.

In the implementation, we do not need to actually create the new images during the selection of thresholds. Instead, we can keep on reducing the graytone range \([g_1, g_2]\) as the thresholding removes either the most bright or the most dark regions. Thus, the algorithm is

Initially, \(g_1 \leftarrow \) actual minimum graytone in image \(P\),
\(g_2 \leftarrow \) actual maximum graytone in image \(P\),
\(j \leftarrow 0\).

Repeat
begin: \(j \leftarrow j + 1\)
Create \(H_d, H_b\)
If the total population counts in either \(H_d\) or \(H_b\) are not significant, stop.
Use the rule in Section 4 to select threshold \(t(j)\) in \([g_1, g_2]\).
If \(t(j) = g_d\) then \(g_1 \leftarrow g_d + 1\)
If \(t(j) = g_b\) then \(g_2 \leftarrow g_b - 1\)
end.

6. EXPERIMENTAL RESULTS

Let us continue with the Landsat image example. After \(g_d = 12\) is selected as the first threshold, the graytone range is reduced to \([13,36]\). The printouts of \(H_d, H_b, P_d, P_b\) for this range are shown in Fig. 6.1. \(g_b = 26\) is selected as the next threshold, and the graytone range is reduced to \([13,25]\). The printouts of \(H_d, H_b, P_d, P_b\) for this new range are shown in Fig. 6.2. \(g_b = 20\) is selected as the next threshold, and the graytone range is reduced to \([13,19]\). Since the total population counts for \(H_d\) and \(H_b\) are not significant, the algorithm stops.

Instead of showing the \((k + 1)\)-valued image \(P_k\), we show the binary sliced image for each graytone range, i.e., for a graytone range \([g_x, g_y]\), sliced image \(S: Z_r \times Z_c \rightarrow \{0, 1\}\) is defined as

\[
S(r, c) = 1 \quad \text{if} \quad g_x \leq P(r, c) \leq g_y,
= 0 \quad \text{otherwise}.
\]

The sliced image for \([1, 12]\) is shown in Fig. 6.3, and the sliced image for \([20, 36]\) is shown in Fig. 6.4. The regions in Fig. 6.3 represent shadows, and the regions in Fig. 6.4 represent vegetated area. The explanations are given in Wang et al. [11]. Because the sliced image for \([26, 36]\) is just a subset of Fig. 6.4, we do not show it.

Experiments were done for a variety of other images such as a checkerboard, a noisy checkerboard, a chair, and a house. The original images, edge images, and sliced images are shown in Figs. 6.5 to 6.8.
7. DISCUSSION

For the Landsat image, the peak of $H_1$ occurs at the graytone 16. By Watanabe's method, this would be selected as threshold. However, in our method, 12, 20, and 26 are selected as thresholds, and 16 is never selected because it is not compatible with the others in getting pure bright or dark regions.

Reasonable results are obtained in the last section for several quite different images, which indicates that the approach is quite general.

REFERENCES

Fig. 6.8. (a) House image. (b) Edge image. (c) Sliced binary image for graytone range [1, 116]. (d) Sliced binary image for graytone range [117, 191]. (e) Sliced binary image for graytone range [192, 255].


