Lecture 6: Segmentation by Point Processing

Harvey Rhody
Chester F. Carlson Center for Imaging Science
Rochester Institute of Technology
rhody@cis.rit.edu

September 27, 2005

Abstract

Applications of point processing to image segmentation by global and regional segmentation are constructed and demonstrated. An adaptive threshold algorithm is presented. Illumination compensation is shown to improve global segmentation. Finally, morphological waterfall region segmentation is demonstrated.

Image Segmentation

Image segmentation can be used to separate pixels associated with objects of interest from the image background.

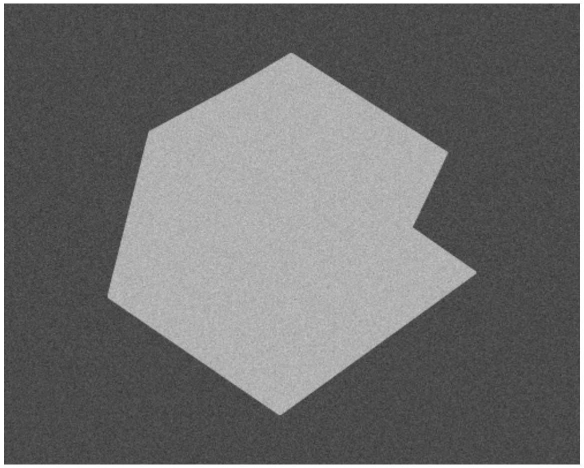
This is an important step in many imaging applications of automated analysis and robotics. The opposing requirements of accuracy and speed are always present.

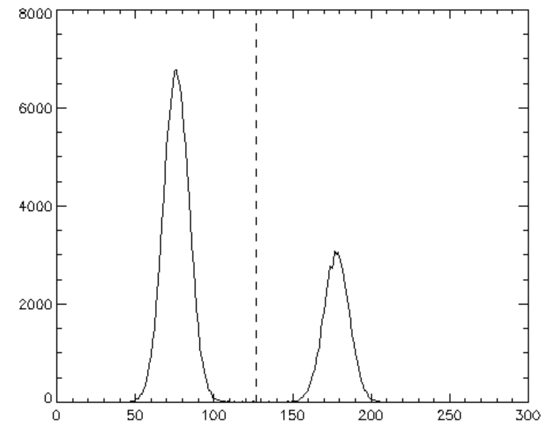
Segmentation on a simple pixel-by-pixel basis using threshold decisions is very fast but restricted to applications where there is sufficient contrast between objects and background.

We will examine techniques for threshold segmentation using global and adaptive methods.

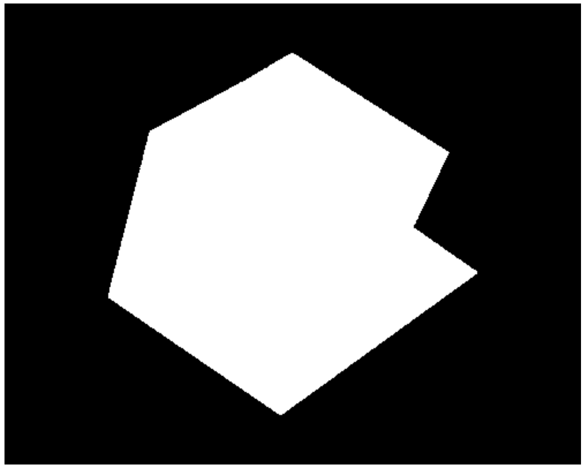
Global Segmentation

The object in this figure is light on a dark background. A histogram shows that the object and background are well-separated.

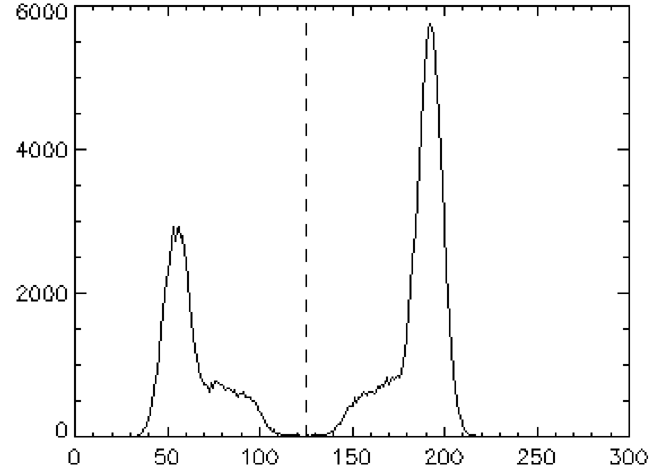




A binary image that is produced by thresholding at the level indicated by the dashed line is shown at the right.



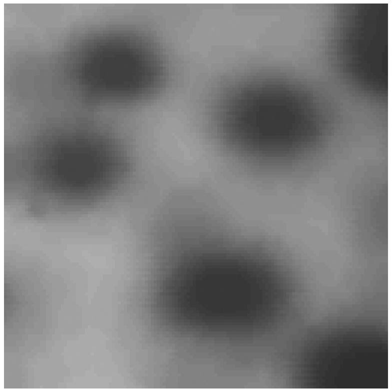
Global Segmentation – Example 1



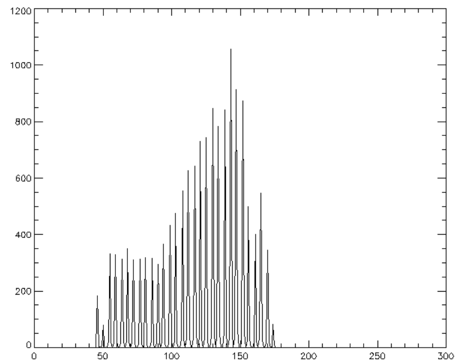
The effect of thresholding the fingerprint image at the threshold indicated is shown as the binary image to the right.

Global Segmentation – Example 2

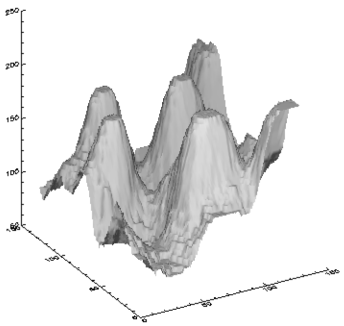
A greyscale image and its histogram is shown below. Selection of a threshold for segmentation is facilitated by examining a surface plot or a contour plot.



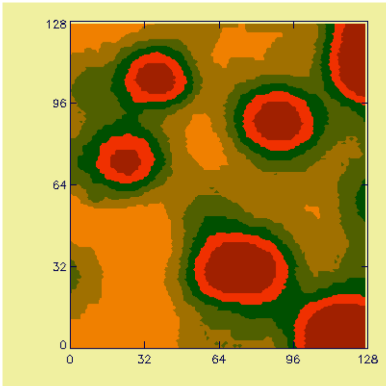
Image



Histogram



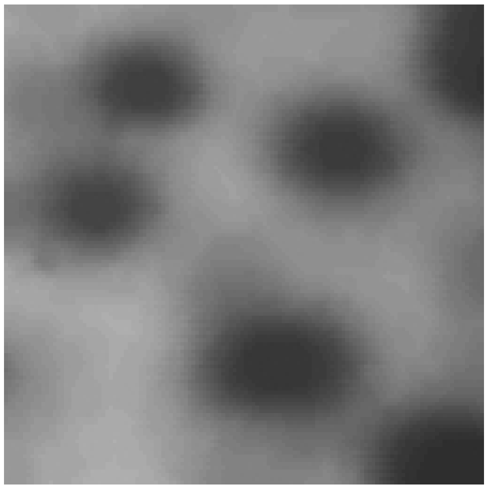
Inverted surface plot



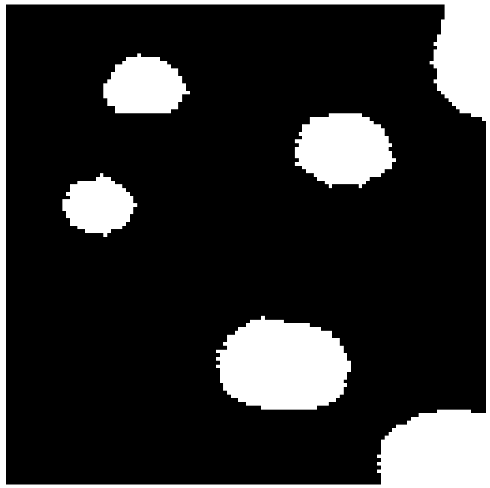
Contour plot

Global Segmentation – Example 2

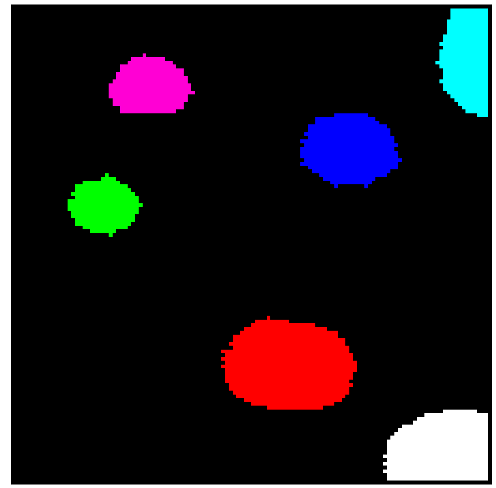
The set of pixels $S = \{p : A[p] \leq T\}$ with $T = 80$ are selected in the middle figure. The objects can be counted using `LABEL_REGION` as shown on the right.



Original



Segmented

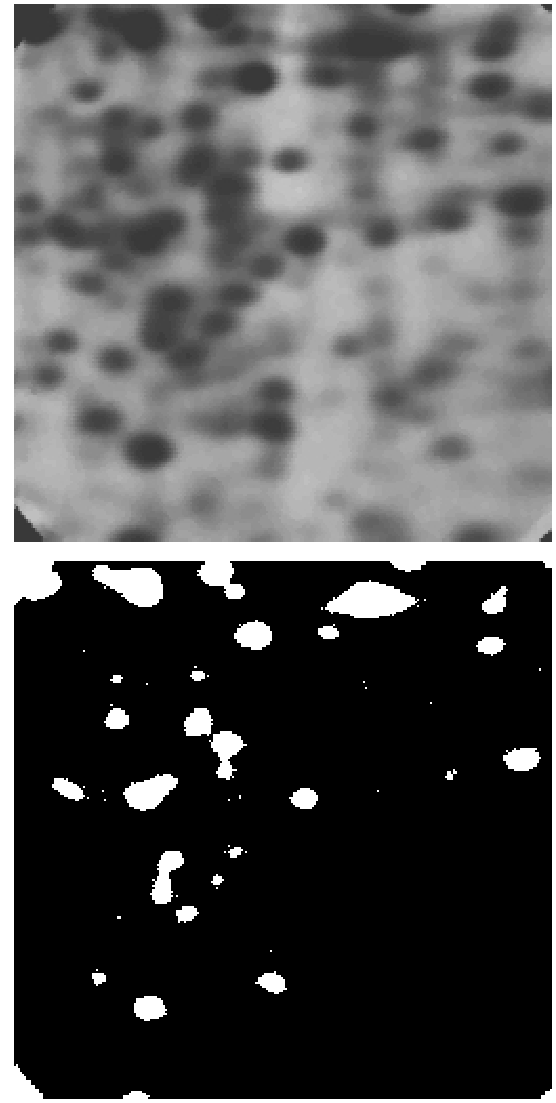


Labeled

Global Segmentation – Example 3

The upper figure is an electrophoresis image in which the dark areas are of interest.

Global thresholding with $T = 70$ produces the bottom image. Note that the objects in the lower right corner are missed. Other thresholds have similar difficulty, either being too high for the upper left corner or too low for the lower right corner.

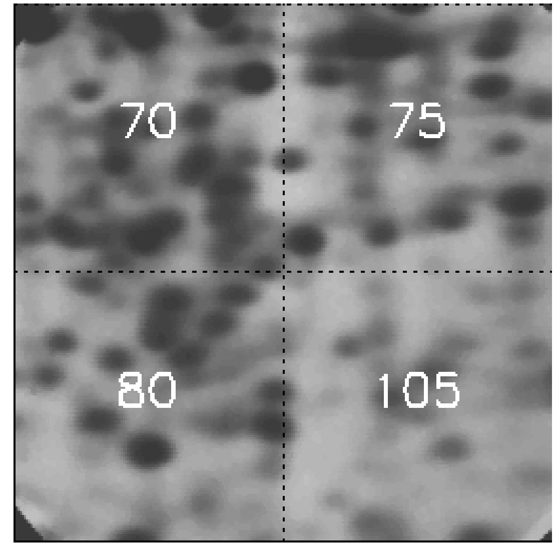


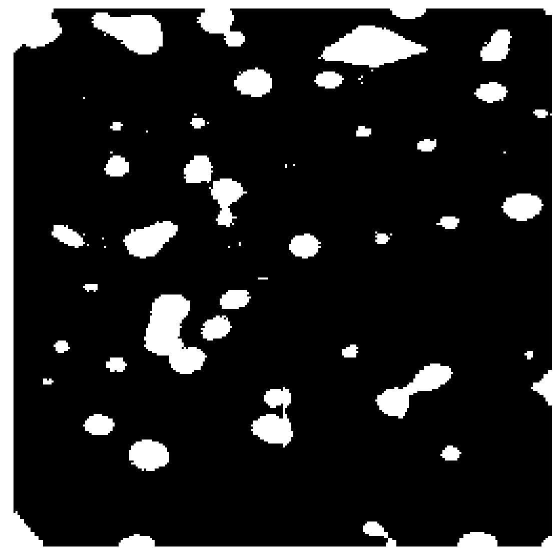
Global Segmentation – Example 3

The image was divided into four regions as shown in the upper image.

A different threshold was used for each region, as indicated on the figure. The result is shown in the lower image.

There was an improvement in the results because the threshold was adapted to the brightness of each region.





Global Segmentation – Example 3

It appears that the average brightness is a function of position. If we assume that the image is of the form

$$A[x,y] = F[x,y] + G[x,y]$$

where $F[x,y]$ is the image of interest and $G[x,y]$ is a varying background, then perhaps we can remove the baseline variation.

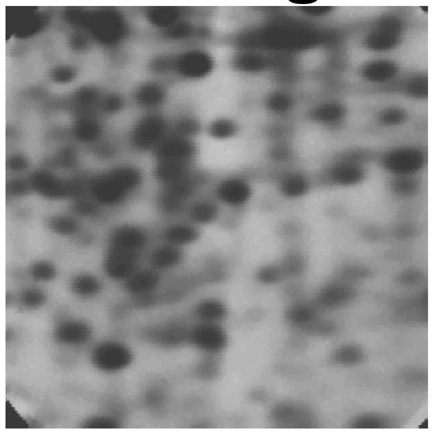
Let $G[x,y]$ be approximated by a surface of the form

$$G[x,y] = a_0 + a_1 x + a_2 y + a_3 xy$$

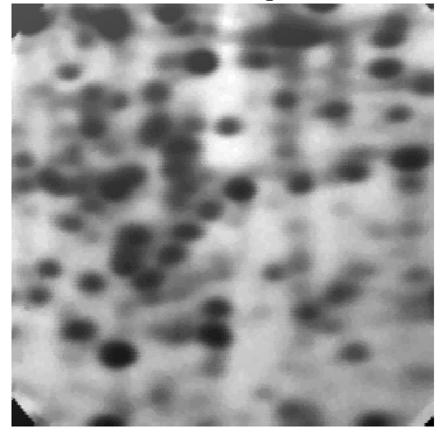
The coefficients can be estimated so that G is a least-mean-square approximation to A. For this image we find

a_0	a_1	a_2	a_3
142	0.06	-0.16	1e-5

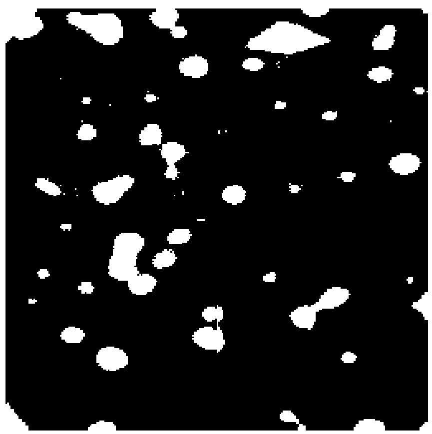
Global Segmentation – Example 3



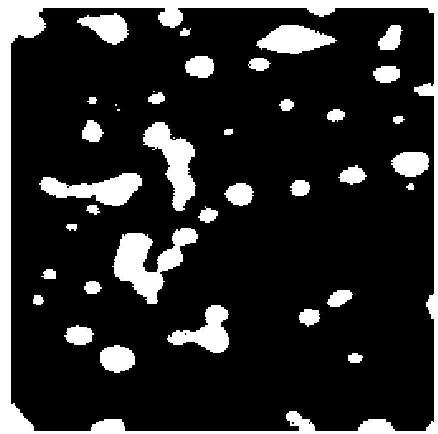
Original Image A



Compensated Image F



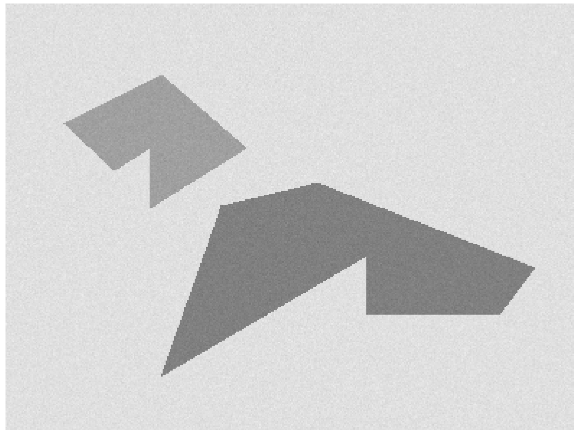
Region Thresholding of A

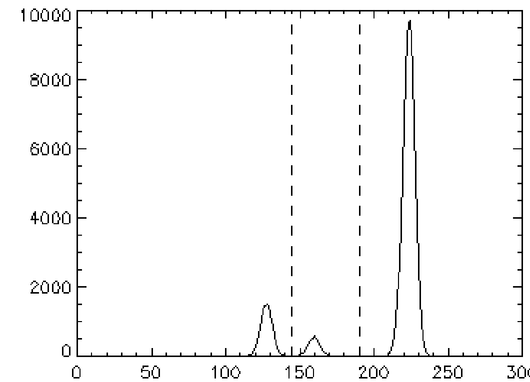


Global Threshold of F

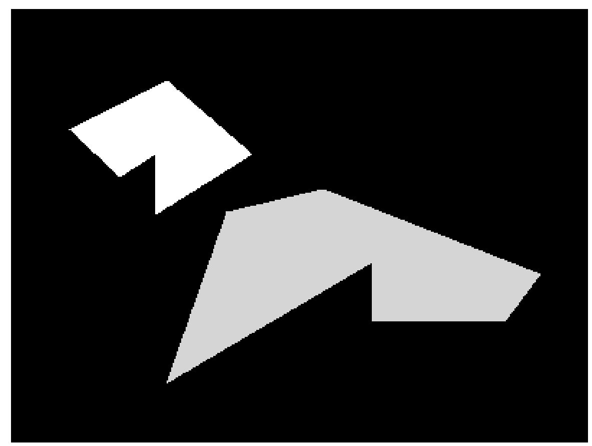
Region thresholding and global thresholding after compensation achieve similar results.

Global Segmentation with Multiple Thresholds





An image that is segmented by using two thresholds as indicated by the dashed lines is shown at the right. The brighter segment corresponds to the region between the thresholds.



Automated Threshold Setting

An algorithm can be made more flexible and more easily used if the threshold can be selected automatically. When global thresholding is appropriate, this can be done by a simple iterative algorithm.

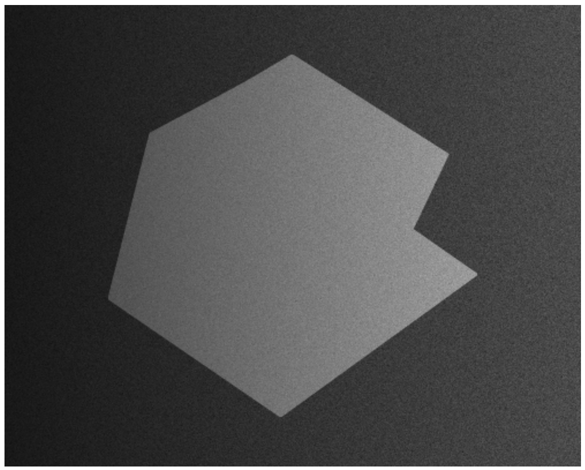
1. Select an estimate for T. This could be the mean grey value of the image.
2. Segment the image using T to produce two sets of pixels: G_1 containing pixels with grey values $> T$ and G_2 containing pixels with grey values $\leq T$.
3. Compute the average values μ_1 and μ_2 of sets G_1 and G_2.
4. Compute a new threshold value

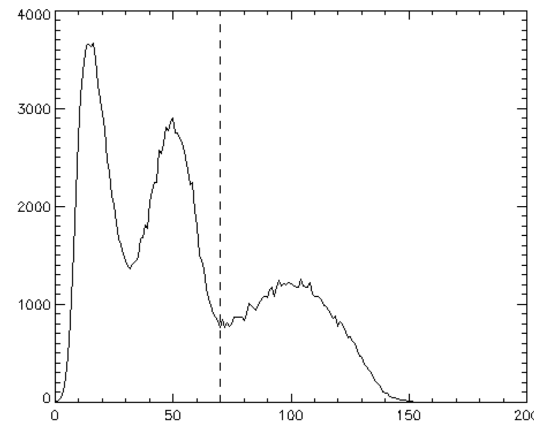
$$T = \frac{\mu_1 + \mu_2}{2}$$

5. Repeat steps 2 through 4 until changes in T are small.

Difficulty with Global Segmentation

A global threshold may fail if the conditions change across an image.



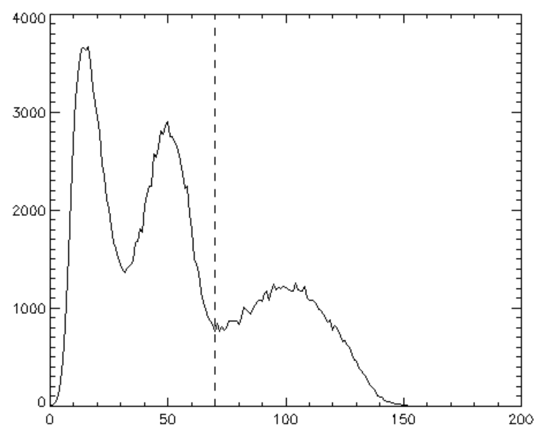


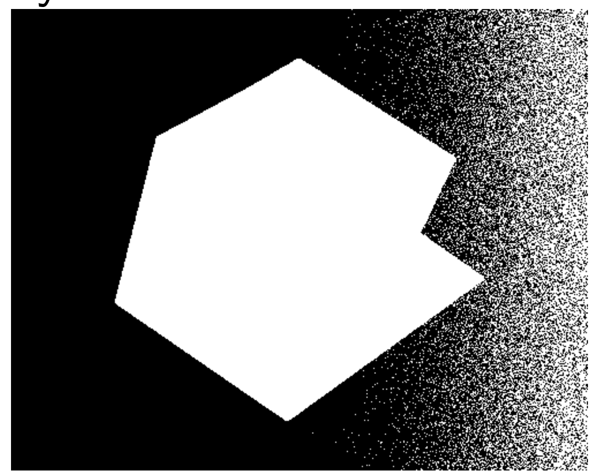
An image with nonuniform illumination, which increases linearly from left to right.

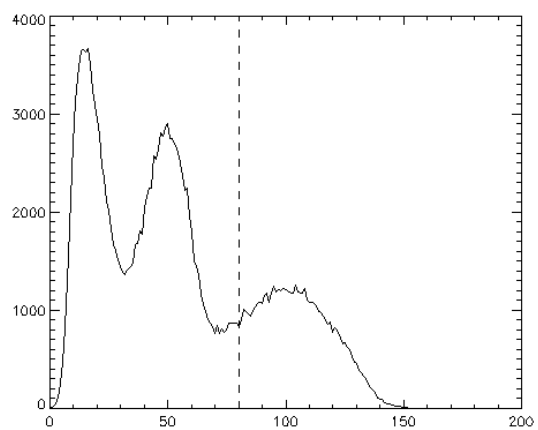
The image histogram shows that there is no suitable threshold for global segmentation.

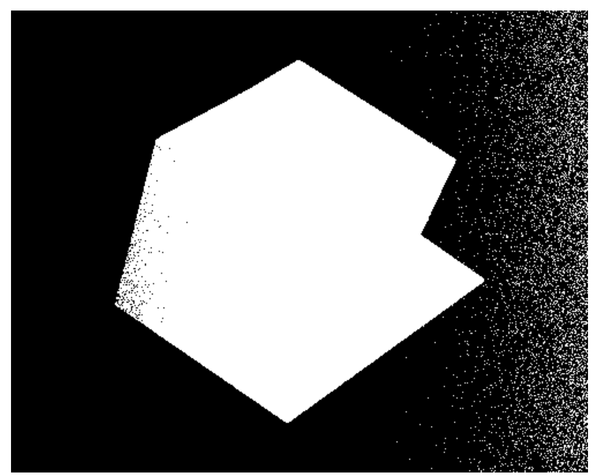
Difficulty with Global Segmentation

Shown below is the effect of segmentation with two similar threshold settings. Note that neither is satisfactory.



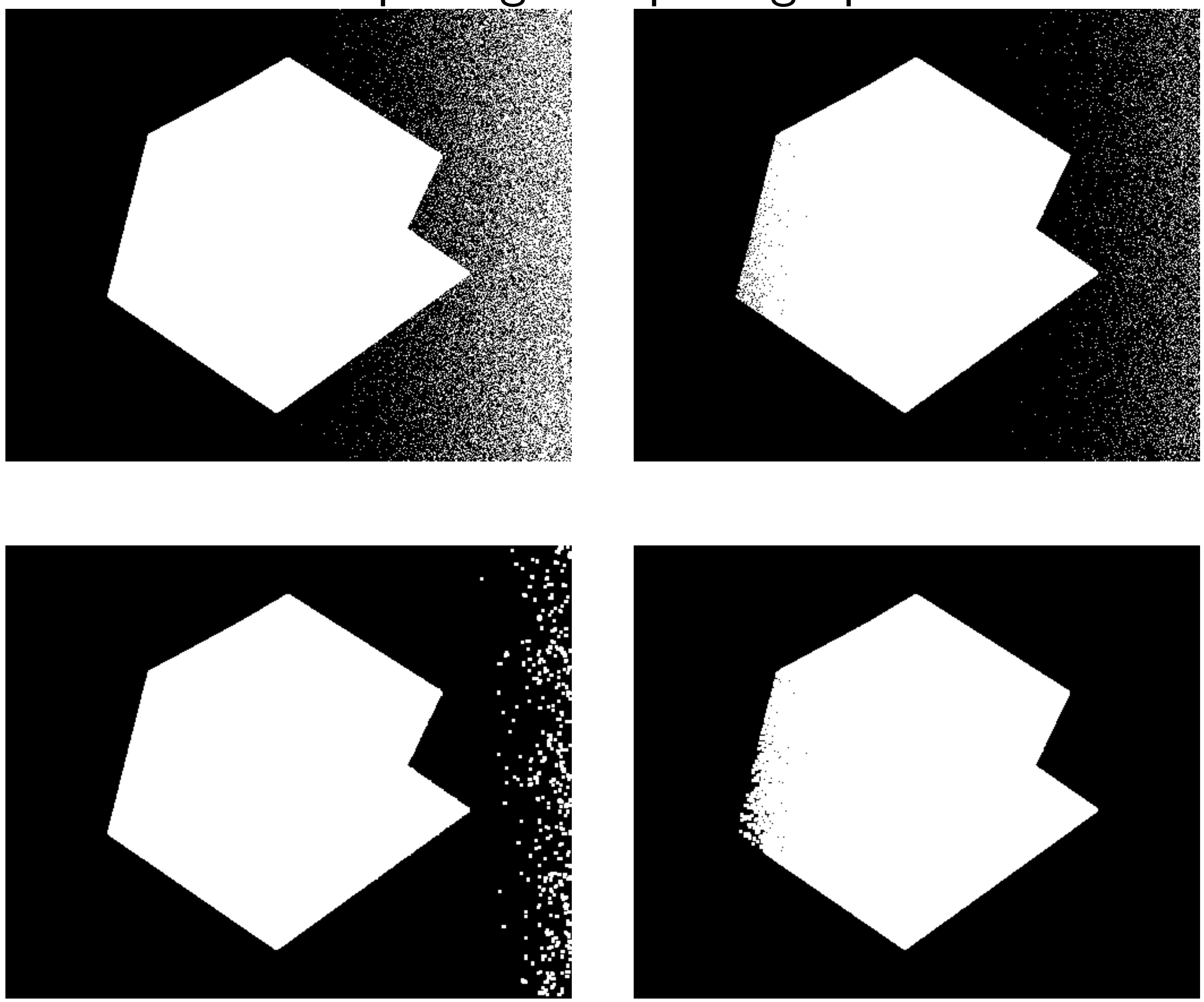






Difficulty with Global Segmentation

We can attempt to clean up the segmentation with other techniques. Shown below is the effect of a morphological opening operation.



$T = 70$ Image $\qquad$ $T = 80$ Image

Effect of Varying Illumination

An image can be considered to be the result of illumination and reflection.

$$f(x, y) = r(x, y)i(x, y)$$

The illumination is usually a slowly varying function compared to the scene reflectance.

If we know the illumination function then we can solve for the reflectance directly.

$$r(x, y) = \frac{f(x, y)}{i(x, y)}$$

The reflectance function can then be processed directly.

If we do not know $i(x, y)$ then we can try to determine it by analysis.

Solving for $i(x, y)$

Consider the function

$$\begin{aligned} z(x,y) &= \ln f(x,y) \\ &= \ln r(x,y) + \ln i(x,y) \\ &= r_l(x,y) + i_l(x,y) \end{aligned}$$

If $r(x, y)$ and $i(x, y)$ are independent random variables, which would be the usual case, then $r_l(x, y)$ and $i_l(x, y)$ are also statistically independent.

The probability density function of independent random variables is the convolution of the separate probability density functions. The probability density function of $i_l(x, y)$ is likely to be approximately $p_i(\xi) = \delta(\xi - \xi_0)$. Convolution with the probability density function of the reflectance, $p_r(\xi)$, will leave the shape essentially unchanged.

Therefore, if $r(x, y)$ can be segmented by global thresholding under constant illumination, we expect that $r_l(x, y)$ can be segmented by global thresholding

Solving for $i(x, y)$

We will approximate $i_l(x, y)$ with a slowly varying function.

$$\hat{i}_l(x, y) = a_0 + a_1 x + a_2 y + a_3 xy$$

We will choose the coefficients to minimize the mean-squared difference between $z(x, y)$ and $\hat{i}_l(x, y)$.

Let

$$e(x, y) = z(x, y) - \hat{i}_l(x, y) = z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy)$$

and let the mean-squared error be denoted by

$$\begin{aligned} \langle e^2 \rangle &= \frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} e^2(x, y) \\ &= \frac{1}{MN} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \left(z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy)\right)^2 \end{aligned}$$

Solving for $i(x, y)$

Differentiate with respect to the coefficients a_k, $k = 0, 1, 2, 3$ and set each equation equal to zero. We will use the bracket notation for the double sums to simplify the appearance of the equations.

$$\frac{\partial \langle e^2 \rangle}{a_0} = 2\langle (z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy)) \rangle = 0$$
$$\frac{\partial \langle e^2 \rangle}{a_1} = 2\langle (z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy))\, x \rangle = 0$$
$$\frac{\partial \langle e^2 \rangle}{a_2} = 2\langle (z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy))\, y \rangle = 0$$
$$\frac{\partial \langle e^2 \rangle}{a_3} = 2\langle (z(x, y) - (a_0 + a_1 x + a_2 y + a_3 xy))\, xy \rangle = 0$$

Solving for $i(x, y)$

The equations can be simplified to

$$
\begin{aligned}
a_0 + a_1\langle x\rangle + a_2\langle y\rangle + a_3\langle xy\rangle &= \langle z(x,y)\rangle \\
a_0\langle x\rangle + a_1\langle x^2\rangle + a_2\langle xy\rangle + a_3\langle x^2y\rangle &= \langle xz(x,y)\rangle \\
a_0\langle y\rangle + a_1\langle xy\rangle + a_2\langle y^2\rangle + a_3\langle xy^2\rangle &= \langle yz(x,y)\rangle \\
a_0\langle xy\rangle + a_1\langle x^2y\rangle + a_2\langle xy^2\rangle + a_3\langle x^2y^2\rangle &= \langle xyz(x,y)\rangle
\end{aligned}
$$

All of the bracketed terms can be computed from the image information. We can then solve for the coefficients in this set of linear equations. The matrix form is

$$
\begin{bmatrix} 1 & \langle x\rangle & \langle y\rangle & \langle xy\rangle \\ \langle x\rangle & \langle x^2\rangle & \langle xy\rangle & \langle x^2y\rangle \\ \langle y\rangle & \langle xy\rangle & \langle y^2\rangle & \langle xy^2\rangle \\ \langle xy\rangle & \langle x^2y\rangle & \langle xy^2\rangle & \langle x^2y^2\rangle \end{bmatrix}
\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}
=
\begin{bmatrix} \langle z\rangle \\ \langle xz\rangle \\ \langle yz\rangle \\ \langle xyz\rangle \end{bmatrix}
$$

$$
\mathbf{Ca} = \mathbf{v} \Rightarrow \mathbf{a} = \mathbf{C}^{-1}\mathbf{v}
$$

Solving for $i(x, y)$

The terms in the matrix are based only on the image coordinates while those on the right depend upon both the image and the coordinates.

```
;Open the image
F=read_image('D:\Harvey\Classes\SIMG782\2004\lectures\lecture_06\doc\fig7.png')
sf=size(F,/dim)

;Construct coordinate arrays for the image
x=findgen(sf[0])#replicate(1,sf[1])
y=findgen(sf[1])##replicate(1,sf[0])

;Compute the log image
FL=BYTSCL(ALOG(F))

;Construct the moments used to solve for the a coefficients
mx=mean(x) & my=mean(y) & mxy=mean(x*y)
mx2=mean(x^2) & my2=mean(y^2) & mx2y=mean(x^2*y)
mxy2=mean(x*y^2) & mx2y2=mean(x^2*y^2)
C=[[1,mx,my,mxy],[mx,mx2,mxy,mx2y],[my,mxy,my2,mxy2],[mxy,mx2y,mxy2,mx2y2]]
CI=invert(C)
```

Solving for $i(x, y)$

```
;Construct an illumination predictor for FL mL=mean(FL) &
mLx=mean(FL*x) & mLy=mean(FL*y) & mLxy=mean(FL*x*y)
v=transpose([mL,mLx,mLy,mLxy])
aL=CI##v
GL=aL[0]+aL[1]*x+aL[2]*y+aL[3]*x*y
```

For this image, which is of size 500×400 we find

$$\mathbf{a} = \begin{bmatrix} 1.0 & 249.5 & 199.5 & 49775.3 \\ 249.5 & 83083.5 & 49775.3 & 16575158.0 \\ 199.5 & 49775.3 & 53133.5 & 13256808.0 \\ 49775.3 & 16575158.0 & 13256808.0 & 4414517248.0 \end{bmatrix}^{-1} \begin{bmatrix} 147.7 \\ 41253.9 \\ 29511.4 \\ 8245948.0 \end{bmatrix}$$

$$= \begin{bmatrix} 95.227890 \\ 0.207723 \\ -0.001120 \\ 0.000018 \end{bmatrix}$$

Solving for $i(x, y)$

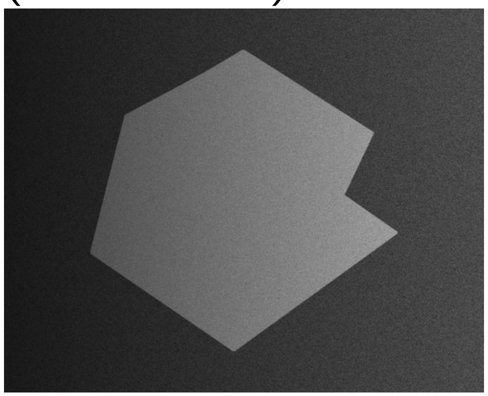
Having computed the coefficients, the rest is now easy.

```
;Illumination approximation
GL=aL[0]+aL[1]*x+aL[2]*y+aL[3]*x*y

;Compute the log reflectance
RL=bytscl(FL-GL)

;Compute the log reflectance histogram
hr=histogram(RL)

;Threshold the log reflectance image
TL=175
RLT=RL GE TL
```

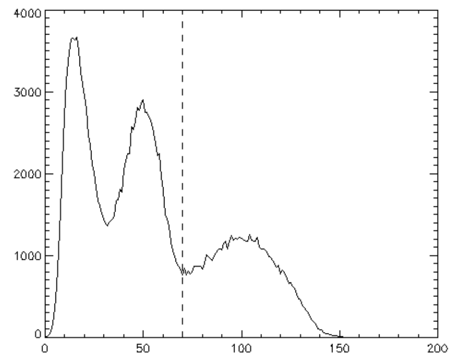
The results are illustrated on the following pages.

Segmentation Example

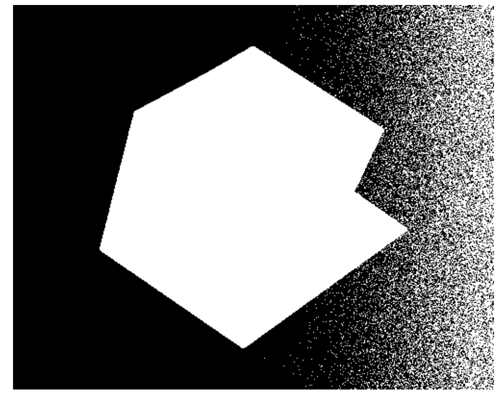
The figure below has an illumination gradient from left to right across the image. As shown by its histogram, it is difficult to segment. The log(reflectance) has a separable histogram.



Original

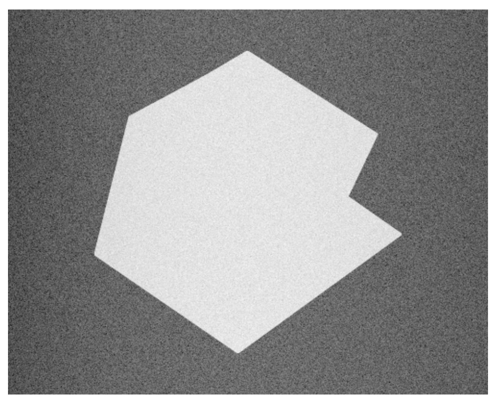


Histogram of Original

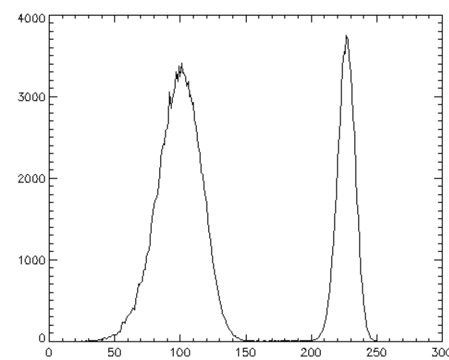


Segmentation

Log(Illumination)



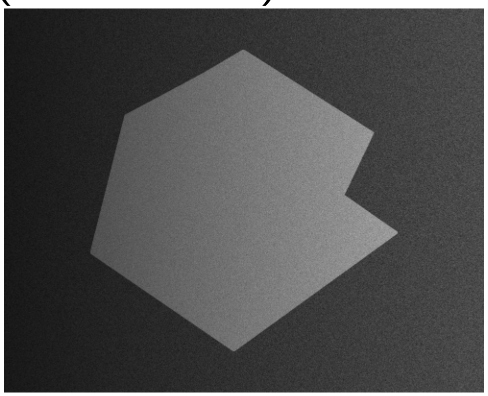
Log(Reflectance)



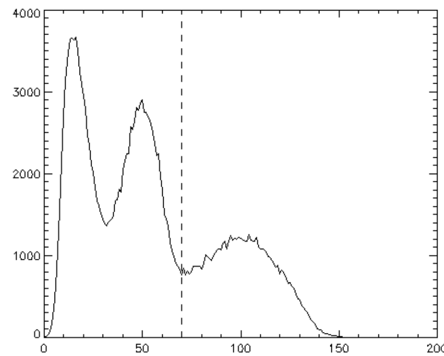
Histogram

Segmentation Example

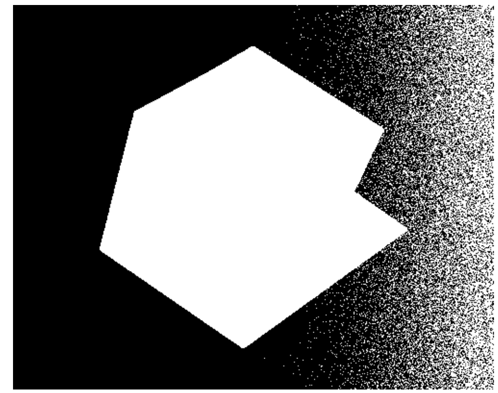
The figure below has an illumination gradient from left to right across the image. As shown by its histogram, it is difficult to segment. The log(reflectance) has a separable histogram.



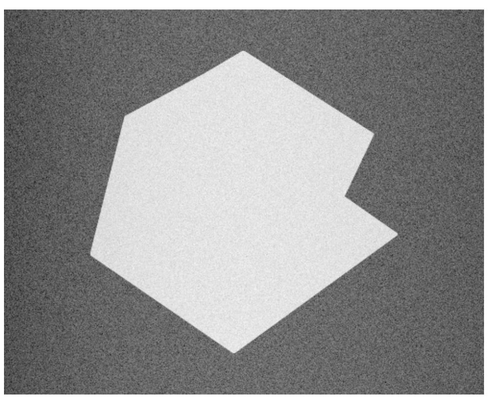
Original



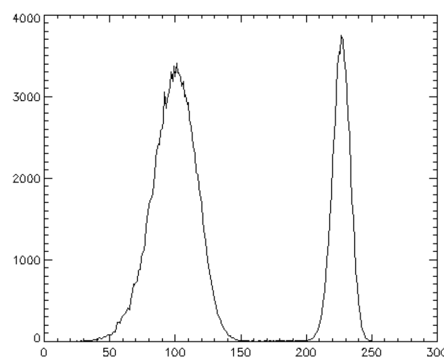
Histogram



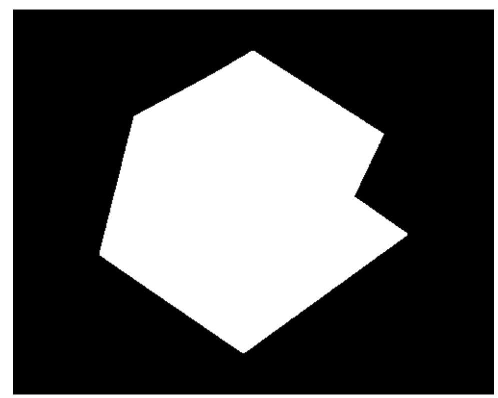
Segmentation



Log(Reflectance)



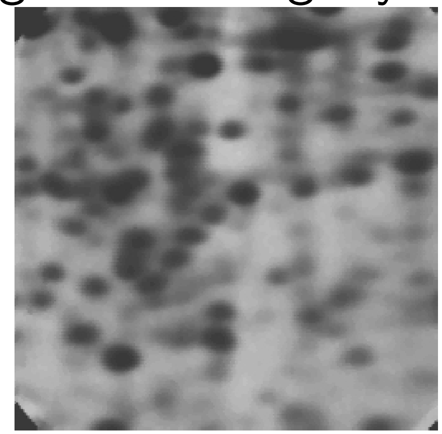
Histogram



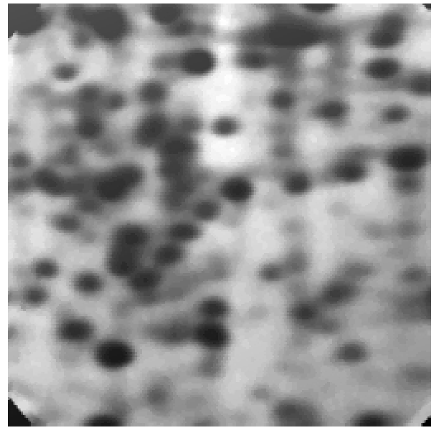
Segmentation

Example 3 – Continued

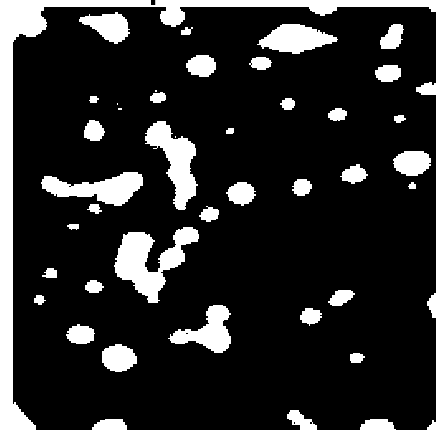
Removal of illumination trend by compensation of the log image has slightly better results for the electrophoresis image.



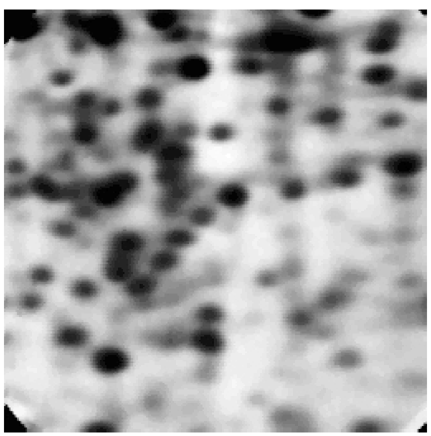
Original Image



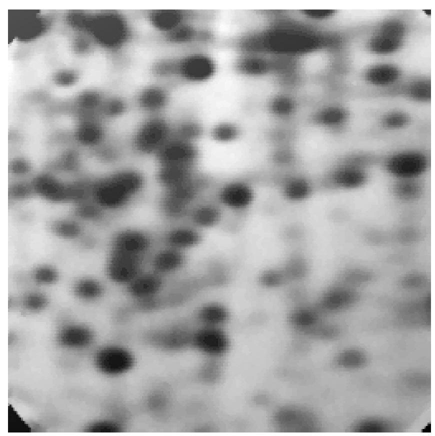
Illumination Subtraction



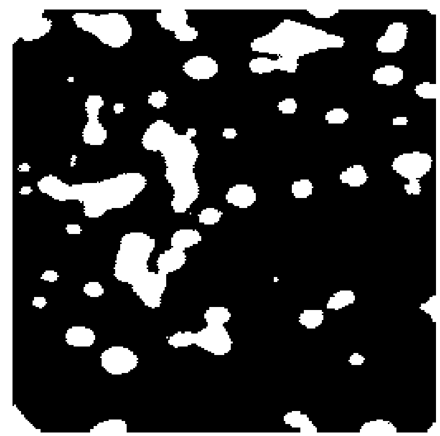
Global Threshold



Log Image



Illumination Division



Global Threshold

Morphological Waterfall Segmentation

Watershed segmentation is a metaphor for a computational method that iteratively fills the local minima of a greyscale image to find segment boundaries.

As regions are filled from the bottom, they reach a level where they would merge. At this point "dams" are constructed to maintain the separation.

The "dams" become the segment boundaries.

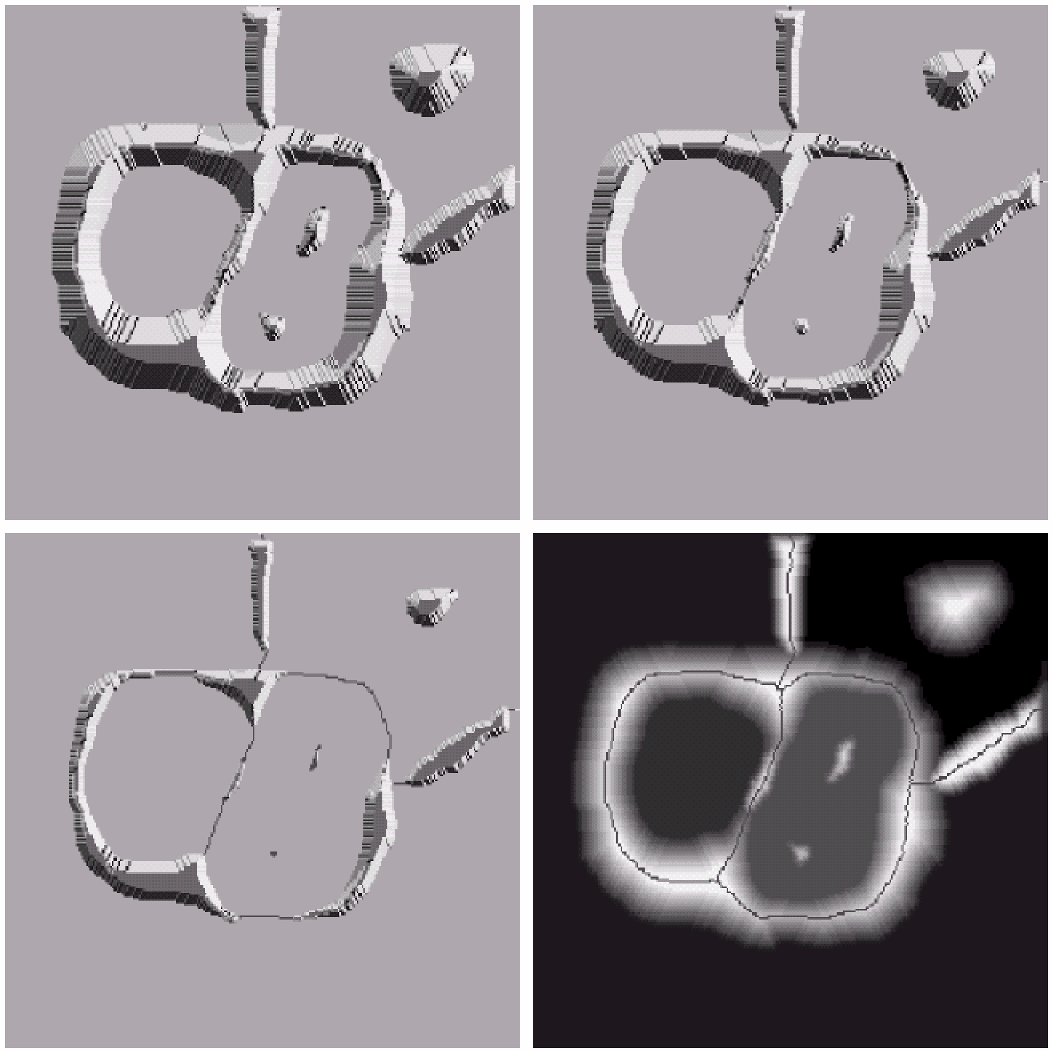
The iterative erosion process is illustrated in the figures below (Gonzalez & Woods, Section 10.5).

Chapter 10
Image Segmentation

a b
c d

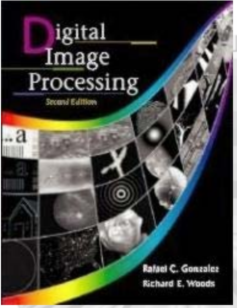
FIGURE 10.44 (a) Original image. (b) Topographic view. (c)–(d) Two stages of flooding.

Chapter 10
Image Segmentation

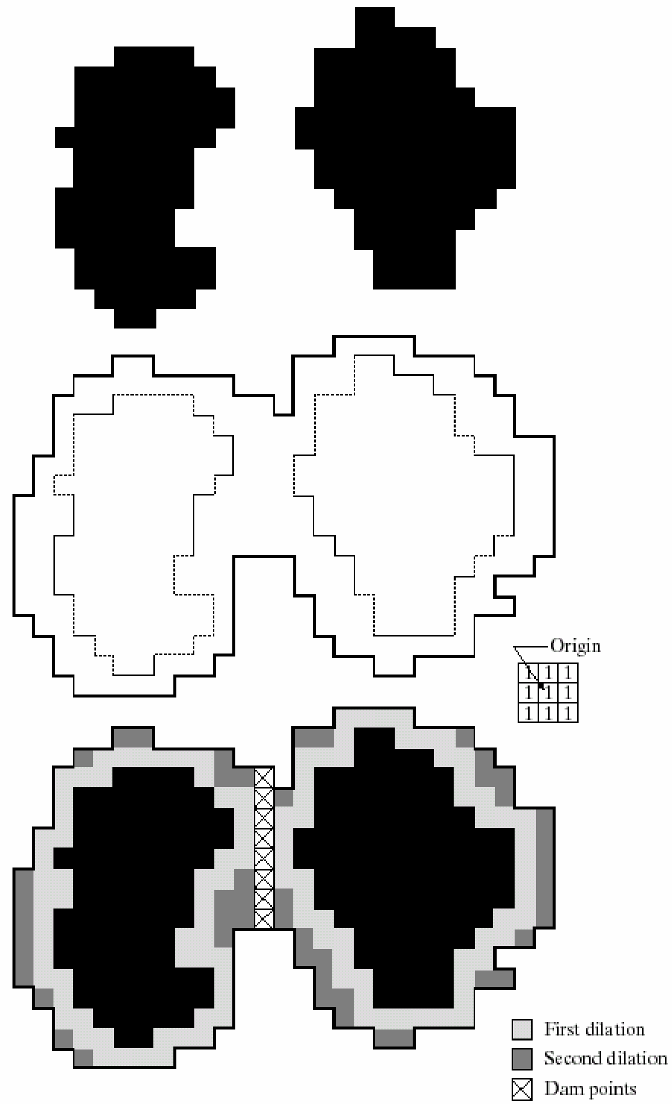


e f
g h

FIGURE 10.44 *(Continued)* (e) Result of further flooding. (f) Beginning of merging of water from two catchment basins (a short dam was built between them). (g) Longer dams. (h) Final watershed (segmentation) lines. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

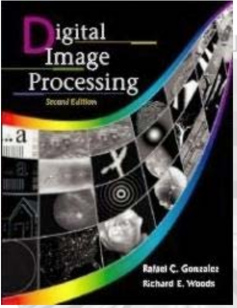


Chapter 10
Image Segmentation

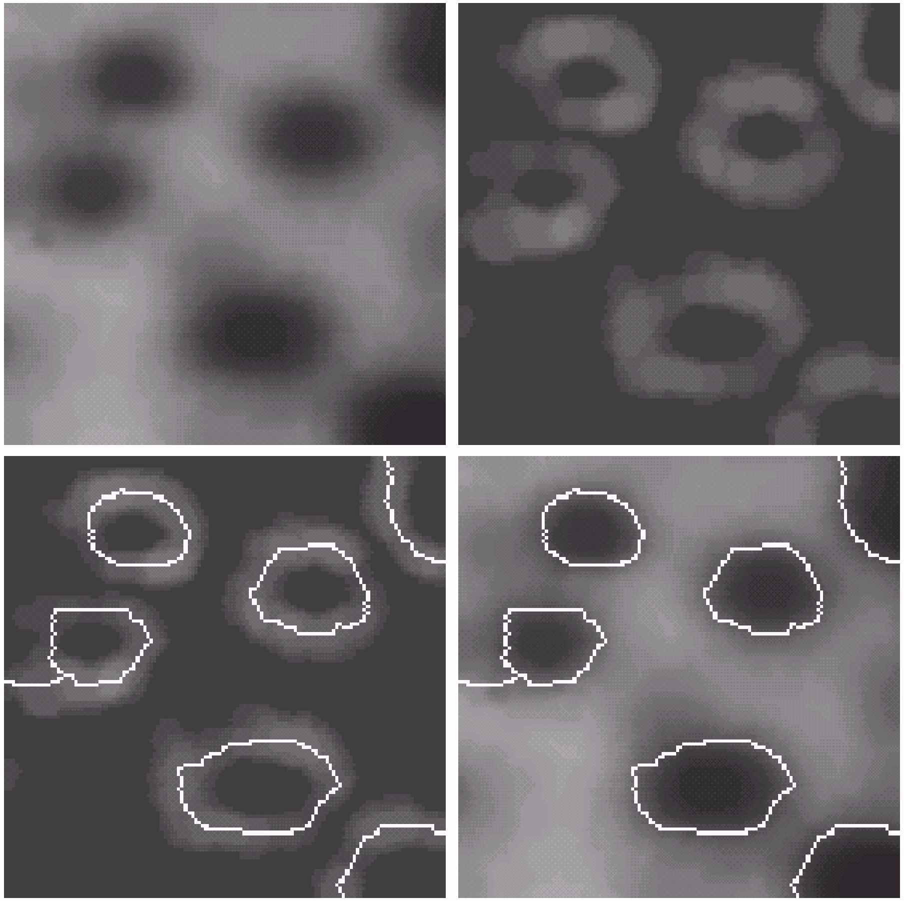


a
b c
d

FIGURE 10.45 (a) Two partially flooded catchment basins at stage $n-1$ of flooding. (b) Flooding at stage n, showing that water has spilled between basins (for clarity, water is shown in white rather than black). (c) Structuring element used for dilation. (d) Result of dilation and dam construction.



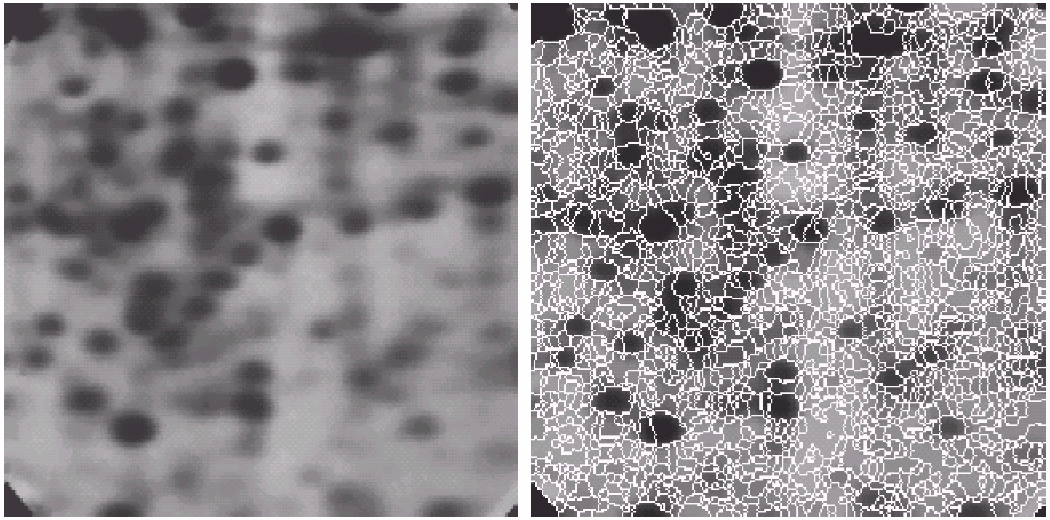
Chapter 10
Image Segmentation



a b
c d

FIGURE 10.46 (a) Image of blobs. (b) Image gradient. (c) Watershed lines. (d) Watershed lines superimposed on original image. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

Chapter 10
Image Segmentation

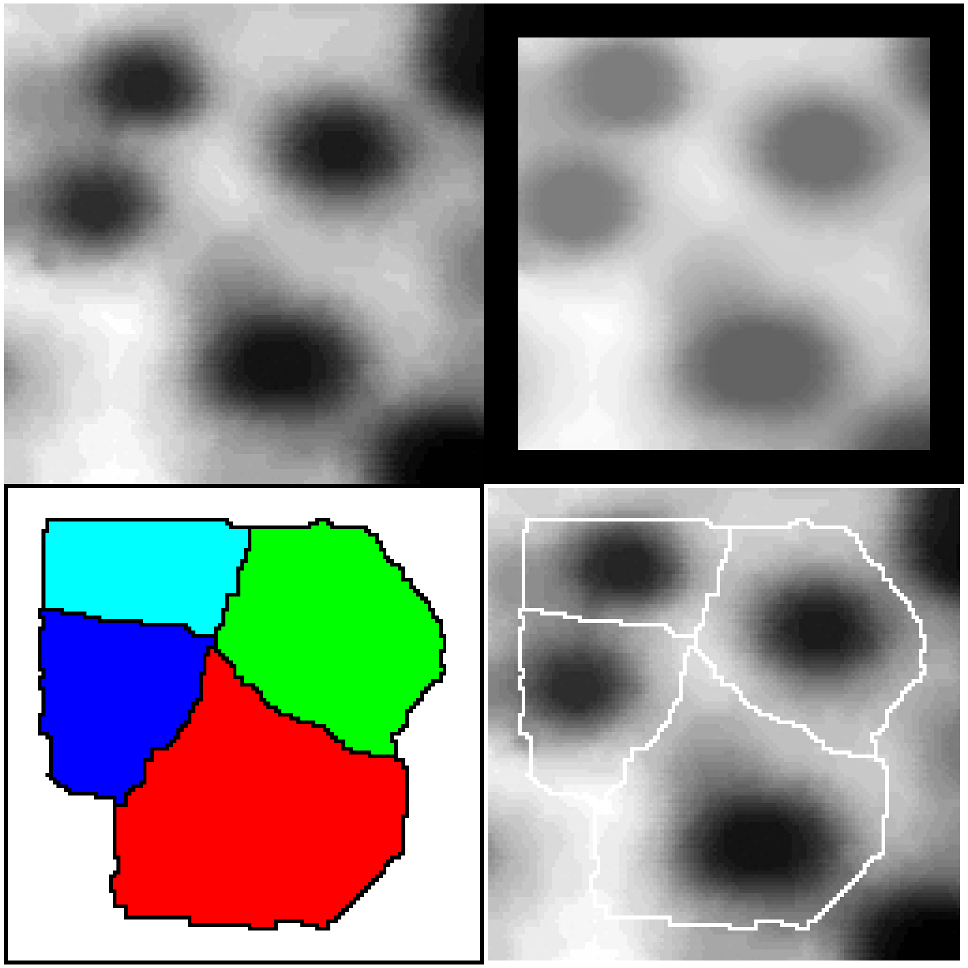


a b

FIGURE 10.47 (a) Electrophoresis image. (b) Result of applying the watershed segmentation algorithm to the gradient image. Oversegmentation is evident. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

Watershed Segmentation – Example 1

The small greyscale image of smooth black blobs is a natural test candidate for watershed segmentation. This image was segmented using the IDL watershed.pro function. Code shown on next page.



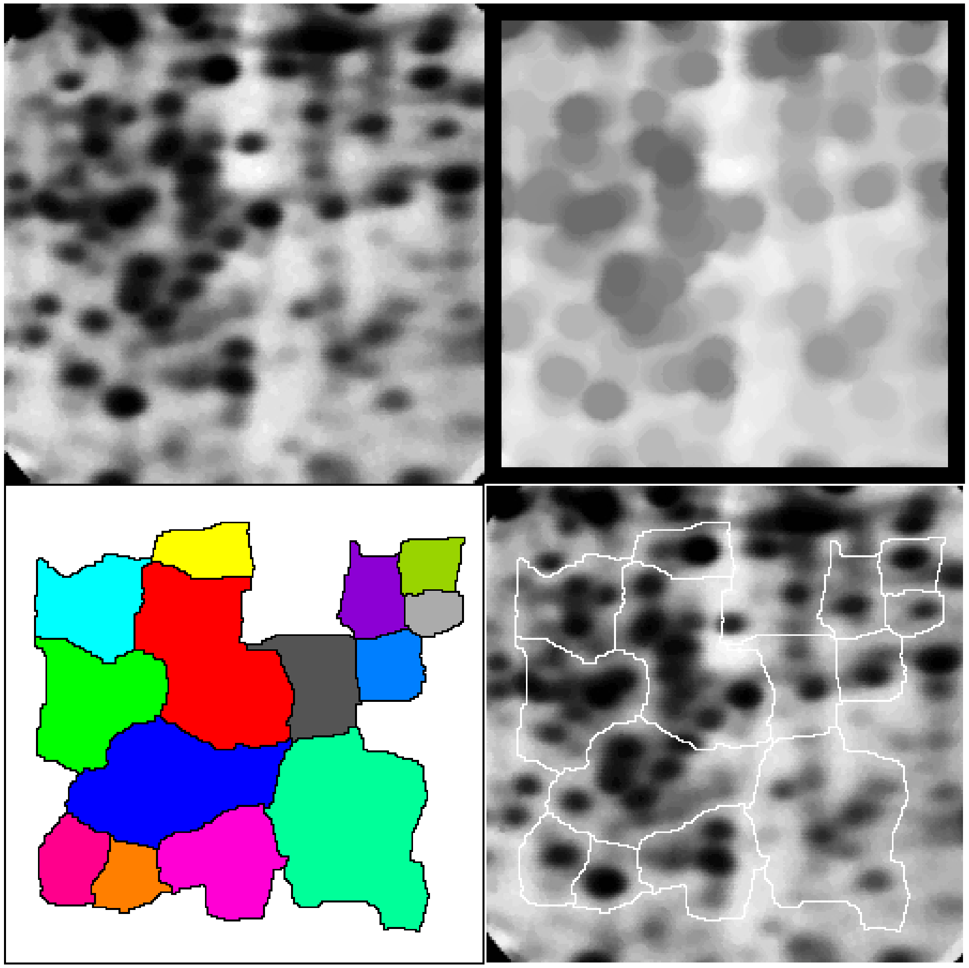
Original Image	After morphological greyscale erosion with disk of radius 9
Watershed regions	Boundaries overlaid on original image.

Watershed Demo Code

```
;Watershed Demo
;Radius of disc for greyscale erosion
r = 9

;Create a disc of radius r
disc = SHIFT(DIST(2*r+1), r, r) LE r

B=READ_IMAGE('doc/fig11a.png')
sa=size(B,/dim)
window,/free,xs=2*sa[0],ys=2*sa[1]

;window,/free,xs=sb[0],ys=sb[1],xpos=0,ypos=0
TVSCL, b,0

;Remove holes of radii less than r
c = MORPH_CLOSE(b, disc, /GRAY)
TVSCL, c,1

;Create watershed image
d = WATERSHED(c,connectivity=8)

;Display it, showing the watershed regions
tek_color
TV, d,2
loadct,0,/silent

;Merge original image with boundaries of watershed regions
e = a > (MAX(a) * (d EQ 0b))
TVSCL, e,3
```

Watershed Segmentation – Example 2

Revisit segmentation of electrophoresis image



Watershed Segmentation – Example 3

Segmentation of pollen image from IDL demonstration package.