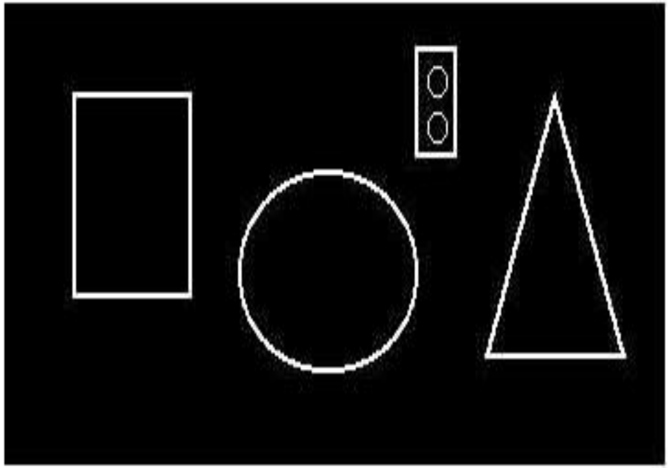
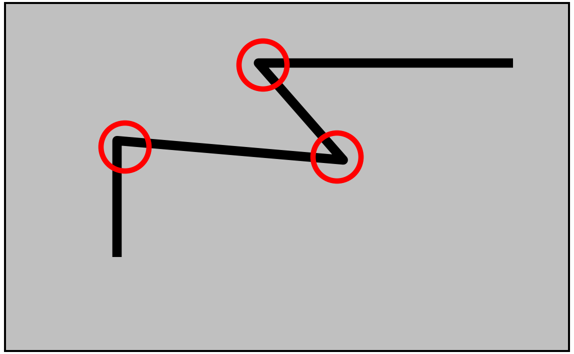
# Hough Transform Generalizations

- It locates straight lines (SHT) - standard, simple HT
- It locates straight line intervals
- It locates circles
- It locates algebraic curves
- It locates arbitrary specific shapes in an image
  - **But you pay progressively for complexity of shapes by time and memory usage**



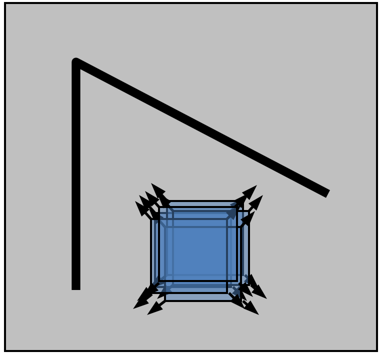
# Harris corner detector

- C.Harris, M.Stephens. “A Combined Corner and Edge Detector”. 1988

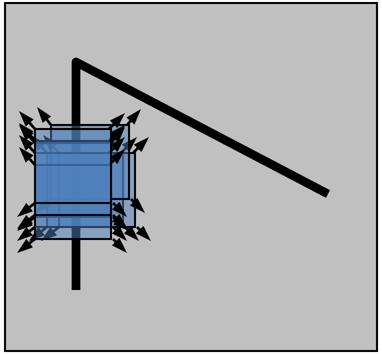


*from Rick Szeliski’s lecture notes, CSE576, Spring 05*

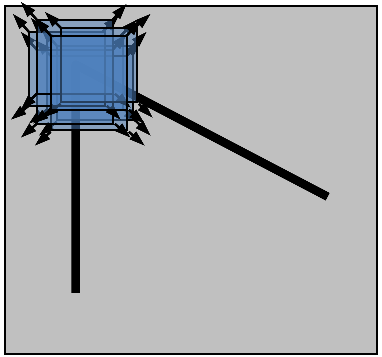
# Harris Detector: Basic Idea



“flat” region:
no change in
all directions



“edge”:
no change along
the edge direction



“corner”:
significant
change in all
directions

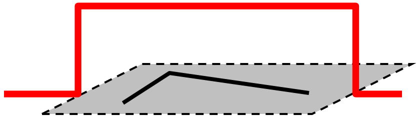
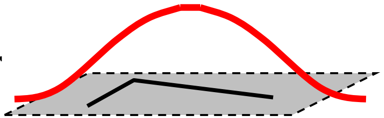
# Harris Detector: Mathematics

Change of intensity for the shift $[u,v]$:

Window function

Shifted intensity

Intensity

Window function $w(x,y)$ =  or 

1 in window, 0 outside

Gaussian

# Harris Detector: Mathematics

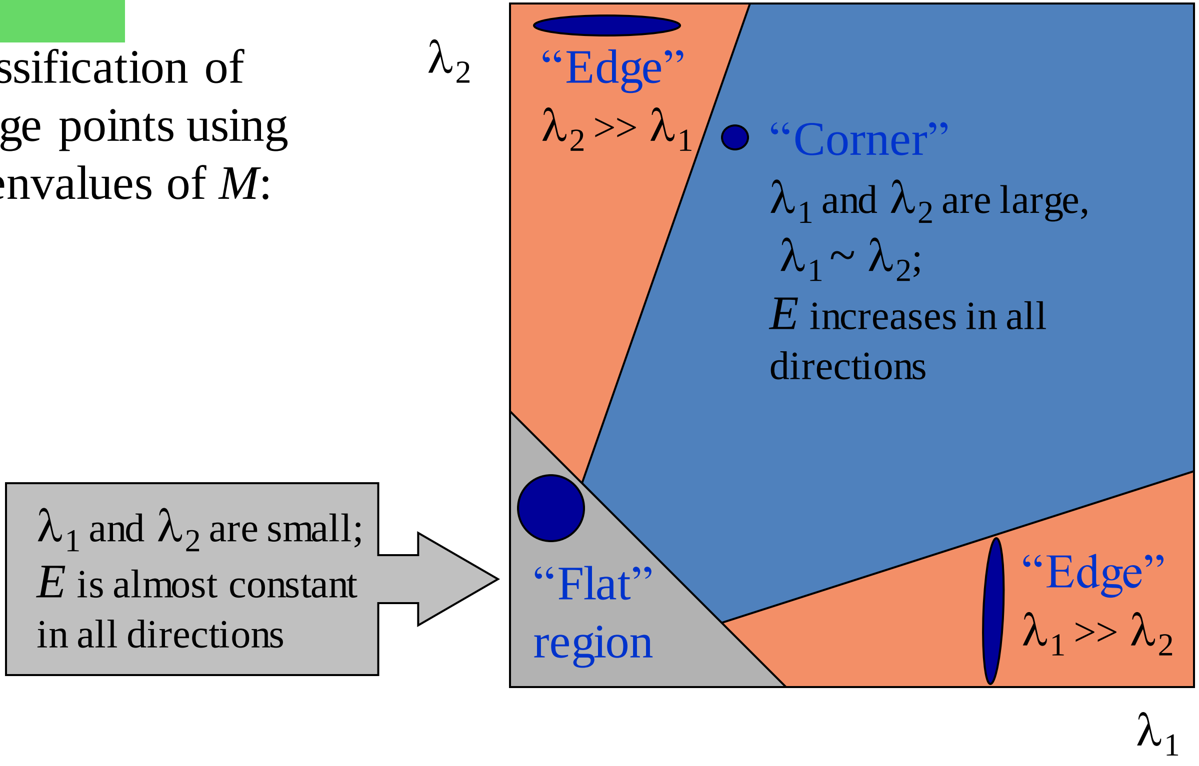
# Harris Detector: Mathematics

For small shifts $[u,v]$ we have a *bilinear* approximation:

where $M$ is a $2\times2$ matrix computed from image derivatives:

# Harris Detector: Mathematics

Classification of image points using eigenvalues of $M$:

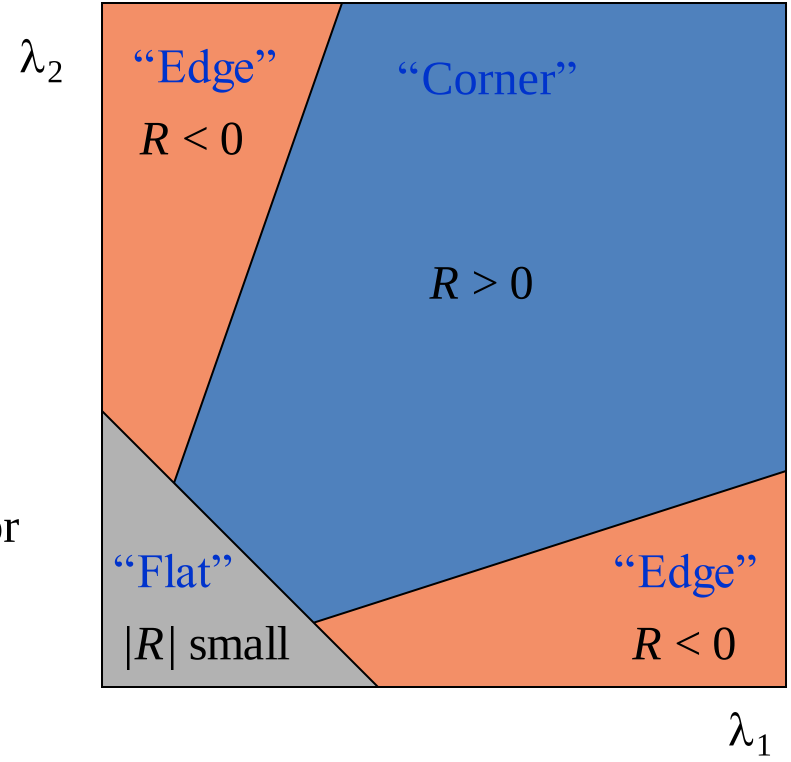


# Harris Detector: Mathematics

Measure of corner response:

($k$ – empirical constant, $k$ = 0.04-0.06)

# Harris Detector: Mathematics




- $R$ depends only on eigenvalues of M
- $R$ is large for a corner
- $R$ is negative with large magnitude for an edge
- $|R|$ is small for a flat region

# Harris Detector

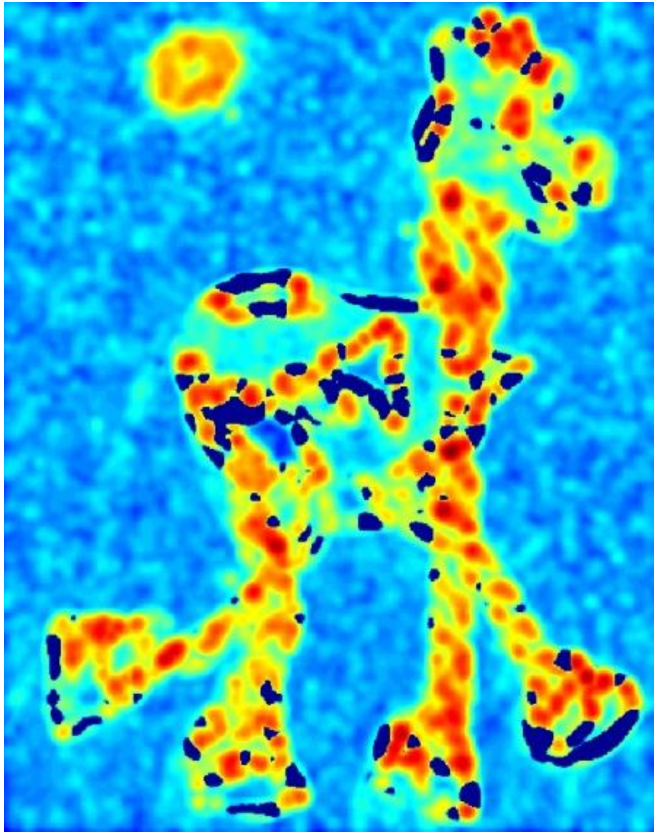
- The Algorithm:
  - Find points with large corner response function $R$ ($R$ > threshold)
  - Take the points of local maxima of $R$

# Harris Detector: Workflow

- Smooth image first !
  - Detection involves 1st and 2nd derivatives
  - Smoothing reduces effect of noise on the gradient maps

# Harris Detector: Workflow

Compute corner response $R$



# Harris Detector: Workflow

Find points with large corner response: $R$>threshold

# Harris Detector: Workflow

Take only the points of local maxima of $R$

# Harris Detector: Workflow

# Harris Detector: Summary

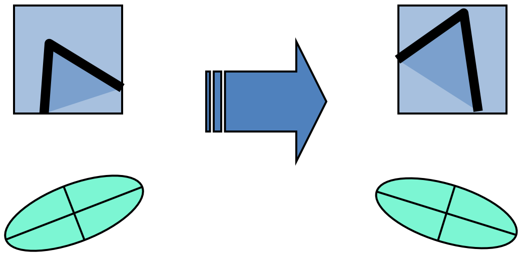
- Average intensity change in direction $[u,v]$ can be expressed as a bilinear form:

- Describe a point in terms of eigenvalues of $M$:
*measure of corner response*

- A good (corner) point should have a *large intensity change* in *all directions*, i.e. $R$ should be large positive

# Harris Detector: Some Properties

- Rotation invariance



Ellipse rotates but its shape (i.e. eigenvalues) remains the same

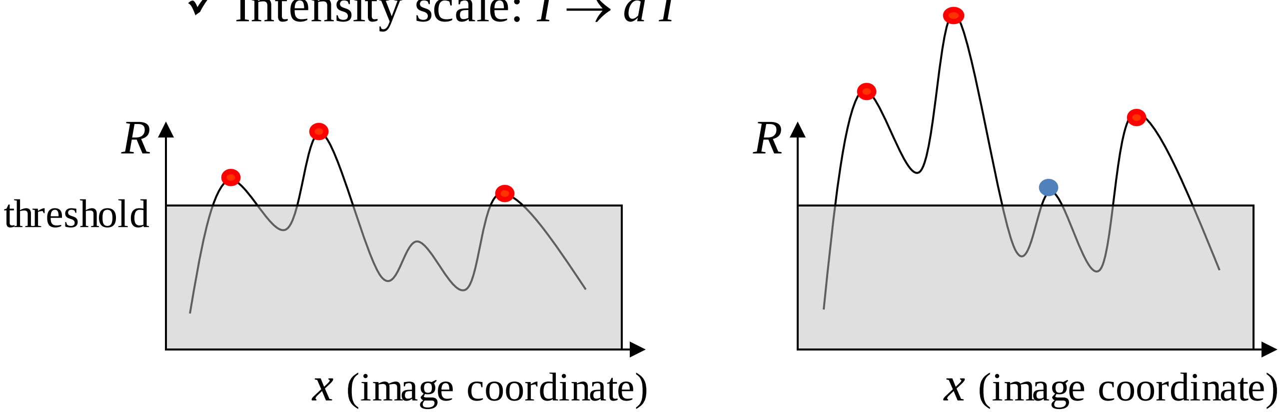
*Corner response $R$ is invariant to image rotation*

# Harris Detector: Some Properties

- Partial invariance to *affine intensity* change

✓ Only derivatives are used => invariance to intensity shift $I \rightarrow I + b$

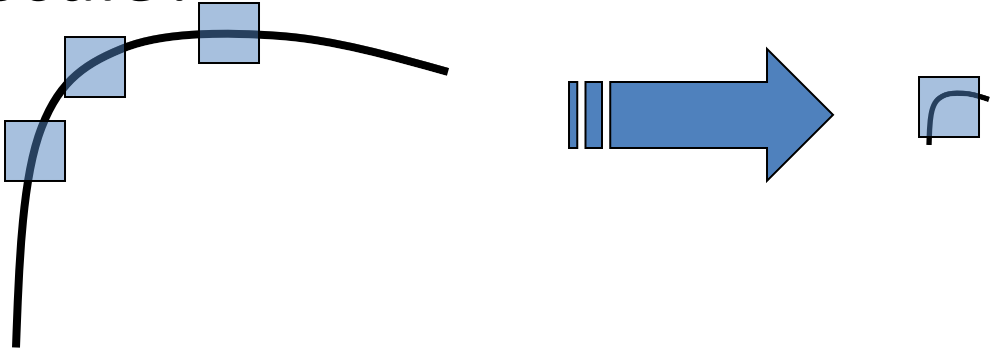
✓ Intensity scale: $I \rightarrow a\, I$



# Application: Finding corresponding points

# Harris Detector: Some Properties

- But: non-invariant to *image scale*!



All points will be classified as edges

Corner !

02.11.2021

# Digital Image Processing (CSE/ECE 478)

# Lecture-18: Image Segmentation (contd.)

Ravi Kiran

Sudipta Banerjee

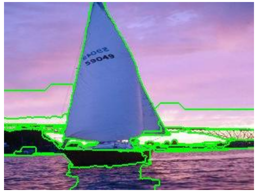
Center for Visual Information Technology (CVIT), IIIT Hyderabad

*Many slides borrowed from Vineet Gandhi @CVIT!*

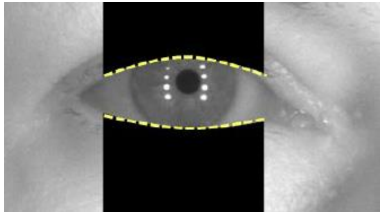
# Image Segmentation

Partitioning an image into a collection of connected sets of pixels.

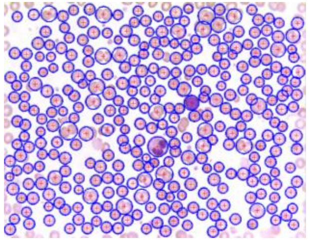
1. into regions, which usually cover the image




2. into linear structures, such as
   - line segments
   - curve segments



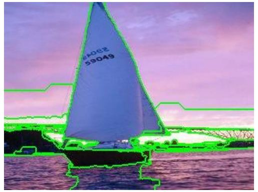
3. into 2D shapes, such as
   - circles
   - ellipses
   - ribbons (long, symmetric regions)



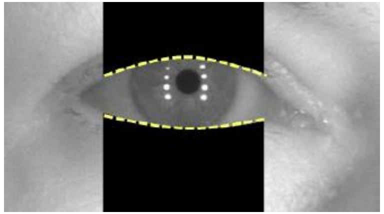
# Image Segmentation

Partitioning an image into a collection of connected sets of pixels.

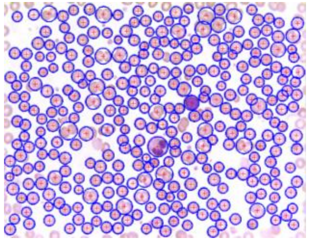
1. into regions, which usually cover the image




2. into linear structures, such as
   - line segments
   - curve segments



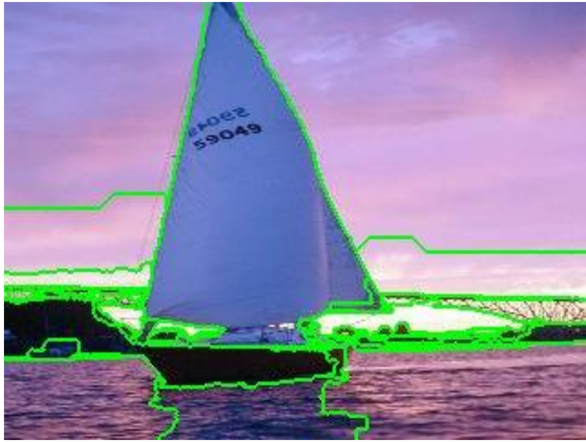
3. into 2D shapes, such as
   - circles
   - ellipses
   - ribbons (long, symmetric regions)



# Region Segmentation: Segmentation Criteria

A segmentation is a partition of an image I into a set of regions S satisfying:

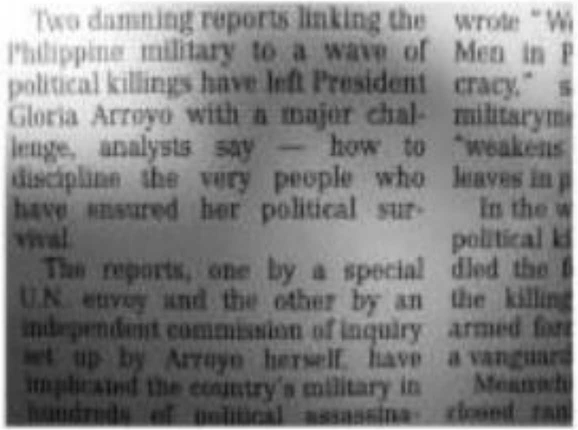
1. $\cup \, S_i = S$ — Partition covers the whole image.
2. $S_i \cap S_j = \phi, \; i \neq j$ — No regions intersect.
3. $\forall \, S_i, \; P(S_i) = \text{true}$ — Homogeneity predicate
4. $P(S_i \cup S_j) = \text{false}$, $i \neq j$, $S_i$ adjacent $S_j$ — Union of adjacent regions does not satisfy homogeneity.



# Segmentation: Thresholding based approaches

# Two class Segmentation: Motivating example

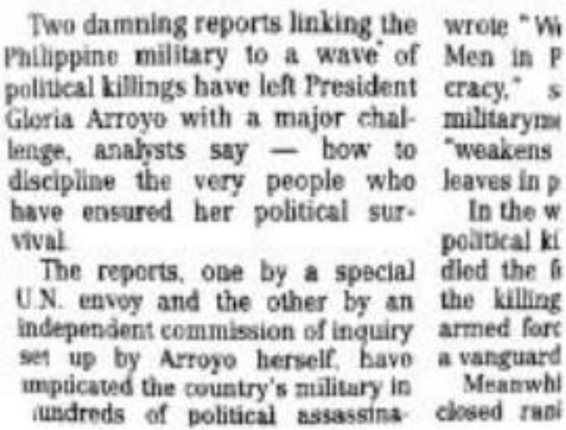
- Separate pixels associated with object of interest from background



Two damning reports linking the Philippine military to a wave of political killings have left President Gloria Arroyo with a major challenge, analysts say — how to discipline the very people who have ensured her political survival.

The reports, one by a special U.N. envoy and the other by an independent commission of inquiry set up by Arroyo herself, have implicated the country's military in hundreds of political assassina-

wrote "W
Men in P
cracy," s
militarym
"weakens
leaves in p
In the w
political ki
dled the f
the killing
armed forc
a vanguard
Meanwhi
closed ran



Two damning reports linking the Philippine military to a wave of political killings have left President Gloria Arroyo with a major challenge, analysts say — how to discipline the very people who have ensured her political survival.

The reports, one by a special U.N. envoy and the other by an independent commission of inquiry set up by Arroyo herself, have implicated the country's military in hundreds of political assassina-

wrote "W
Men in P
cracy," s
militarym
"weakens
leaves in p
In the w
political ki
dled the f
the killing
armed forc
a vanguard
Meanwhi
closed ran

# Thresholding

- Separate pixels associated with object of interest from background

- Given a image f(x,y), the segmented image g(x,y) is given by:

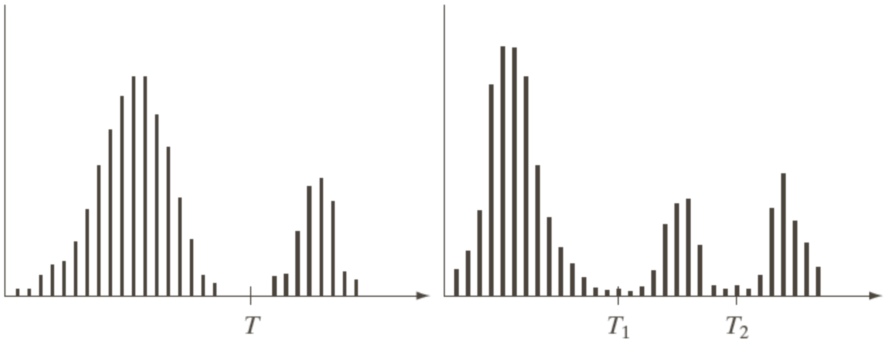
$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases}$$

If T is constant over the entire image → Global Thresholding
If T changes over the image → Variable Thresholding

The main question is: **How to find T?**

# Thresholding

- How to find T?
- One Idea is to explore the intensity histograms (if there is clear separation)

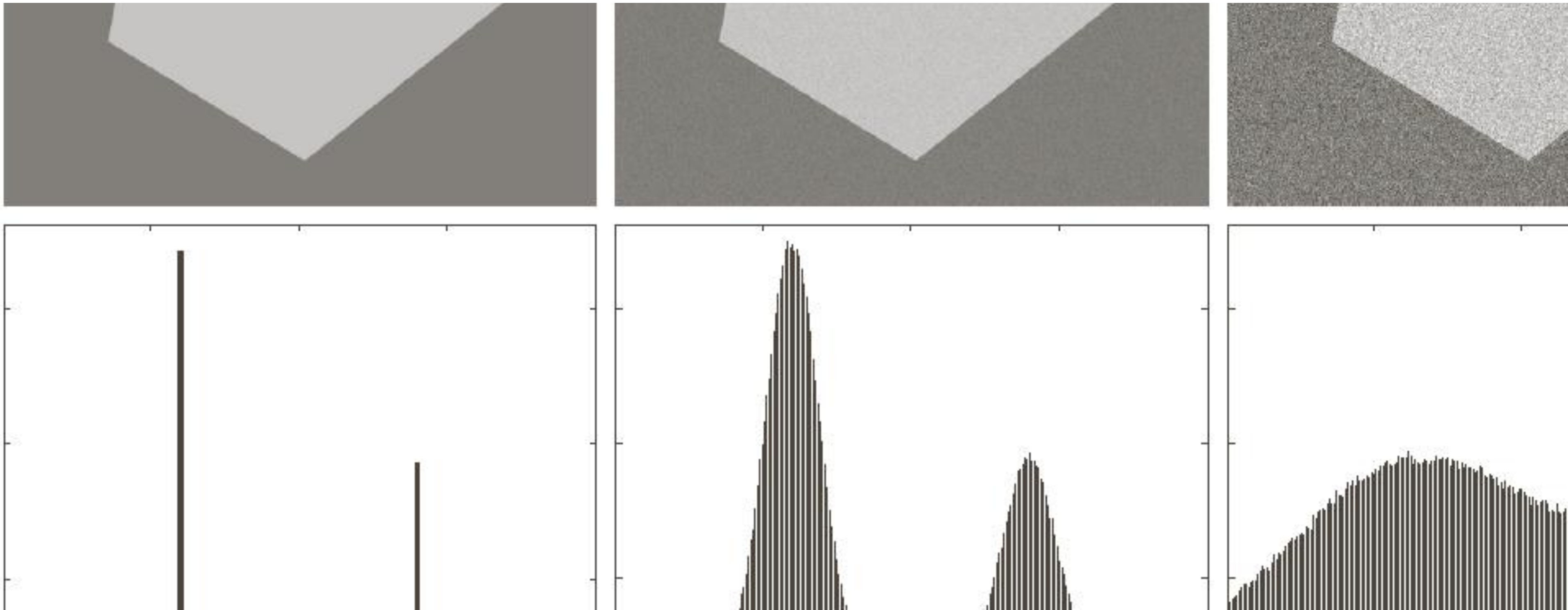


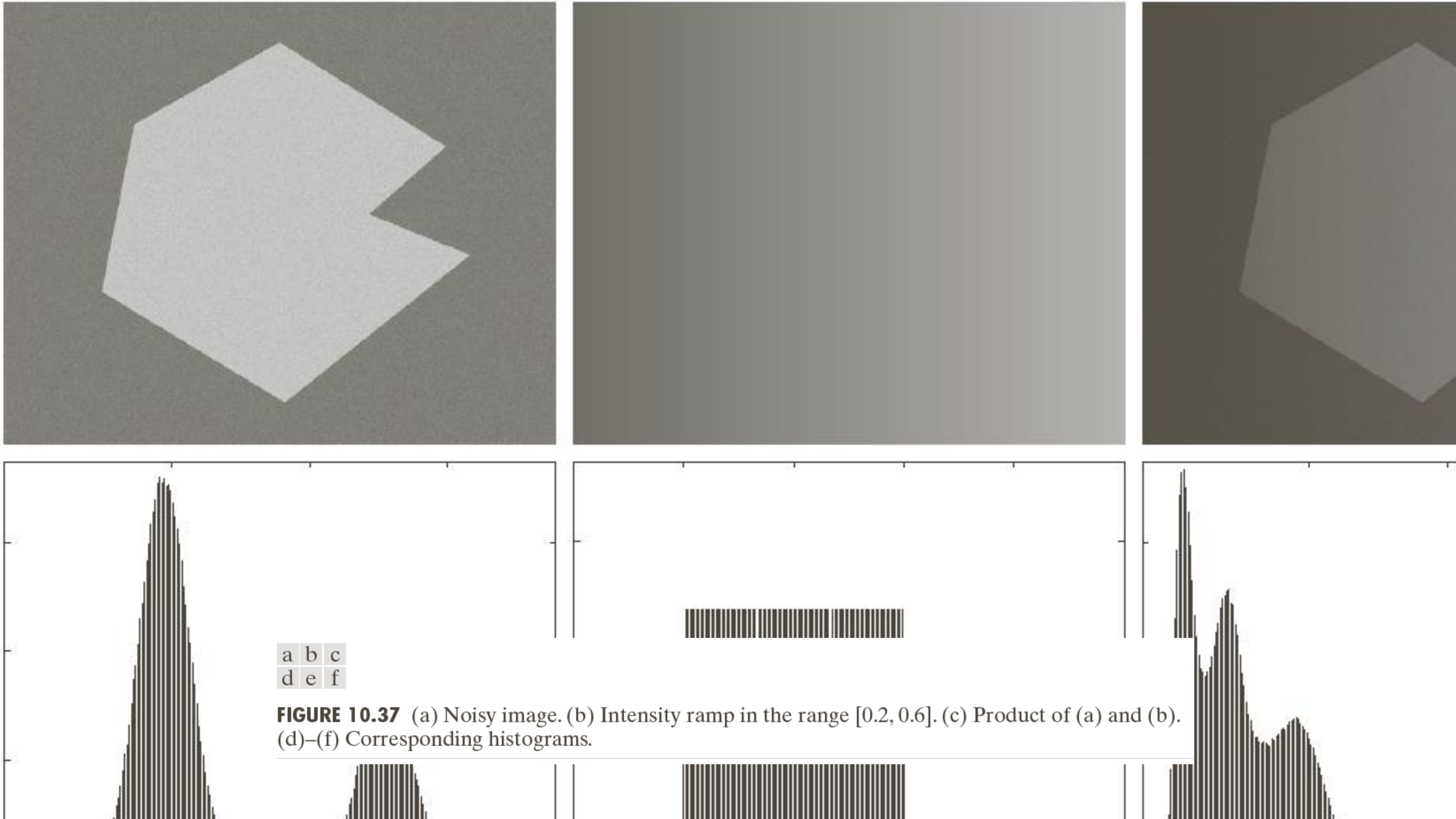

a b

**FIGURE 10.35** Intensity histograms that can be partitioned (a) by a single threshold, and (b) by dual thresholds.

| | | |
|---|---|---|
| a | b | c |
| d | e | f |

**FIGURE 10.36** (a) Noiseless 8-bit image. (b) Image with additive Gaussian noise of mean deviation of 10 intensity levels. (c) Image with additive Gaussian noise of mean 0 and stand 50 intensity levels. (d)–(f) Corresponding histograms.





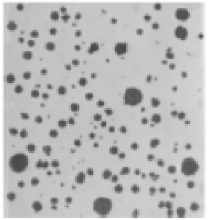
a b c
d e f

**FIGURE 10.37** (a) Noisy image. (b) Intensity ramp in the range [0.2, 0.6]. (c) Product of (a) and (b). (d)–(f) Corresponding histograms.

## Choosing a threshold is something of a “black art”:

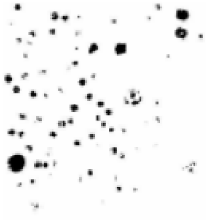
```
n=imread('nodules1.jpg');
figure(1); imshow(n);
n1=im2bw(n,0.35);
n2=im2bw(n,0.75);
figure(2), imshow(n1);
figure(3), imshow(n2);
```

a



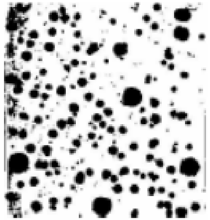
n: Original image

b



n1: Threshold too low

c



n2: Threshold too high

# Finding T: Basic Global Thresholding

Iterative approach

1. Select an initial estimate of global threshold T
2. Segment the image using T
   - This will produce two groups of pixels (G1 and G2)
3. Compute the average (mean) intensity values m1 and m2 for the pixels in G1 and G2 respectively
4. Compute a new threshold value T_new = (m1+m2)/2
5. If |T_new – T| < eps, stop.
6. Else, set T = T_new. Go to Step 2.

# Finding T: Basic Global Thresholding

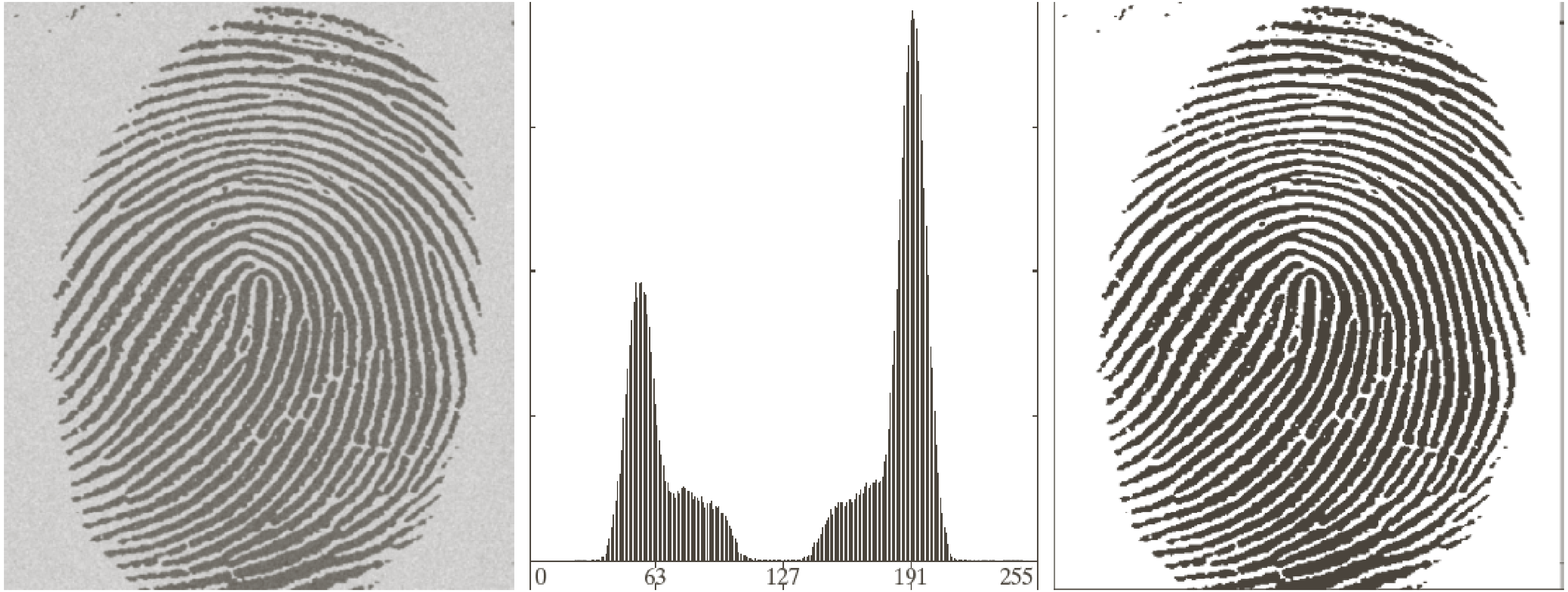
Iterative approach

1. Select an initial estimate of global threshold T
2. Segment the image using T
   – This will produce two groups of pixels (G1 and G2)
3. Compute the average (mean) intensity values m1 and m2 for the pixels in G1 and G2 respectively
4. Compute a new threshold value T_new = (m1+m2)/2
5. If |T_new – T| < eps, stop.
6. Else, set T = T_new. Go to Step 2.

Matlab function: `opthr`

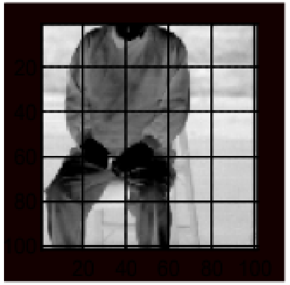
a b c

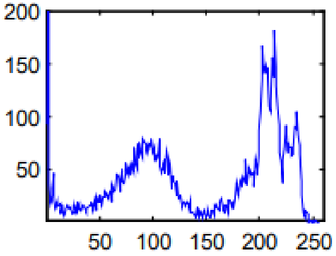
**FIGURE 10.38** (a) Noisy fingerprint. (b) Histogram. (c) Segmented result using a global thresh
was added for clarity). (Original courtesy of the National Institute of Standards and Technol

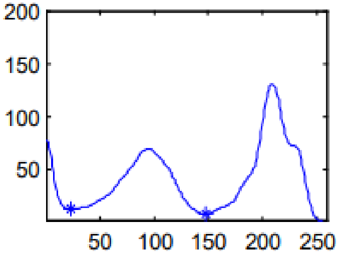


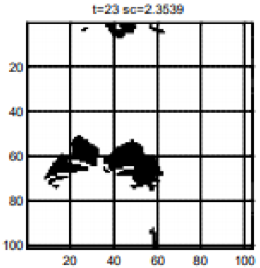
# Global Thresholding

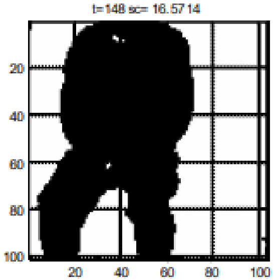
T is usually located at the valley/ one of the valleys

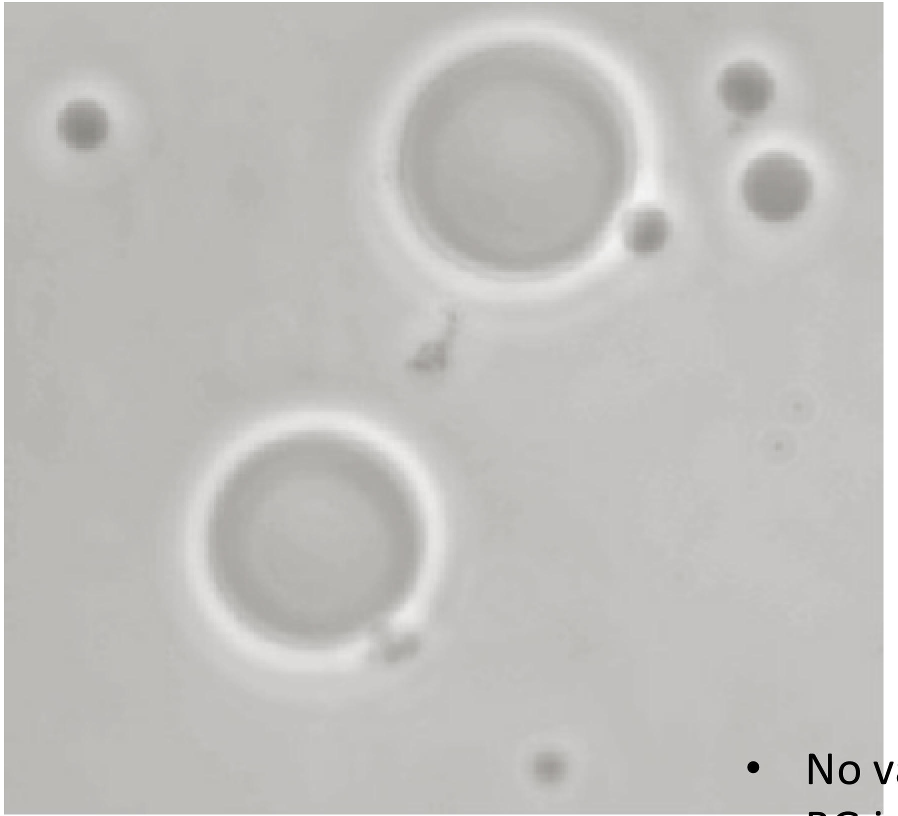


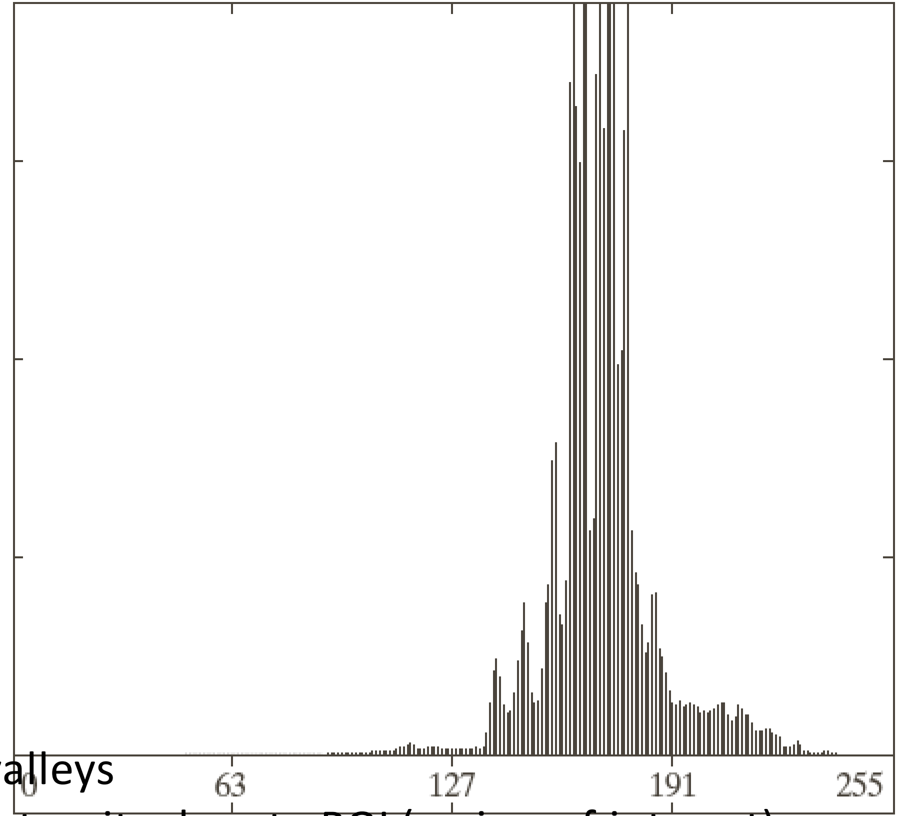




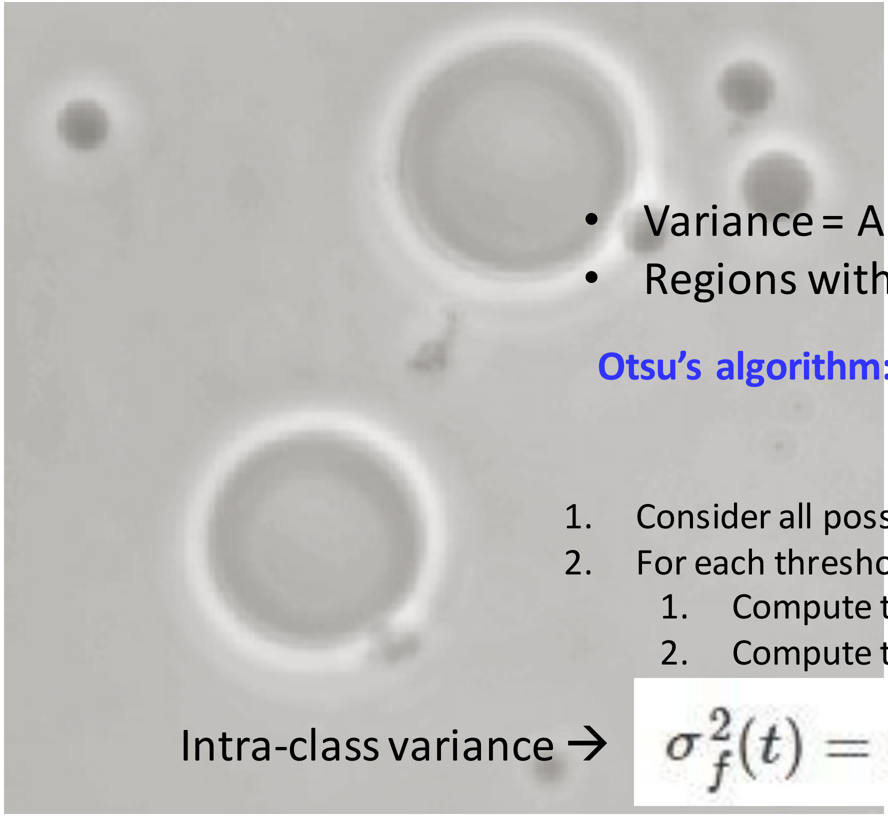


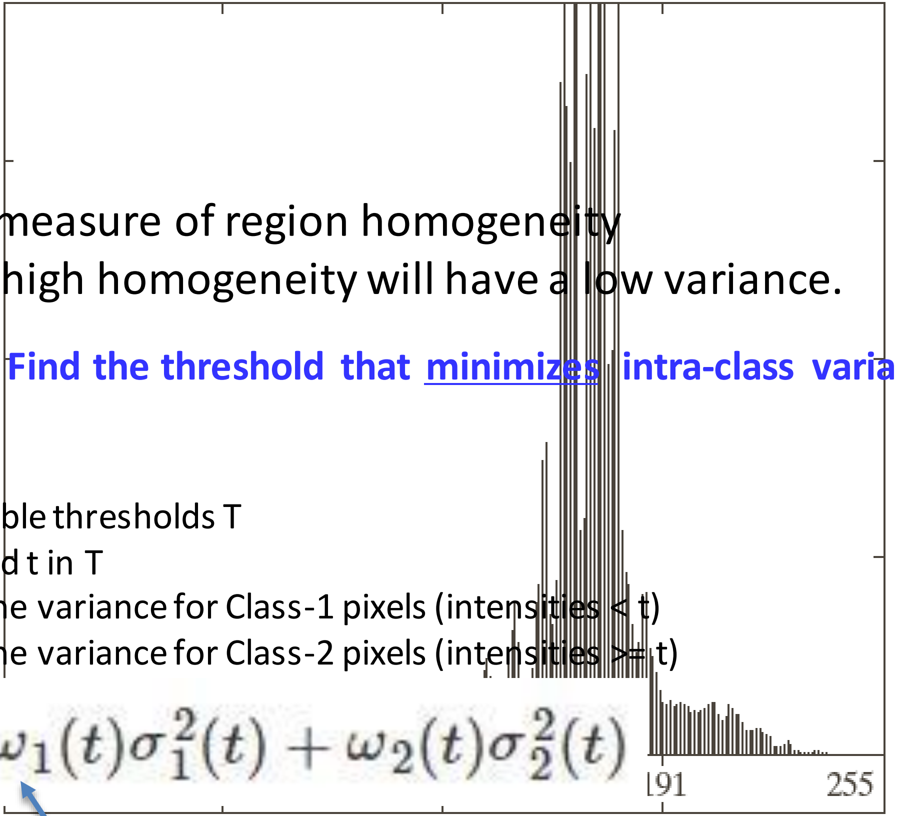

- No valleys
- BG intensity close to ROI (region-of-interest) intensity

# Global Thresholding: Otsu's Method

- Based on histograms
- Automatically finds the optimal threshold maximizing the between class variance
- Proposed in 1975

- Preliminaries:
  - What is the formula for mean and variance of intensities in an image ?
  - What does variance measure ?
  - A probabilistic / normalized-histogram perspective for mean, variance






- Variance = A measure of region homogeneity
- Regions with high homogeneity will have a low variance.

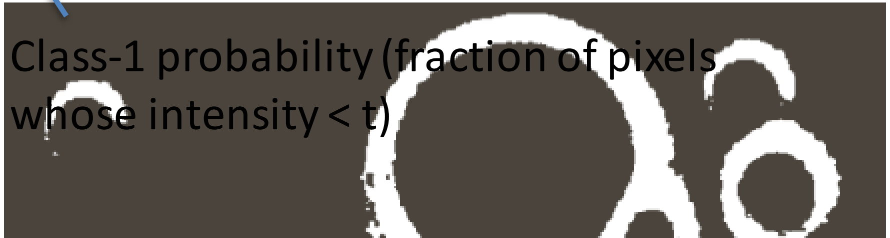
**Otsu's algorithm: Find the threshold that <u>minimizes</u> intra-class variance.**

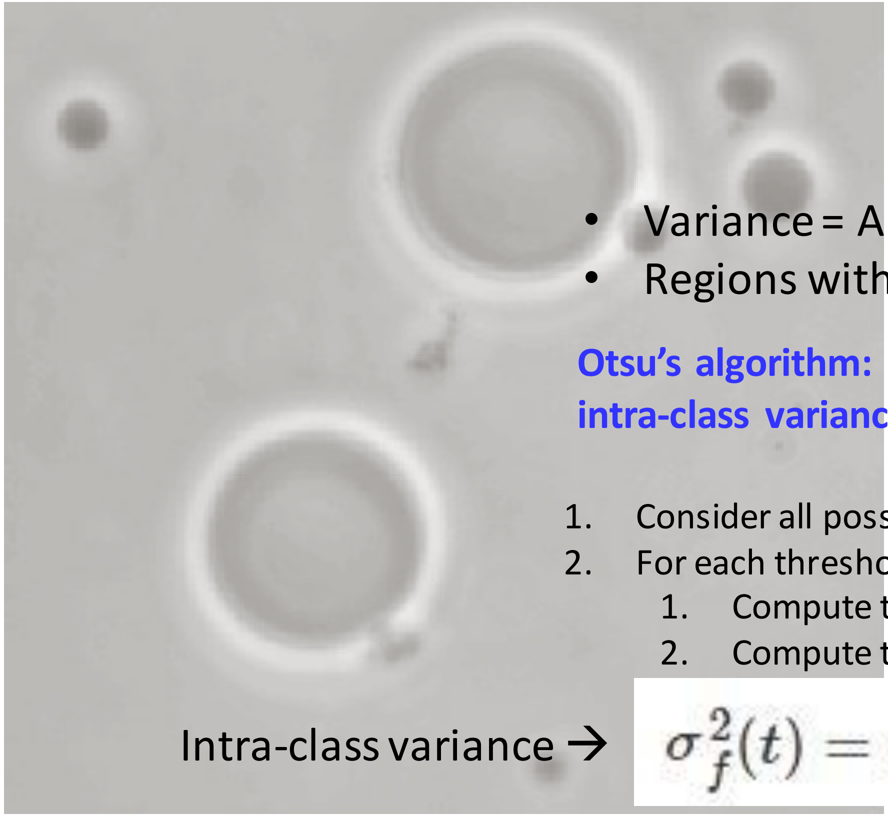
1. Consider all possible thresholds T
2. For each threshold t in T
   1. Compute the variance for Class-1 pixels (intensities < t)
   2. Compute the variance for Class-2 pixels (intensities >= t)

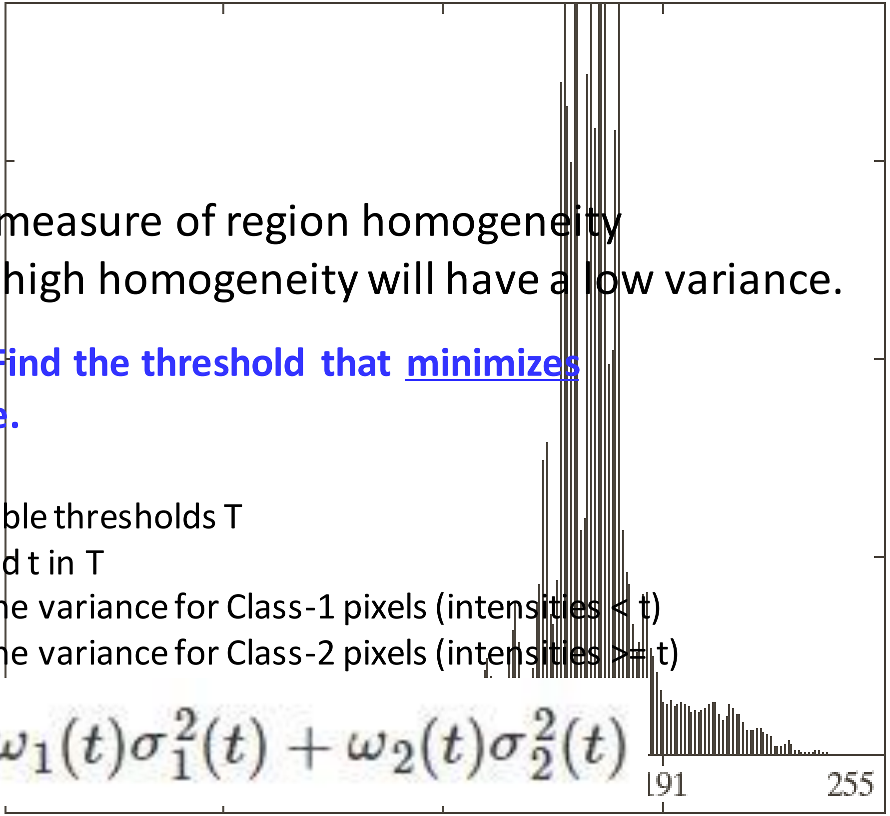
Intra-class variance →

$$\sigma_f^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

Class-1 probability (fraction of pixels whose intensity < t)








- Variance = A measure of region homogeneity
- Regions with high homogeneity will have a low variance.

**Otsu's algorithm: Find the threshold that <u>minimizes</u> intra-class variance.**

1. Consider all possible thresholds T
2. For each threshold t in T
   1. Compute the variance for Class-1 pixels (intensities < t)
   2. Compute the variance for Class-2 pixels (intensities >= t)

Intra-class variance →

$$\sigma_f^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

# Otsu's Method

- Compute the normalized histogram of the input image.
- Denote the components of the histogram by $p_i, \; i = 0, 1, 2, 3, \dots, L-1$
- Suppose a threshold is selected $k, 0 < k < L-1$
- $C_1$ is the set of pixels with levels $[0, 1, 2, 3, \dots, k]$
- $C_2$ is the set of pixels with levels $[k+1, k+2, k+3, \dots, L-1]$

- Obtain the value of threshold which **maximizes** the between class variance

$$\boxed{\sigma_B^2(k) = P_1(k)\,(m_1(k) - m_G)^2 + P_2(k)\,(m_2(k) - m_G)^2}$$

# Otsu's Method

$$\boxed{\sigma_B^2(k) = P_1(k)\,(m_1(k) - m_G)^2 + P_2(k)\,(m_2(k) - m_G)^2}$$

- $P_1(k)$ is probability of $C_1$ occuring

$$P_1(k) = \sum_{i=0}^{k} p_i \;, k = 0,1,2, \dots , k$$

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i \; = 1 - P_1(k)\;, k = 0,1,2, \dots , k$$

- $m_1(k)$ and $m_2(k)$ are means of $C_1$ and $C_2$

$$m_1(k) = \frac{\sum_{i=0}^{k} i\, p_i}{P_1(k)} \qquad m_2(k) = \frac{\sum_{i=k+1}^{L-1} i\, p_i}{P_2(k)}$$

# Otsu's Method

$$\boxed{\sigma_B^2(k) = P_1(k)\,(m_1(k) - m_G)^2 + P_2(k)\,(m_2(k) - m_G)^2}$$

$$\boxed{\sigma_B^2(k^*) = \max_{0 \le k \le L-1} \sigma_B^2(k)}$$

In simple words, we evaluate all values of k and select the value of k that yielded the maximum $\sigma_B^2(k)$

This idea can be easily extended to compute multiple thresholds!

| a | b |
|---|---|
| c | d |

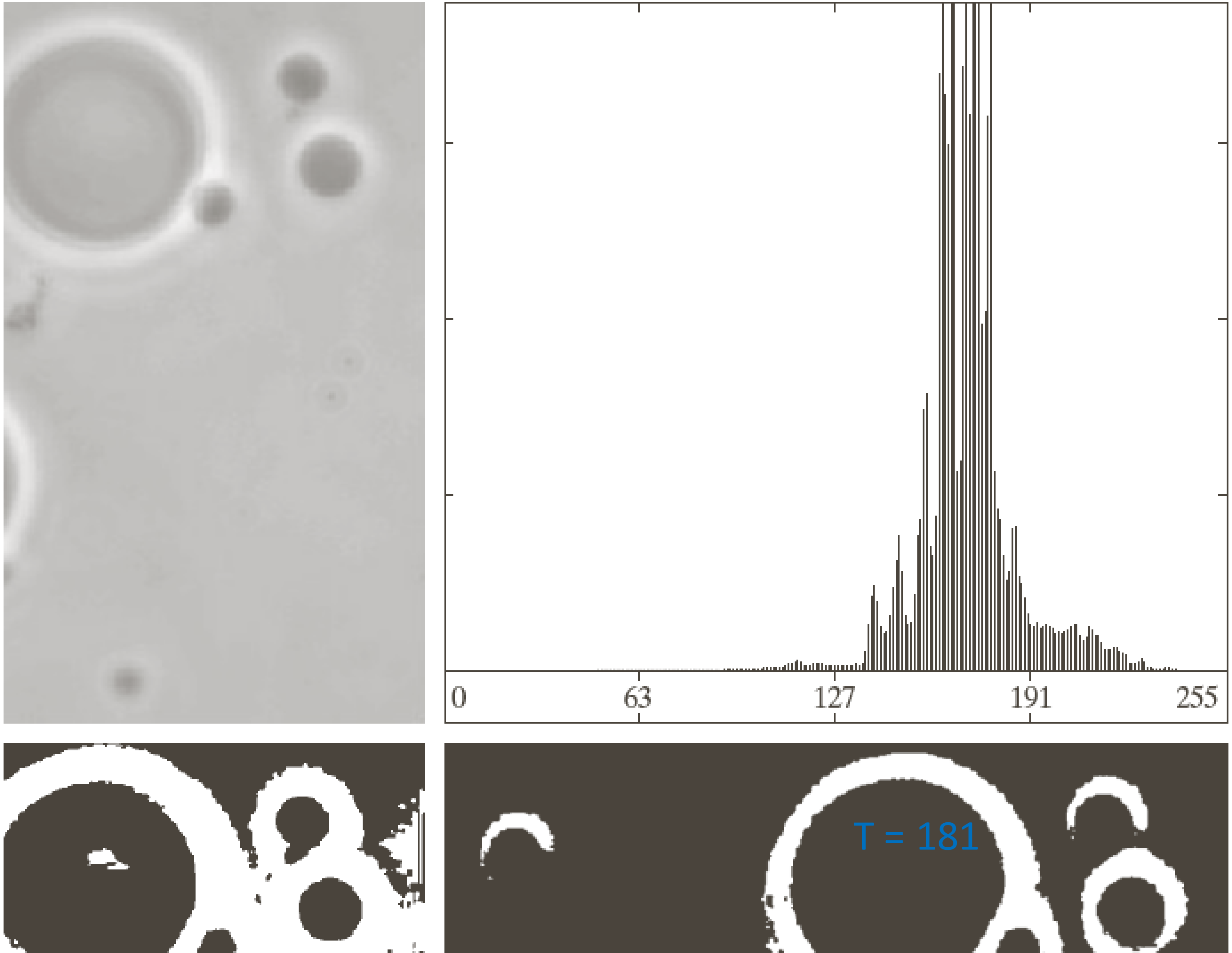
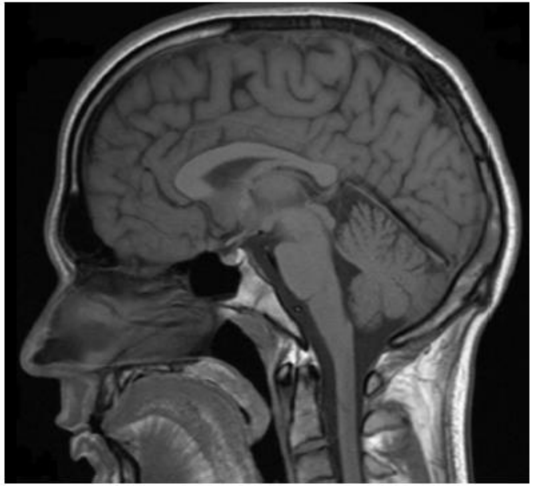
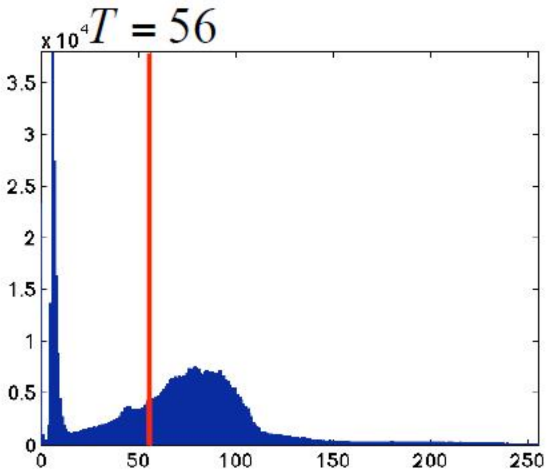
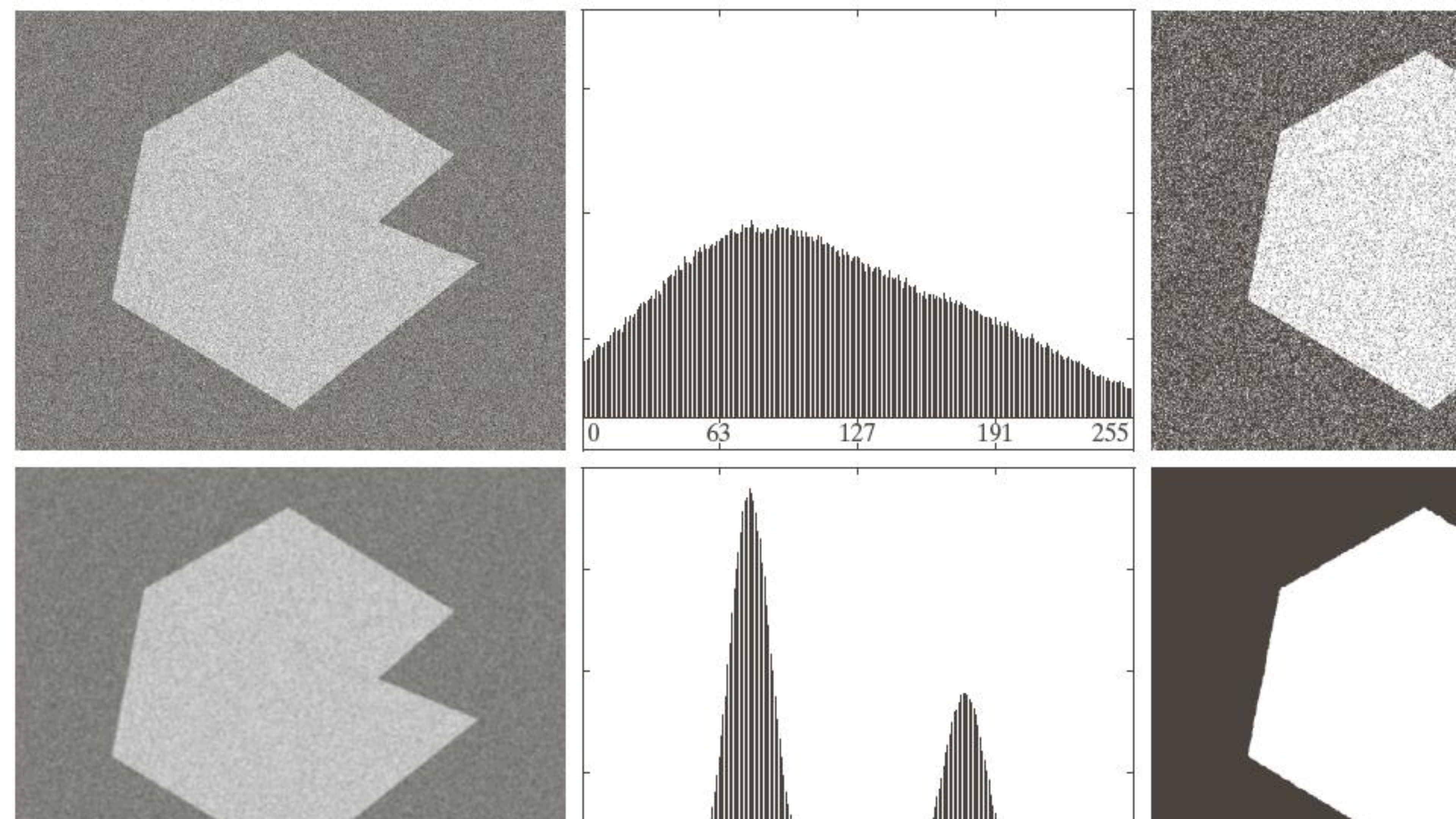



**FIGURE 10.39**
(a) Original image.
(b) Histogram (high peaks were clipped to highlight details in the lower values).
(c) Segmentation result using the basic global algorithm from Section 10.3.2.
(d) Result obtained using Otsu's method.
(Original image courtesy of

# Otsu’s Method



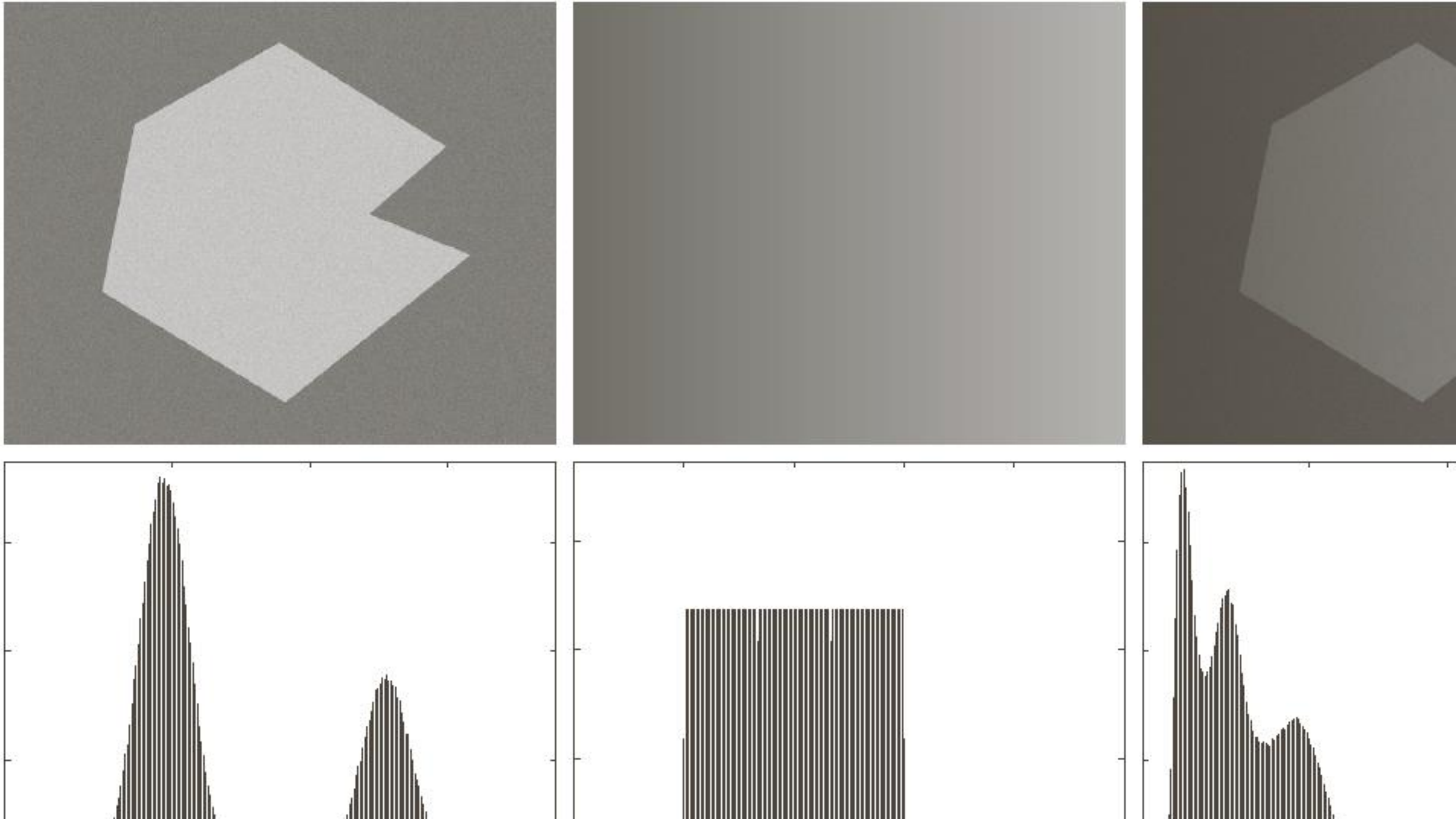


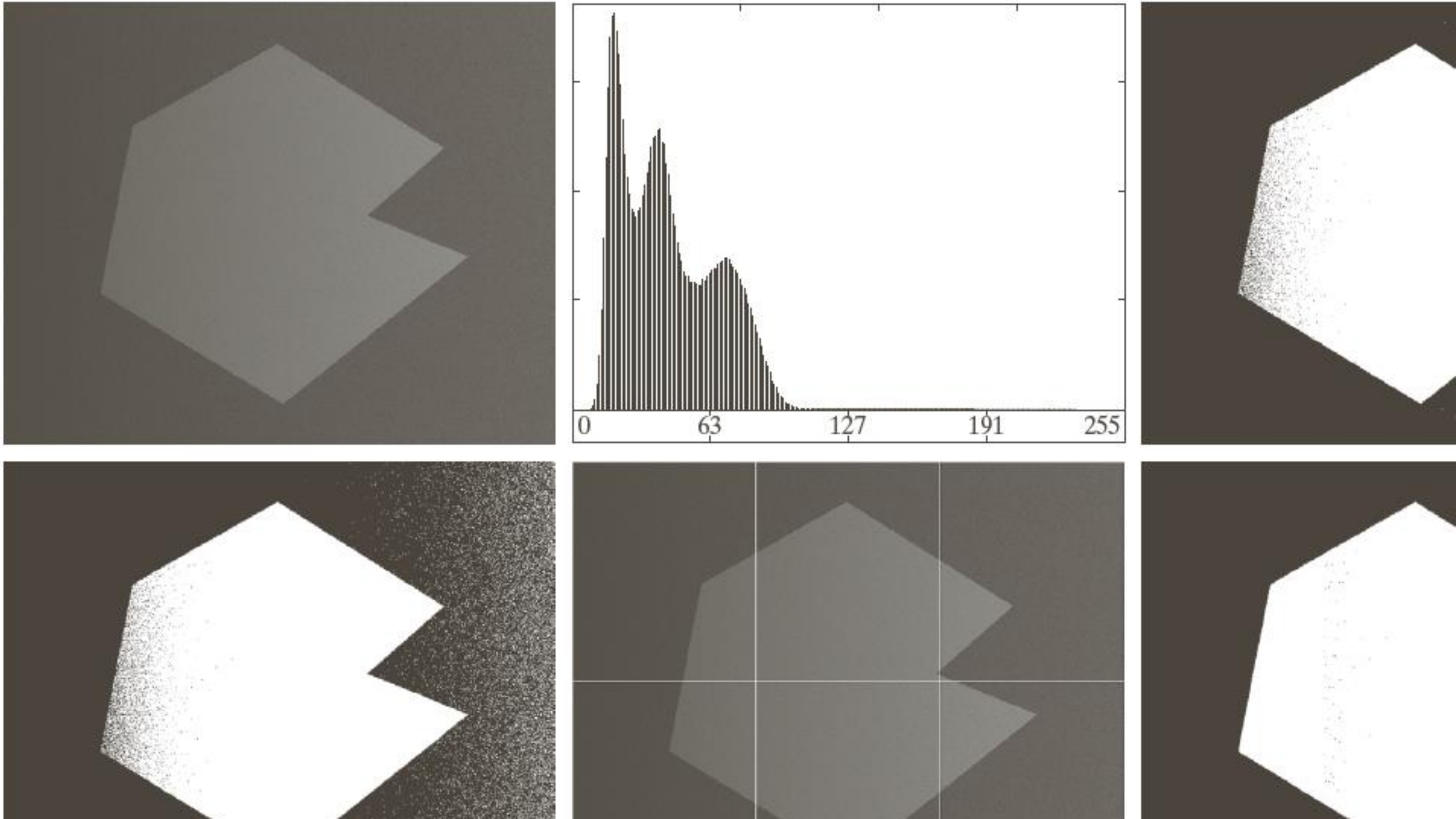



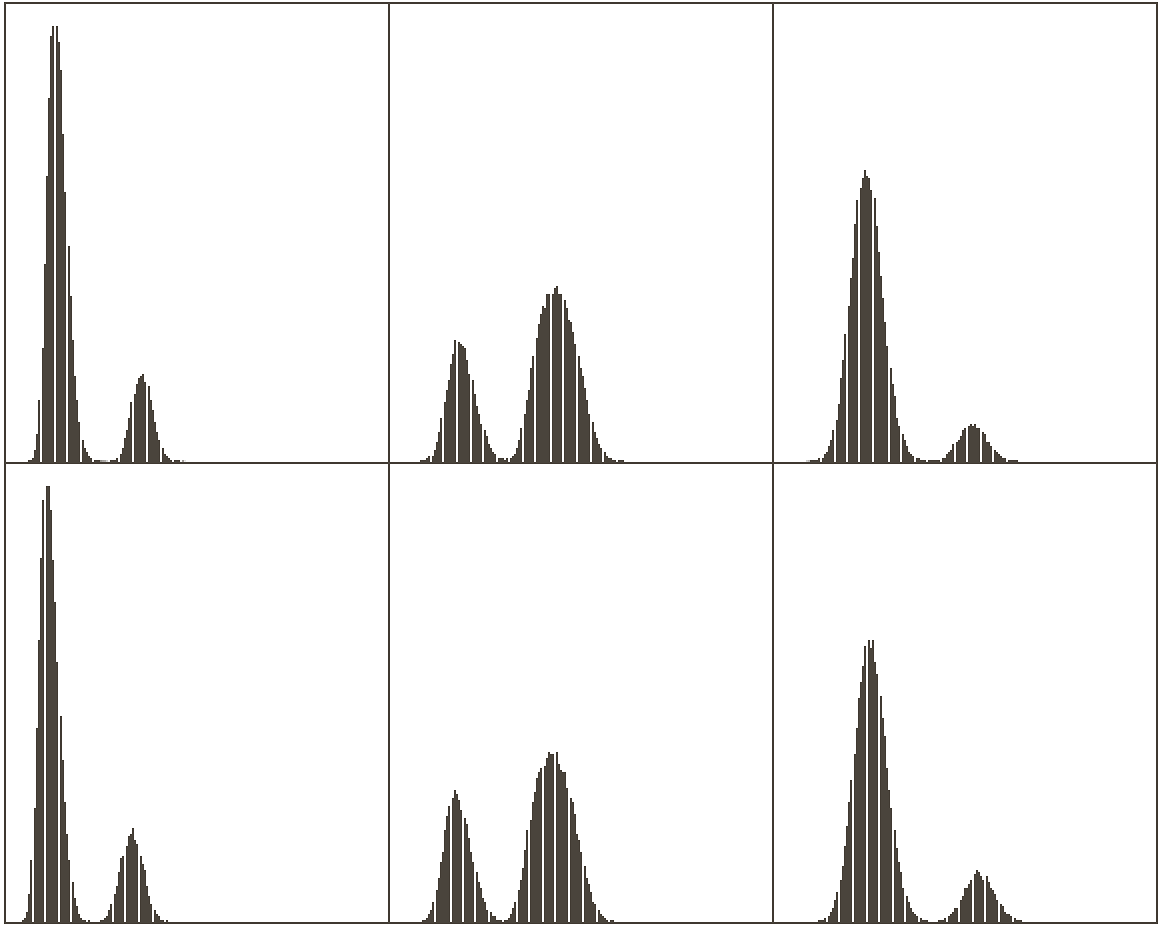
0
63
127
191
255

# Otsu’s method: Main Limitation

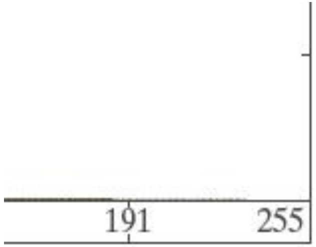

Pai et al. PR 2010

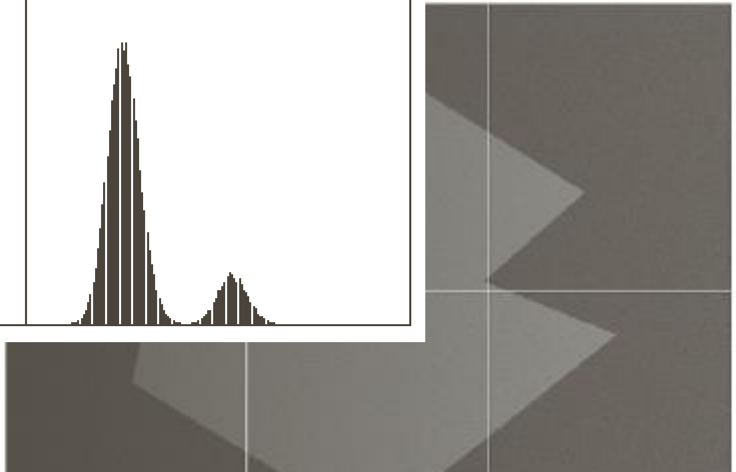



0
63
127
191
255



ogram. (c) Segmentation of (a) usi
l using Otsu's method. (e) Image sub
subimage individually.



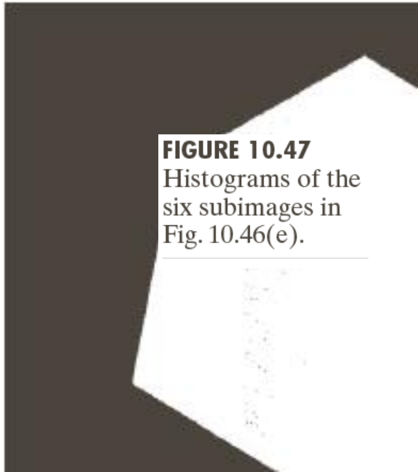


**FIGURE 10.47** Histograms of the six subimages in Fig. 10.46(e).

# Per pixel variable Thresholding

- Compute standard deviation and mean of each pixel (around local neighborhood)
- Let $\sigma_{xy}$ , $m_{xy}$ denote the standard deviation and mean value contained in neighborhood $S_{xy}$ centred around $(x, y)$
- Example threshold function:

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T_{xy} \\ 0 & \text{if } f(x,y) \le T_{xy} \end{cases}$$

$T_{xy} = a\sigma_{xy} + bm_{xy}$ or $T_{xy} = a\sigma_{xy} + bm_G$

| | |
|---|---|
| a | b |
| c | d |

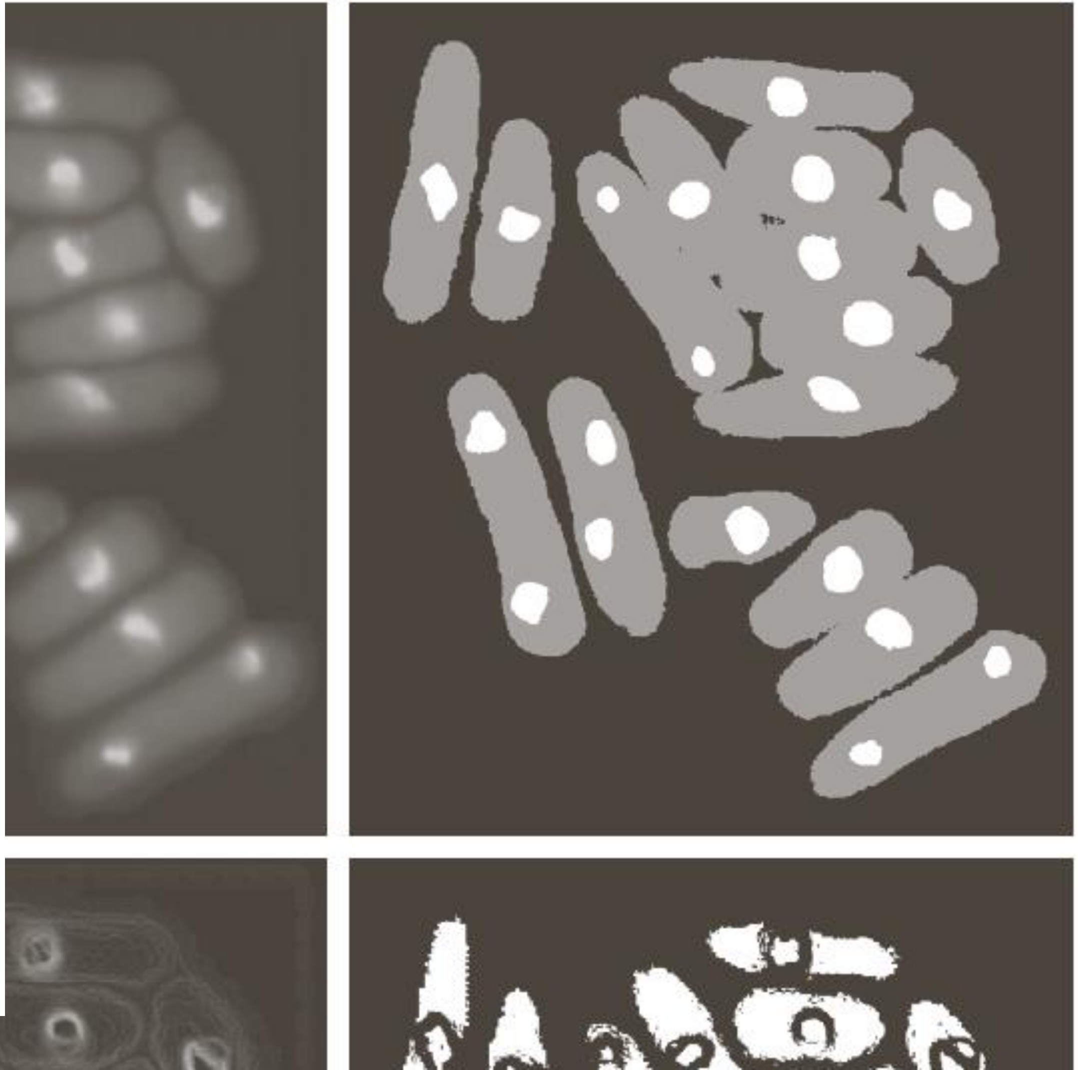
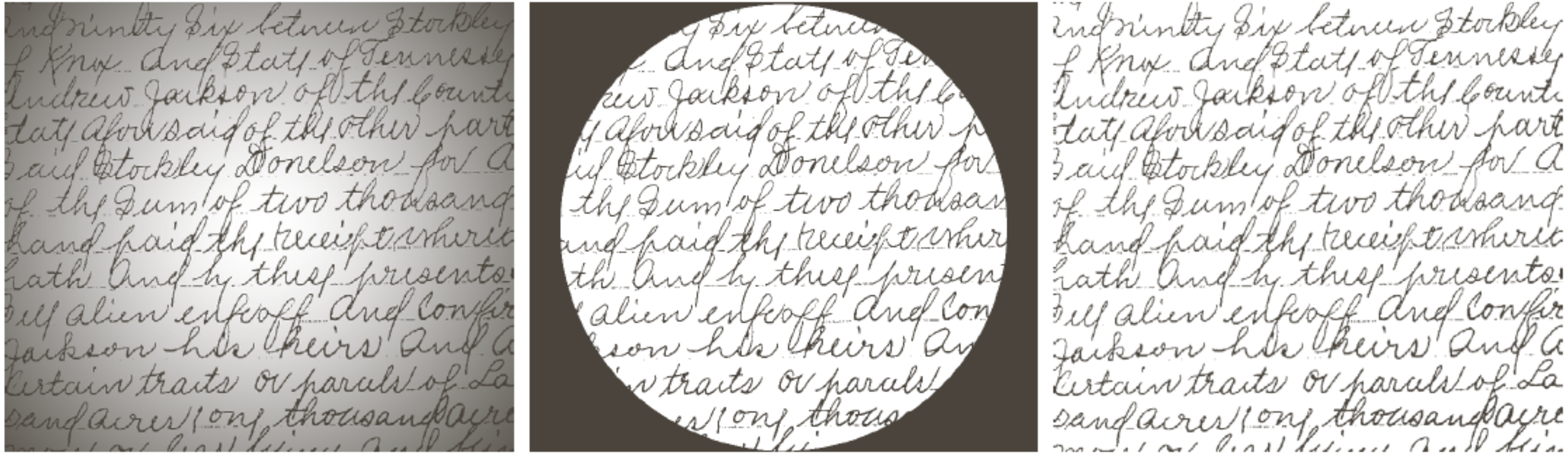


**FIGURE 10.48**
(a) Image from Fig. 10.43.
(b) Image segmented using the dual thresholding approach discussed in Section 10.3.6.
(c) Image of local standard deviations.
(d) Result obtained using local thresholding.

# Per pixel: moving average

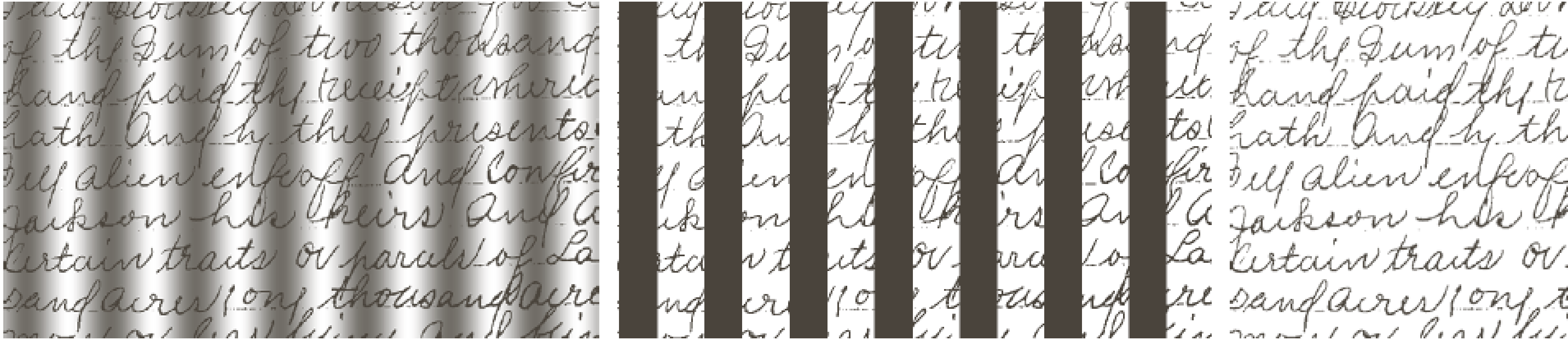


a b c

**FIGURE 10.49** (a) Text image corrupted by spot shading. (b) Result of global thresholding using Otsu's method. (c) Result of local thresholding using moving averages.

a b c

**FIGURE 10.50** (a) Text image corrupted by sinusoidal shading. (b) Result of global threshold
method. (c) Result of local thresholding using moving averages.



# Choosing thresholding algorithms

- Based on typical sizes of objects/regions of interest
  - Small → Adaptive/Local
  - Large → Global

# Thresholding: Summary

- Many methods

- Survey

Sezgin, M and Sankur, B (2004), "Survey over Image Thresholding Techniques and Quantitative Performance Evaluation", Journal of Electronic Imaging 13(1): 146-165

- Comparison

*http://www.fmwconcepts.com/imagemagick/threshold_comparison/index.php*