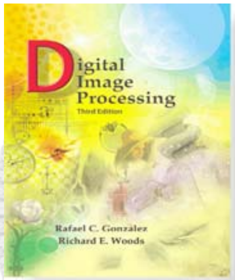
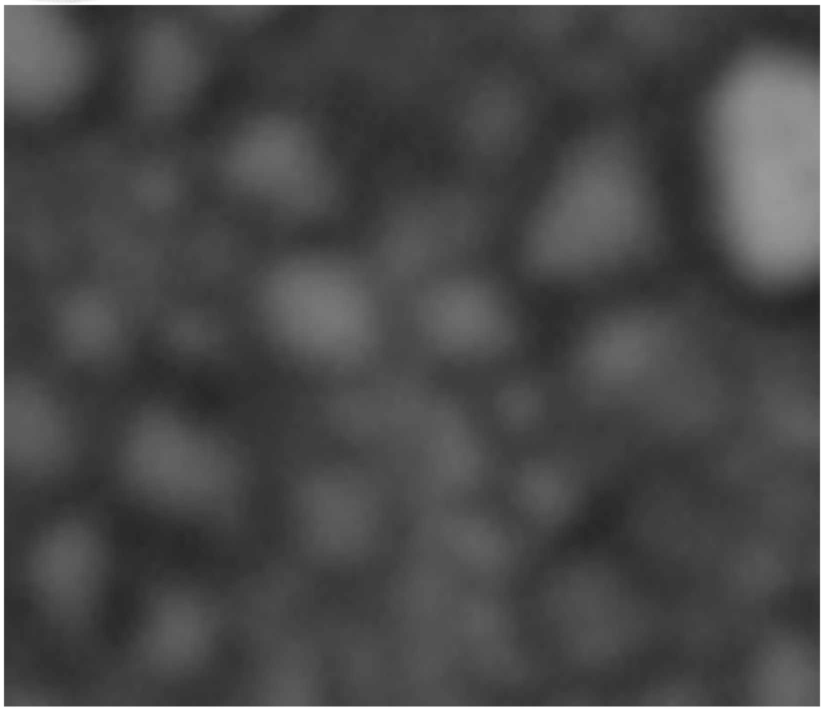
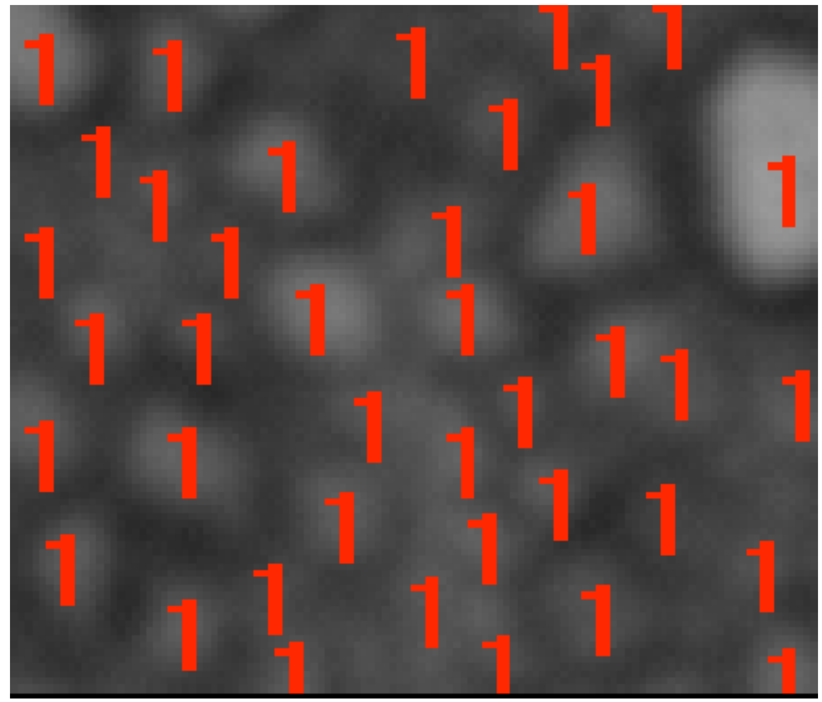


Lecture #22

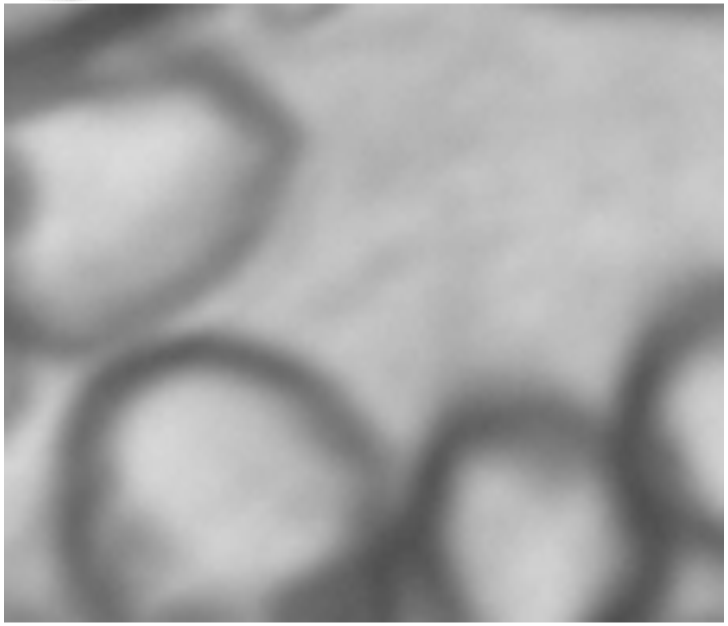
- "Gold Standard" project images
- Otsu thresholding
- Local thresholding
- Region segmentation
- Watershed segmentation
- Frequency-domain techniques

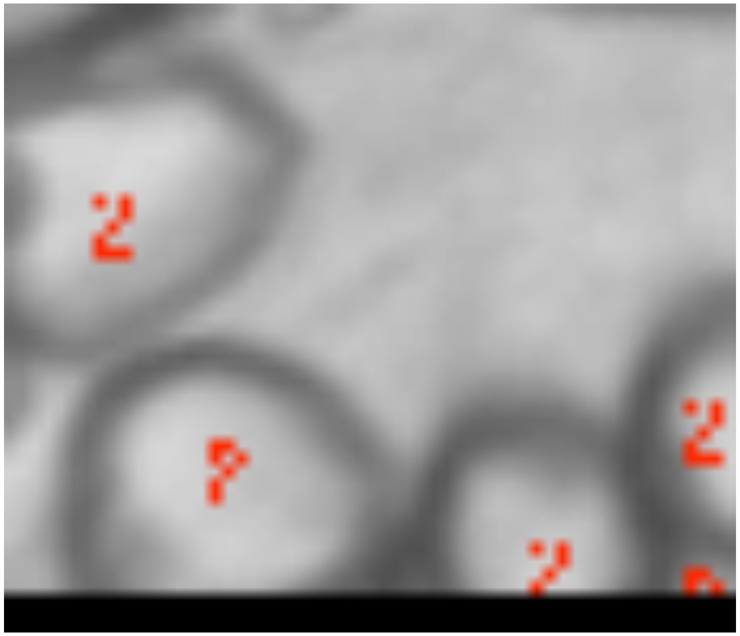
Project Images 1



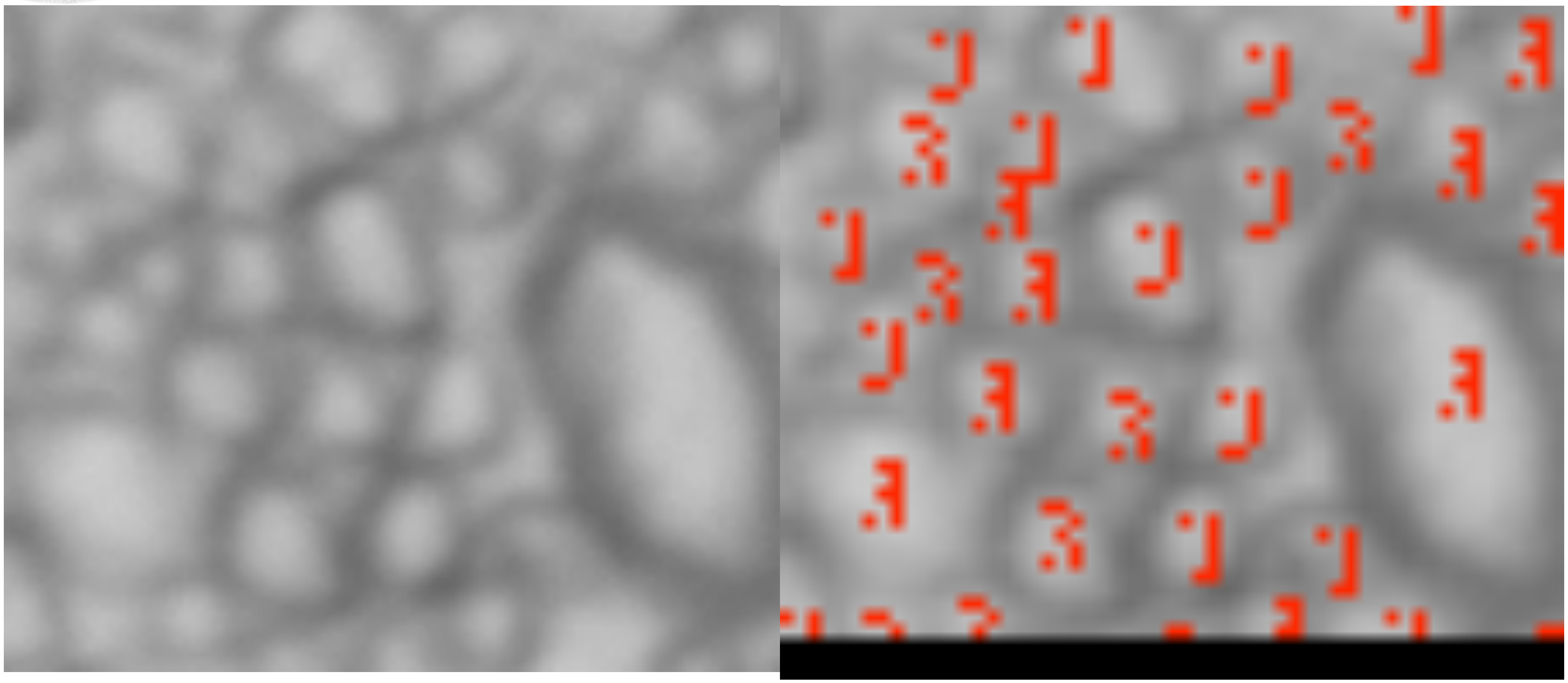


Project Images 2

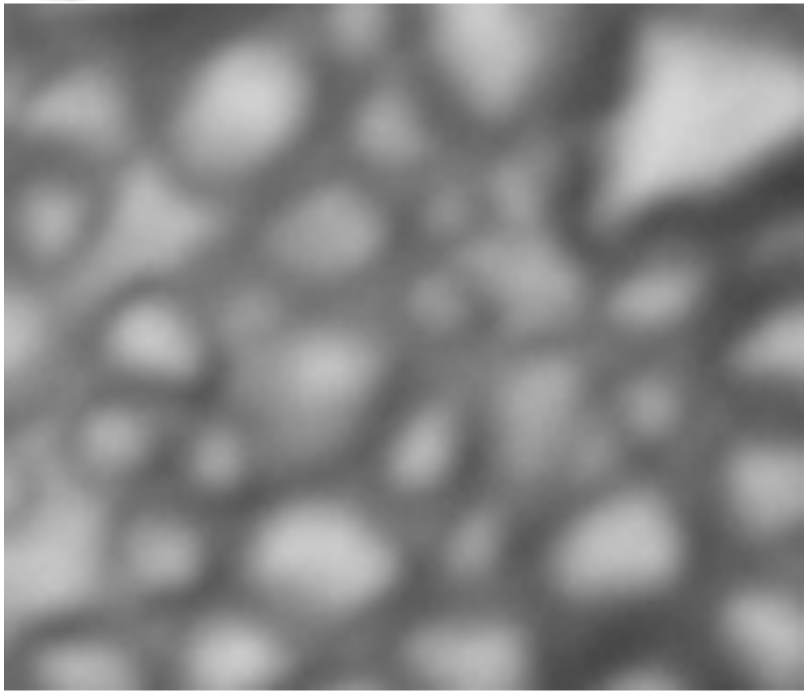


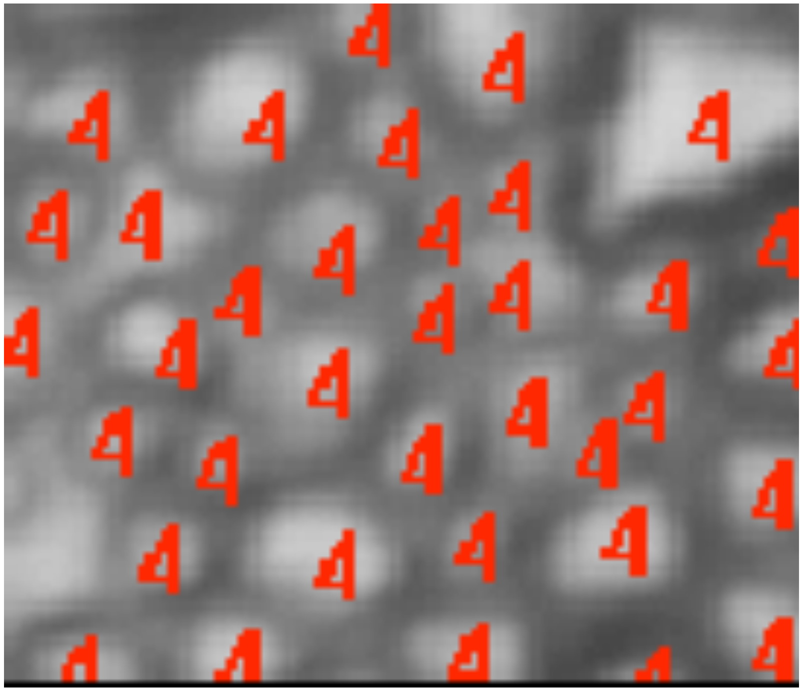


Project Images 3

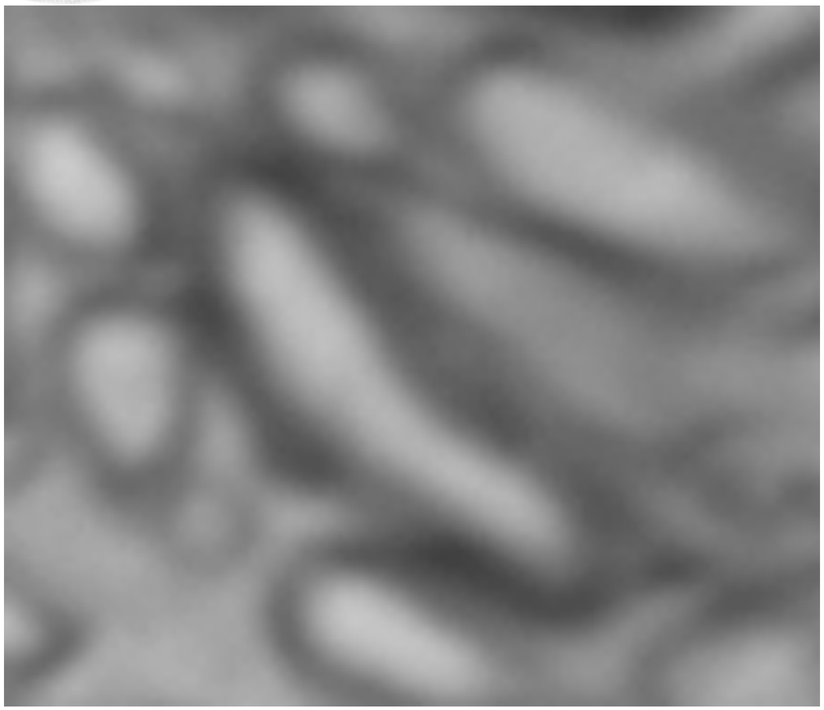


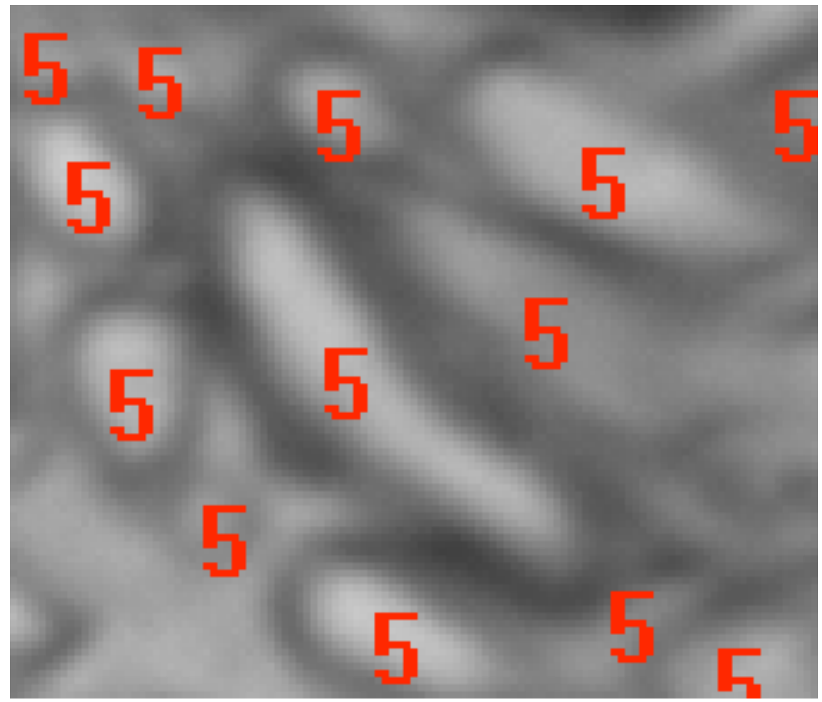
Project Images 4



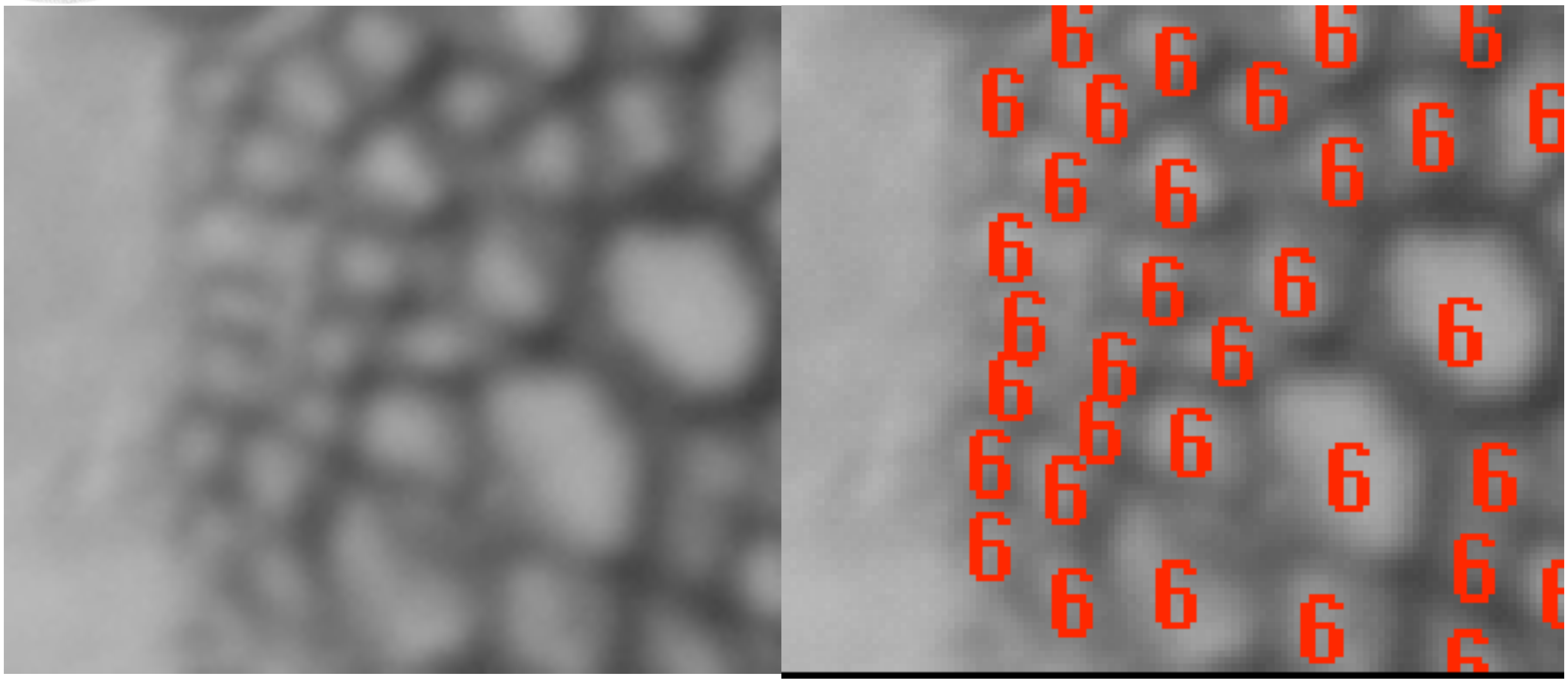


Project Images 5

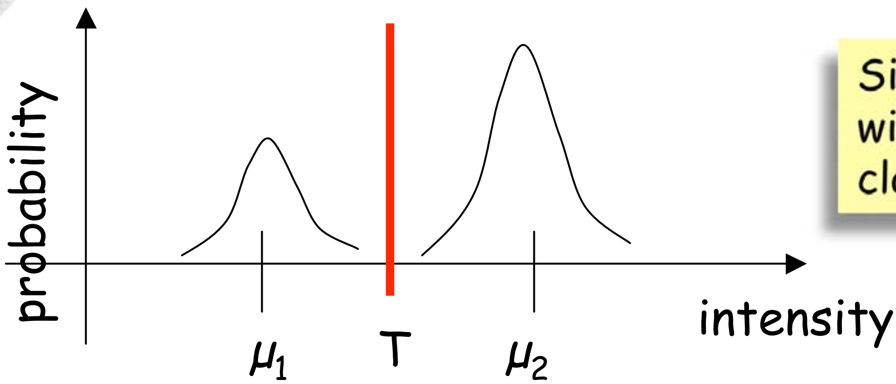




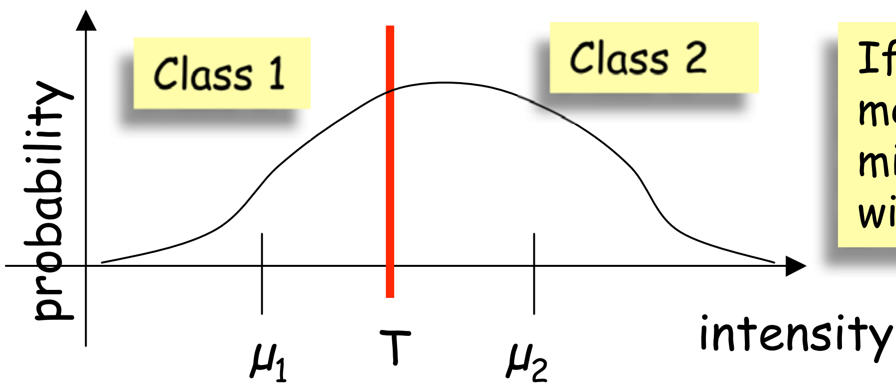
Project Images 6



Optimum Thresholding



Simple bimodal distribution with two homogeneous classes



If there is no valley one method of determining T is to minimize the total variance within both classes.

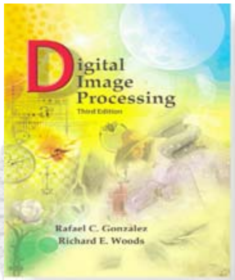


Image Thresholding

- Otsu method minimizes the overall within-class variance by minimizing the weighted sum of class variances

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

Class 1

$$q_1(t) = \sum_{i=0}^{t} P(i) \qquad \sigma_1^2(t) = \frac{1}{q_1(t)} \sum_{i=0}^{t} \left[i - \mu_1(t)\right]^2 P(i)$$

Class 2

$$q_2(t) = \sum_{i=t}^{L-1} P(i) \qquad \sigma_2^2(t) = \frac{1}{q_2(t)} \sum_{i=t+1}^{L-1} \left[i - \mu_2(t)\right]^2 P(i)$$

CDF for each class

Standard deviation of the intensities within each class normalized by the probability of that class

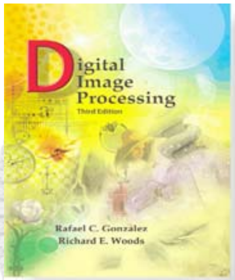


Image Thresholding

http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=16205

1. Compute the histogram of the image. Let each gray level have probability p_i.
2. Compute the cumulative sum $P_1(k)$ for k=0,…,L-1 $$P_1(k) = \sum_{i=0}^{k} p_i$$
3. Compute the cumulative mean m(k) for k=0,…,L-1 $$m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} i p_i$$
4. Compute the global intensity mean m_G $$m_G = \sum_{i=0}^{L-1} i p_i$$
5. Compute the between class variance $\sigma_B^2(k)$ for k=0,…,L $$\sigma_B^2 = P_1(k)\left(m_1(k) - m_G\right)^2 + P_2(k)\left(m_2(k) - m_G\right)^2$$
6. Compute the Otsu threshold k* as the value of k for which $\sigma_B^2(k)$ is maximum
7. Obtain the separability measure η^* $$\eta^* = \frac{\sigma_B^2(k^*)}{\sigma_G^2}$$

The farther apart the means the larger will be $\sigma_B^2(k)$

This is a measure of how easily separable the classes are. Uniform distribution is 0 and a clear, bimodal is 1

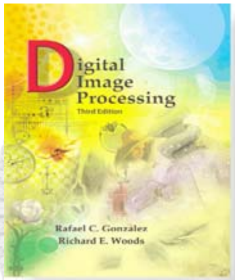
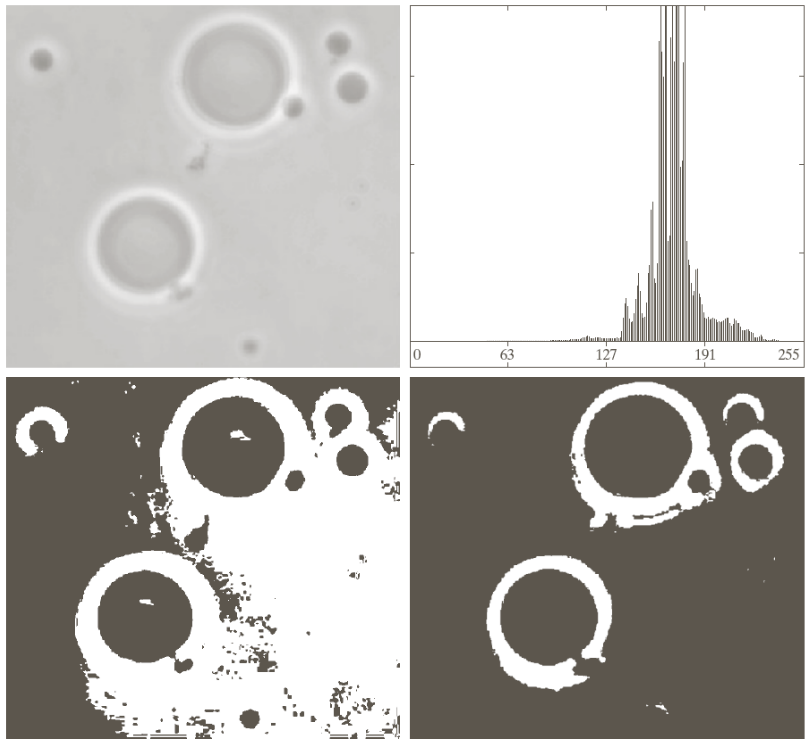


Image Thresholding

Original image.



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 Professor Daniel A. Hammer, the University of Pennsylvania.)

Global thresholding calculating $T=0.5^*(\mu_1+\mu_2)$ until ΔT less than some ε

Global thresholding using Otsu algorithm

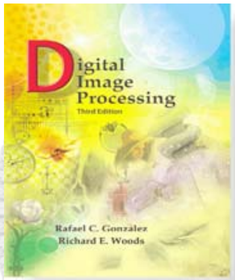
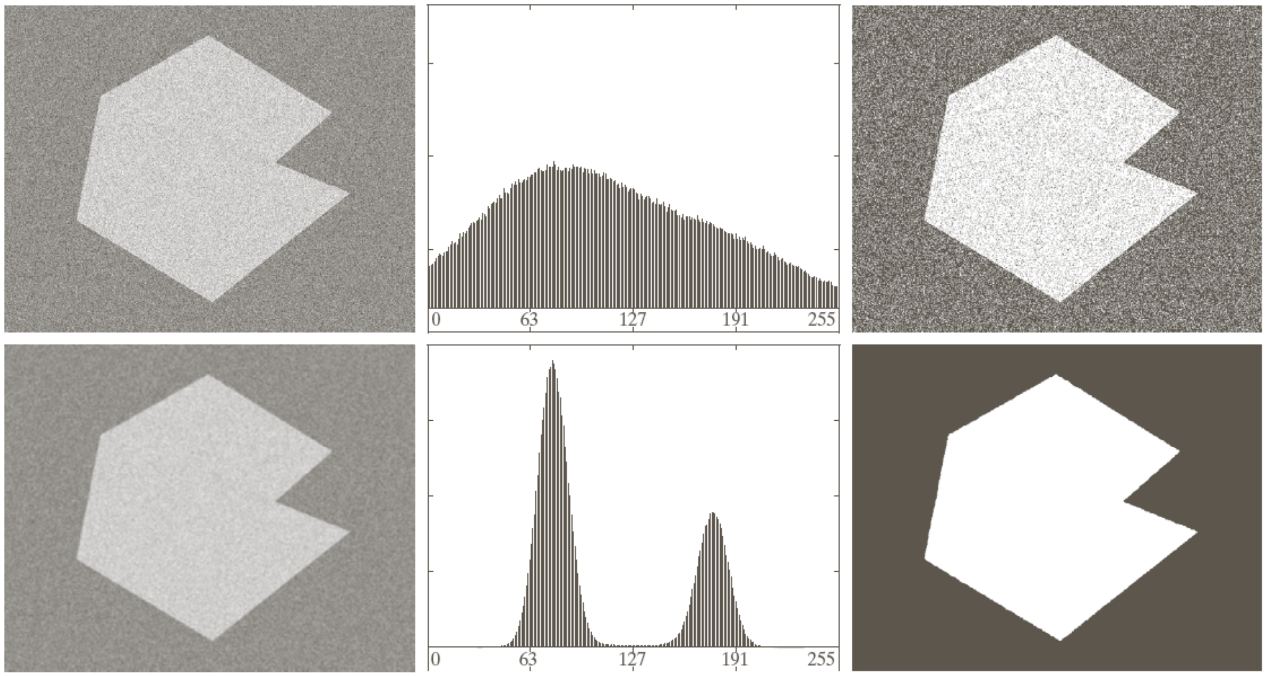


Image Thresholding

Noisy image

Smoothed noisy image



Averaging greatly improves separability

Global thresholding using Otsu algorithm

a b c
d e f

FIGURE 10.40 (a) Noisy image from Fig. 10.36 and (b) its histogram. (c) Result obtained using Otsu's method. (d) Noisy image smoothed using a 5 × 5 averaging mask and (e) its histogram. (f) Result of thresholding using Otsu's method.

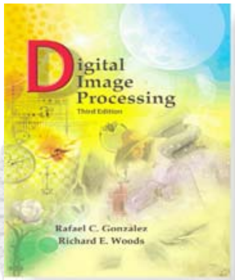
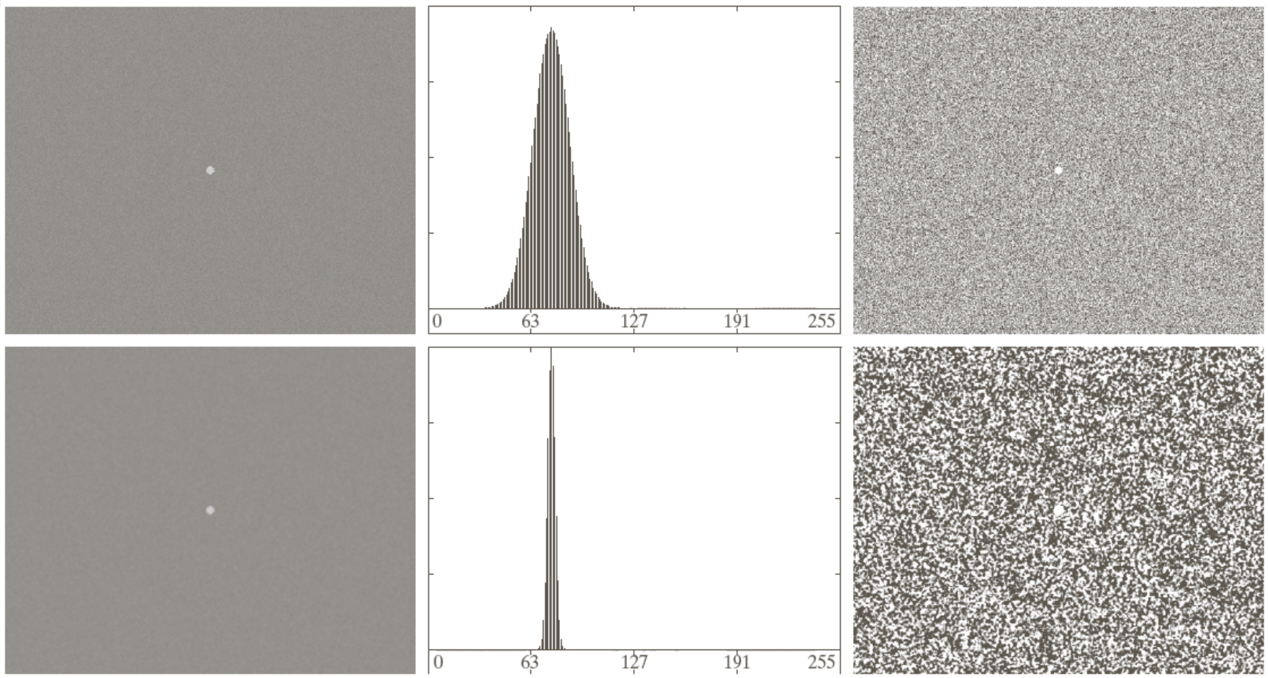


Image Thresholding

Noisy image

Smoothed noisy image



Averaging does not improve separability

Global thresholding using Otsu algorithm

a b c
d e f

FIGURE 10.41 (a) Noisy image and (b) its histogram. (c) Result obtained using Otsu's method. (d) Noisy image smoothed using a 5 × 5 averaging mask and (e) its histogram. (f) Result of thresholding using Otsu's method. Thresholding failed in both cases.

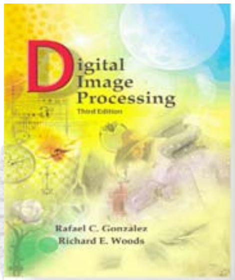


Image Thresholding

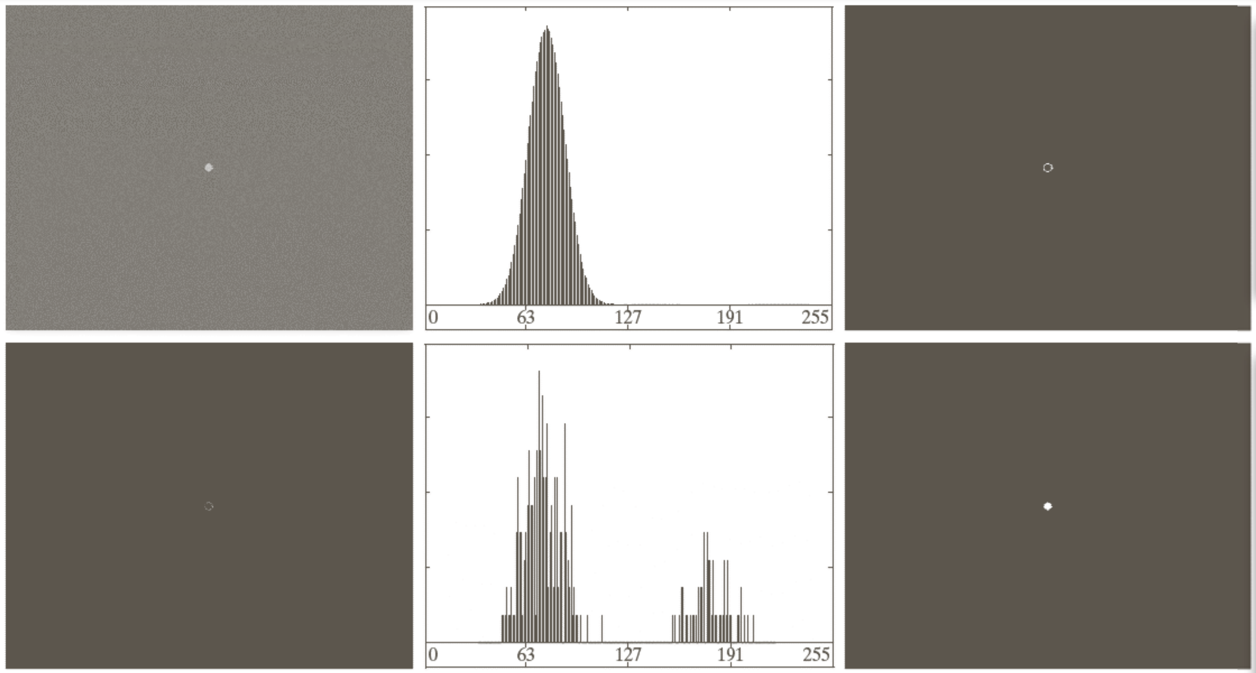
Basic idea is to look at the histogram of pixels only near edges

Noisy image

Thresholded gradient image (very high threshold)

Produce of thresholded gradient image and original image

Global thresholding of ORIGINAL image using Otsu algorithm on histogram of masked image



Edge masking uses only pixels near edges to form histogram

a b c
d e f

FIGURE 10.42 (a) Noisy image from Fig. 10.41(a) and (b) its histogram. (c) Gradient magnitude image thresholded at the 99.7 percentile. (d) Image formed as the product of (a) and (c). (e) Histogram of the nonzero pixels in the image in (d). (f) Result of segmenting image (a) with the Otsu threshold based on the histogram in (e). The threshold was 134, which is approximately midway between the peaks in this histogram.

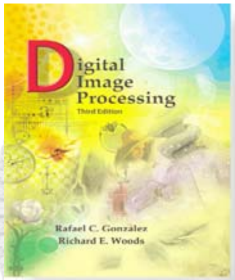


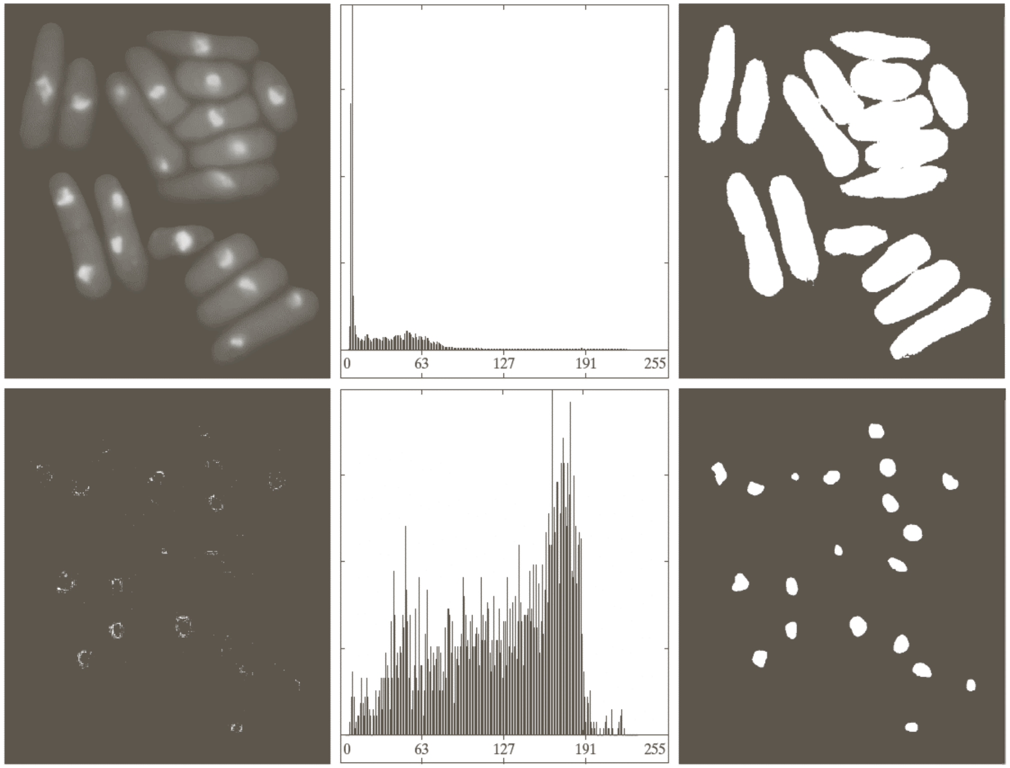
Image Thresholding

Noisy image of yeast cells — want to find bright spots

Otsu thresholding of noisy image histogram

Product of thresholded Laplacian(high threshold) image and original image

Global thresholding of ORIGINAL image using Otsu algorithm on histogram of masked image



Edge masking uses only pixels near edges to form histogram

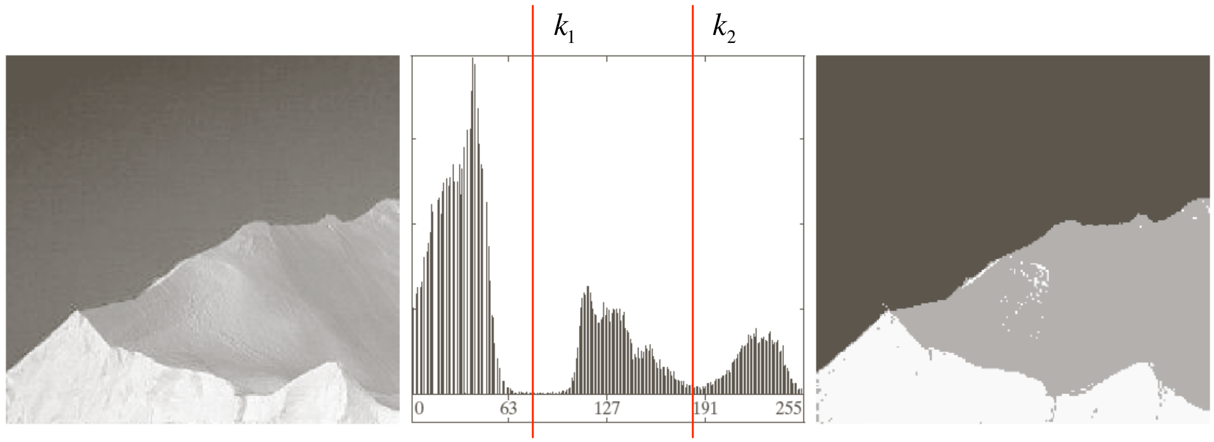
FIGURE 10.43 (a) Image of yeast cells. (b) Histogram of (a). (c) Segmentation of (a) with Otsu's method using the histogram in (b). (d) Thresholded absolute Laplacian. (e) Histogram of the nonzero pixels in the product of (a) and (d). (f) Original image thresholded using Otsu's method based on the histogram in (e). (Original image courtesy of Professor Susan L. Forsburg, University of Southern California.)

Image Thresholding

FIGURE 10.44 Image in Fig. 10.43(a) segmented using the same procedure as explained in Figs. 10.43(d)–(f), but using a lower value to threshold the absolute Laplacian image.

Global thresholding of ORIGINAL image using Otsu algorithm on histogram of masked image. Only difference from previous slide is that lower threshold for absolute Laplacian was used.

Image Thresholding



a b c

FIGURE 10.45 (a) Image of iceberg. (b) Histogram. (c) Image segmented into three regions using dual Otsu thresholds. (Original image courtesy of NOAA.)

Modify Otsu global thresholding by adding a third class

$$\sigma_B^2 = P_1\left(m_1 - m_G\right)^2 + P_2\left(m_2 - m_G\right)^2 + P_3\left(m_3 - m_G\right)^2$$

Compute the between class variance $\sigma_B^2(k_1,k_2)$ for
$k_1=0,\dots,L-1$ and $k_2=0,\dots,L-1$
Otsu optimum thresholds k_1,k_2 when $\sigma_B^2(k_1,k_2)$ is maximum

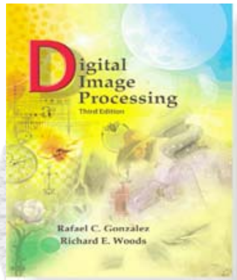


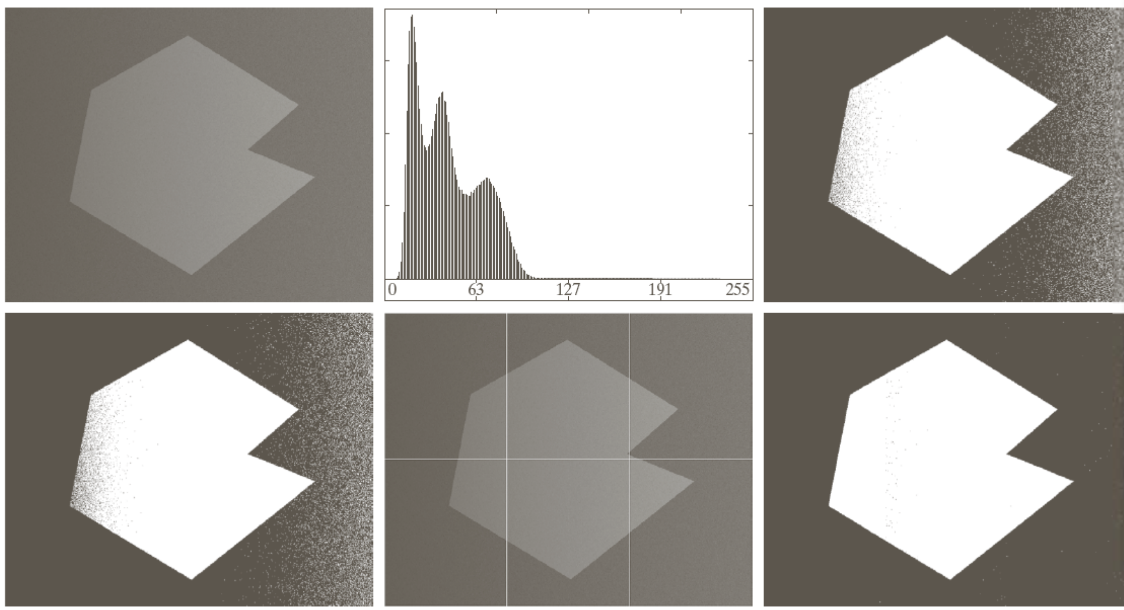
Image Thresholding

Original dark, noisy image

Global thresholding calculating T=0.5*(μ_1+μ_2) until ΔT less than some ε

Global thresholding using Otsu algorithm

Subimage thresholding using Otsu algorithm



Subdivide original image into six subimages and apply Otsu algorithm to histogram of each subimage (next page)

a b c
d e f

FIGURE 10.46 (a) Noisy, shaded image and (b) its histogram. (c) Segmentation of (a) using the iterative global algorithm from Section 10.3.2. (d) Result obtained using Otsu's method. (e) Image subdivided into six subimages. (f) Result of applying Otsu's method to each subimage individually.

Image Thresholding

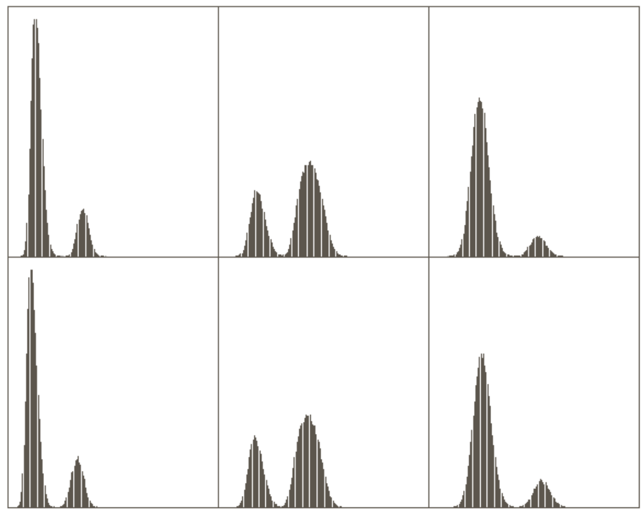


FIGURE 10.47 Histograms of the six subimages in Fig. 10.46(e).

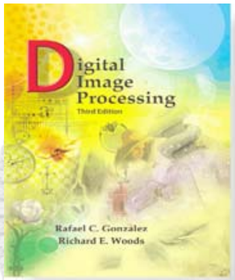


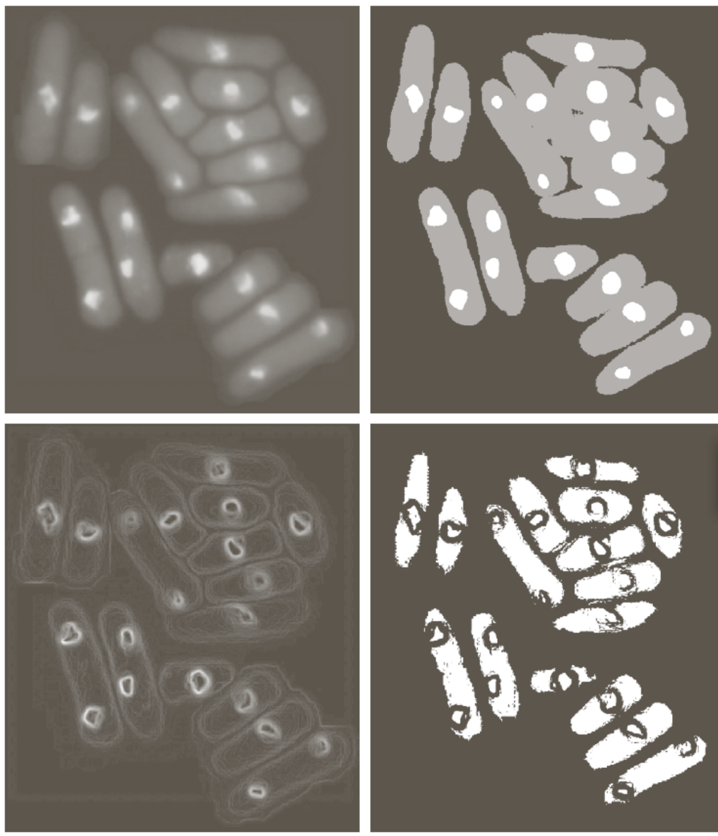
Image Thresholding

Original image

Global thresholding using dual threshold Otsu method

Local standard deviations computed for a moving 3x3 mask

Local thresholding



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.

Predicate (true or false) using local (3x3) std dev

$$g(x,y)=\begin{cases}1 & if \;\; f(x,y)>30\sigma(x,y) \;\; AND \;\; f(x,y)>1.5m_G \\ 0 & otherwise\end{cases}$$

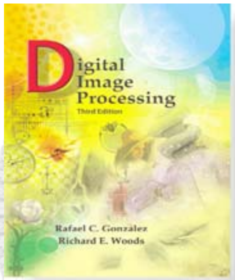
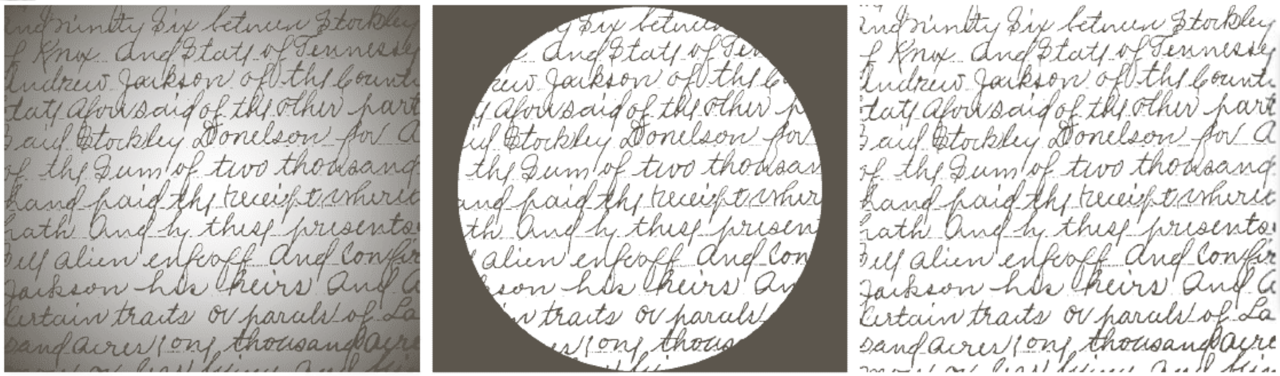


Image Thresholding

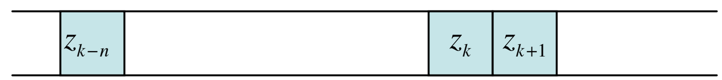


Local thresholding using n=20 moving average

Original image with spot illumination

Global thresholding using Otsu method

$$g(x,y)=\begin{cases}1 & if\ \ f(x,y)>0.5m_{xy}\\ 0 & otherwise\end{cases}$$



Use local moving average where $m(1)=z_1/n$

$$m_{xy}=m(k+1)=\frac{1}{n}\sum_{i=k+2-n}^{k+1} z_i = m(k)+\frac{1}{n}\left(z_{k+1}-z_{k-n}\right)$$

z_k is the intensity point at scanning sequence k (wraps around image)

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.

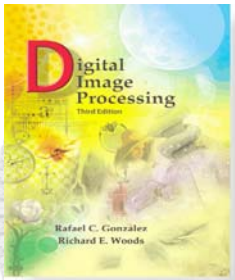


Image Thresholding

Original image with spot illumination

Global thresholding using Otsu method

Local thresholding using n=20 moving average

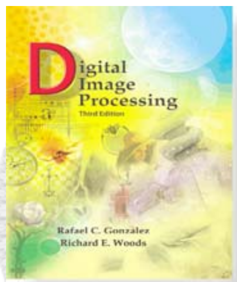
$$g(x,y)=\begin{cases}1 & \text{if } f(x,y)>0.5m_{xy}\\ 0 & \text{otherwise}\end{cases}$$

Use local moving average where $m_1=z_1/n$

$$m_{xy}=m(k+1)=\frac{1}{n}\sum_{i=k+2-n}^{k+1} z_i = m(k)+\frac{1}{n}\left(z_{k+1}-z_{k-n}\right)$$

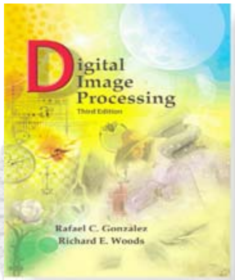
a b c

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



Region Segmentation

- Let f(x,y) be the image, S(x,y) is a binary seed image with 1's at the location of seed points, and Q(x,y) is a predicate function
- Find all connected components in S(x,y) and erode each connected component to one pixel.
- Form an image f_Q such that at each point f(x,y)=1 if Q(x,y) is true, else f(x,y)=0
- Let *g* be an image formed by appending to each seed point in S all 1-valued points in f_Q that are 8-connected to that seed point
- Label each connected component in *g* with a unique label (e.g., 1, 2, 3, …) This is the segmented image



Region Segmentation

Original image

1. Seed image from high percentile thresholding

2. Erode seed image

3. |original-seed| difference image

4. Difference image using dual thresholds $T_1=68$, $T_2=126$

5. Difference image using only the lowest threshold.

6. Region growing from seed image (single threshold) using difference image and 8-connectivity

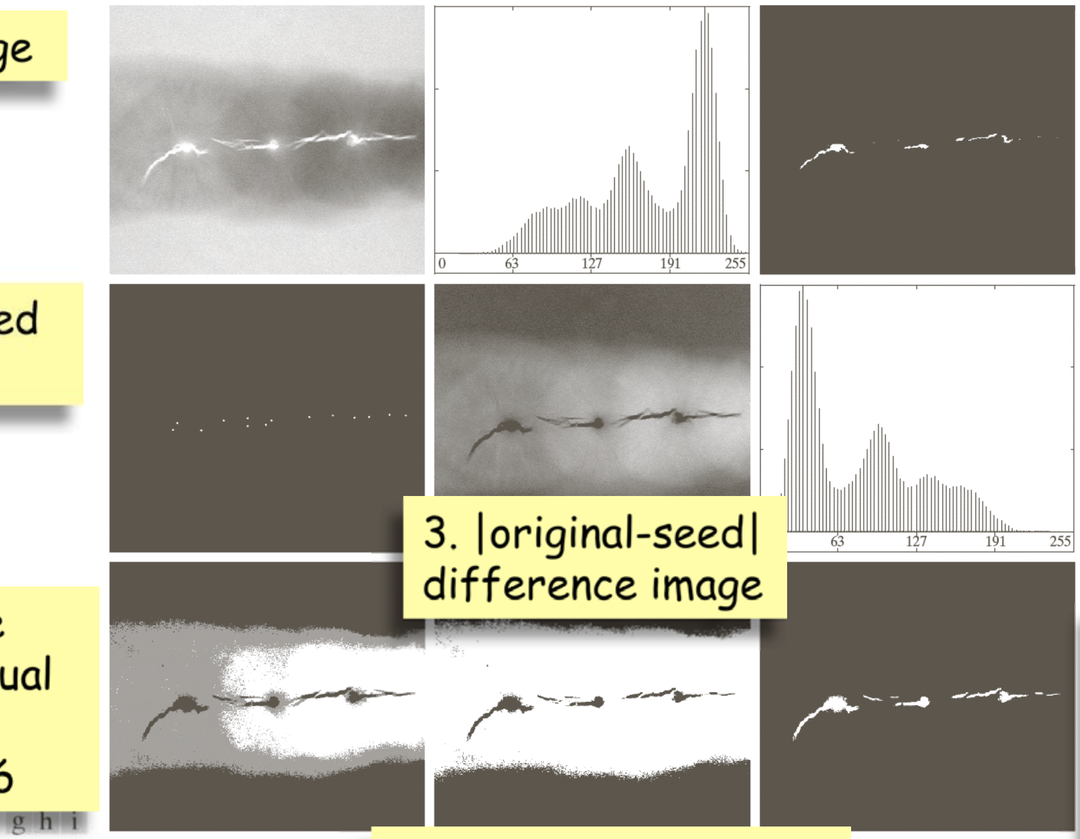
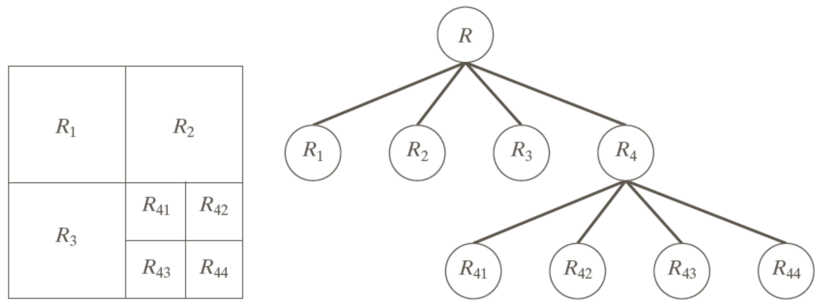


FIGURE 10.51 (a) X-ray image of … ed image. (d) Final see…
(the points were enlarged for clari… een (a) and (c). (f) His…
of (e). (g) Difference image thres… e image thresholded …
smallest of the dual thresholds. (i) … ving. (Original image courtesy
of X-TEK Systems, Ltd.)

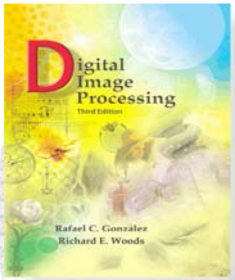
Quadtree Segmentation



a b

FIGURE 10.52 (a) Partitioned image. (b) Corresponding quadtree. *R* represents the entire image region.

An alternative to having a seed image and region growing is to partition an image into sub-images. This process continues until each region has a uniform predicate. The process then merges adjacent regions with similar predicates. This is sometimes called quadtree segmentation.



Quadtree Segmentation

566x566 pixel image

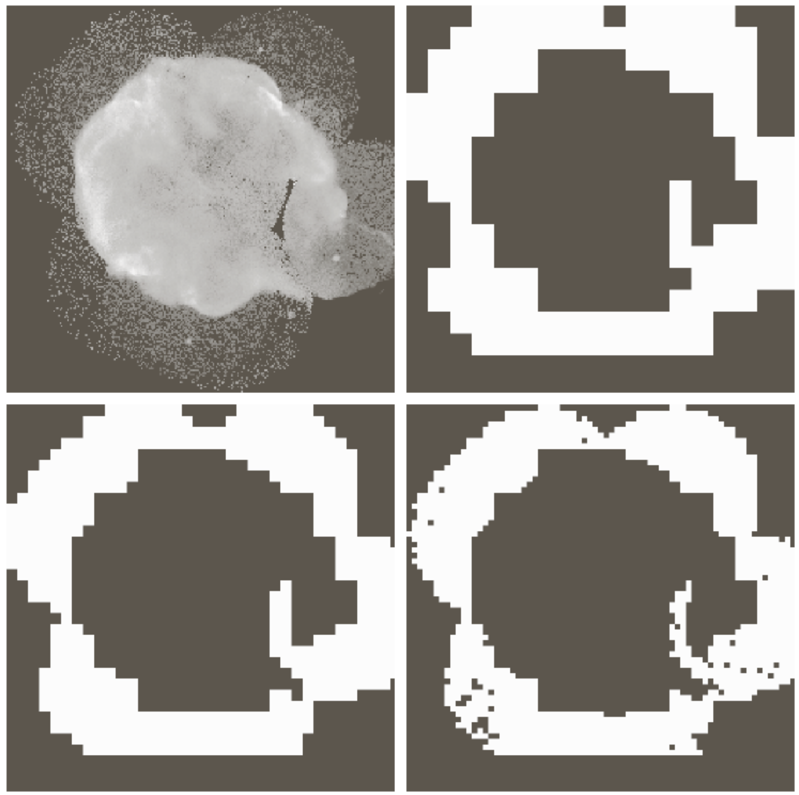
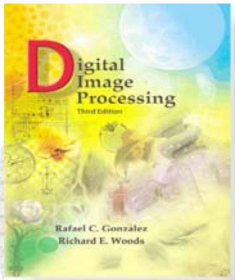


FIGURE 10.53 (a) Image of the Cygnus Loop supernova, taken in the X-ray band by NASA's Hubble Telescope. (b)–(d) Results of limiting the smallest allowed quadregion to sizes of 32×32, 16×16, and 8×8 pixels, respectively. (Original image courtesy of NASA.)

Quadtree segmentation of image using minimum region sizes of 32x32, 16x16, and 8x8 pixels. Predicate is TRUE if $\sigma > a$ and $0 < m < b$



Watershed Segmentation

Original gray scale image

Topographical plot of original image

Punch holes in bottom of each region and flood.

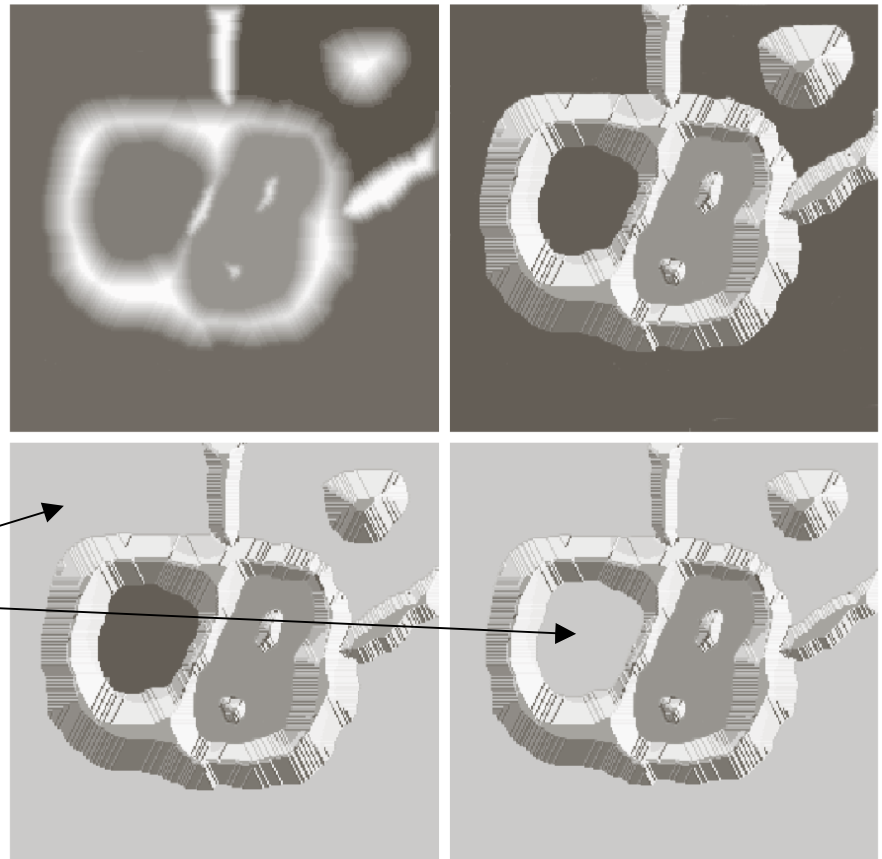
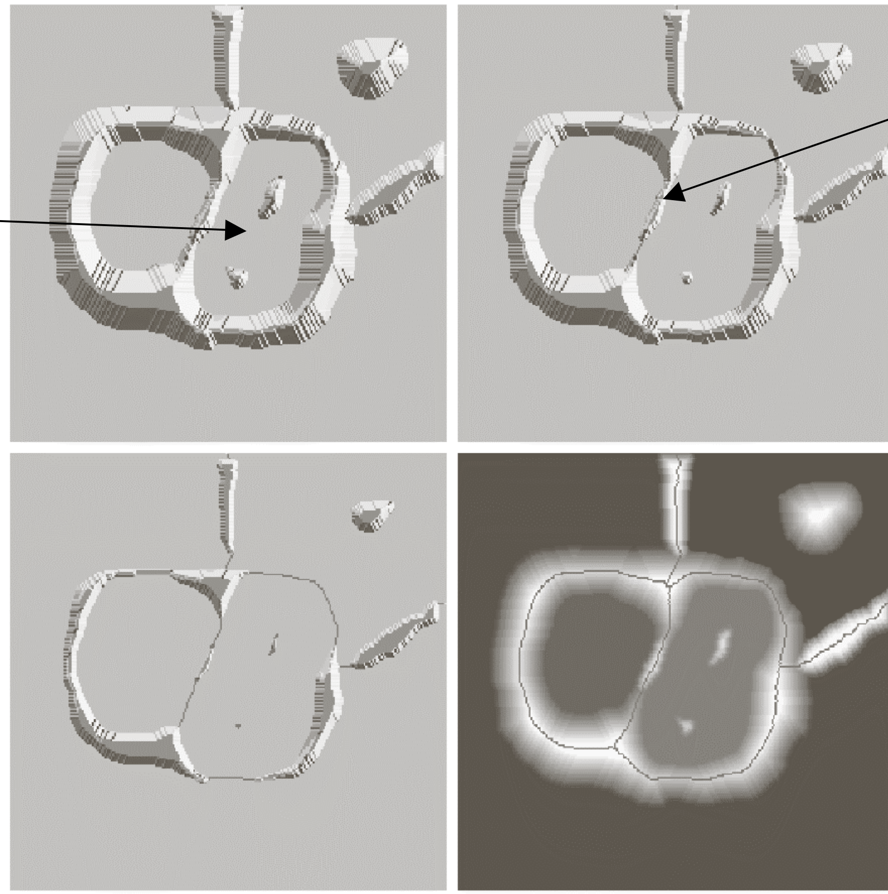


FIGURE 10.54 (a) Original image. (b) Topographic view. (c)–(d) Two stages of flooding.

Watershed Segmentation

Punch another region and continue flooding.

Start building dam between regions

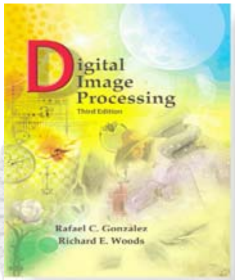


Continue building dams between regions

Final watershed

e f
g h

FIGURE 10.54 *(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.)



Watershed Segmentation

- Mi is the set of coordinates of points in the regional minima of an image g(x,y)
- T[n]={(s,t)|g(s,t)<n} is the set of all points lying below the n plane
- $C_n(M_i)$ denotes the set of coordinates of points associated with minimum M_i that are flooded at stage n.
- $C_n(M_i) \cap T[n]$ restricts the points associated with M_i to those less than n at that stage of the flooding
- The union of flooded catchment basements is $C[n] = \bigcup_{i=1}^{R} C_n(M_i)$
- Let q be in connected component in C[n].
  - If q $\cap$C[n-1] is empty do nothing
  - If q $\cap$C[n-1] contains one connected component of C[n-1] then incorporate into C[n-1]
  - If q $\cap$C[n-1] is contains two or more connected components of C[n-1] build a dam by dilating q $\cap$C[n-1]

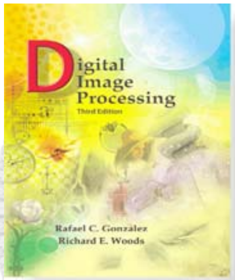
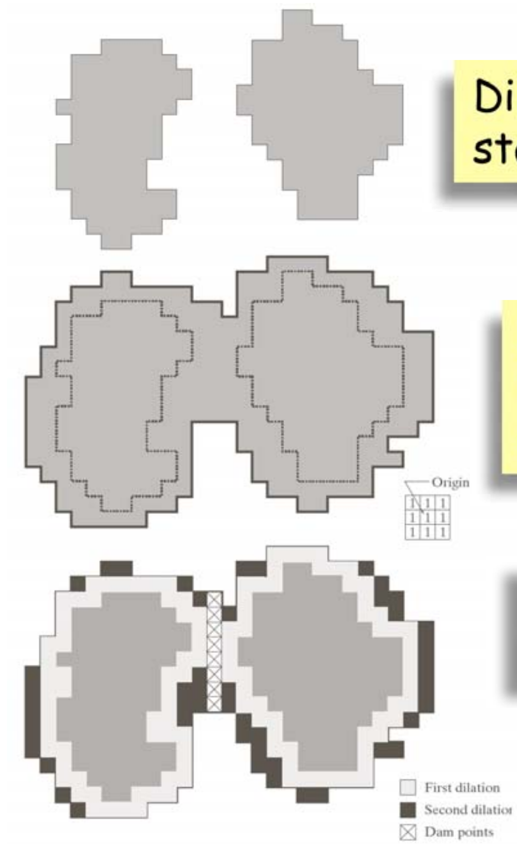


Image Segmentation



Dilating from lowest points at stage n-1

Dam construction begins when dilations overlap at stage n

Dam constructed between basins

a
b c
d

FIGURE 10.55 (a) Two partially flooded catchment basins at stage $n - 1$ of flooding. (b) Flooding at stage n, showing that water has spilled between basins. (c) Structuring element used for dilation. (d) Result of dilation and dam construction.

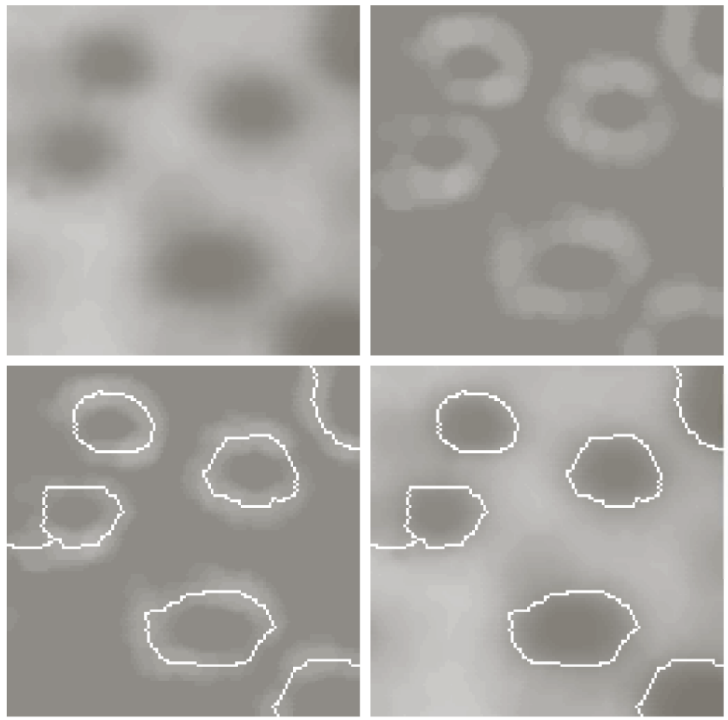
Image Segmentation

Original image

Gradient image

Watershed lines in gradient image

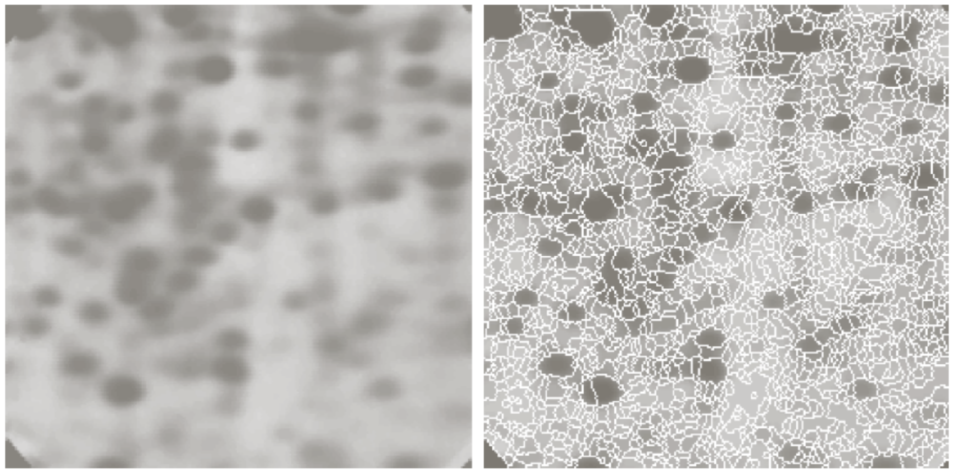
Watershed lines on top of original image



a b
c d

FIGURE 10.56 (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.)

Image Segmentation



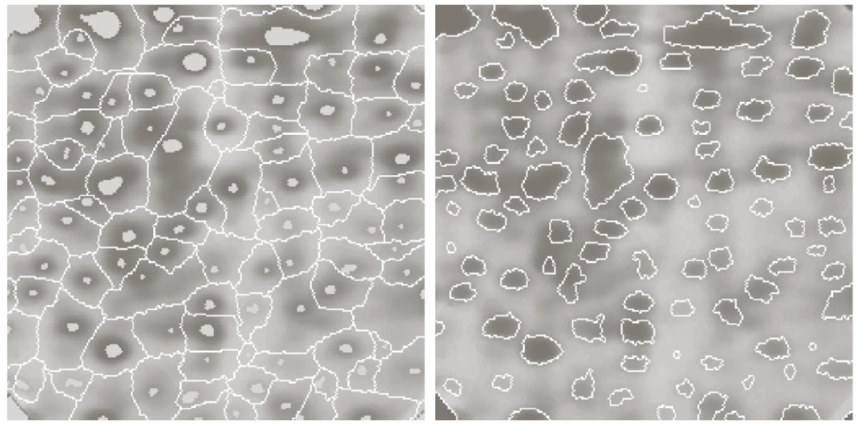
a b

FIGURE 10.57 (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.)

Direct application of watershed segmentation typically results in oversegmentation, i.e., too many regions

Image Segmentation

We can use other segmentation techniques to define markers which will control the segmentation.



Internal markers (light gray regions) define starting points and external markers (gray lines) limit segmentation.

Resulting segmented image.

a b

FIGURE 10.58 (a) Image showing internal markers (light gray regions) and external markers (watershed lines). (b) Result of segmentation. Note the improvement over Fig. 10.47(b). (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

10.57(*b*)

Motion Segmentation

Accumulative Difference Image (ADI) for a moving rectangle.

Positive ADI

Negative ADI

$$A_k(x,y)=\begin{cases}A_{k-1}(x,y)+1 & \text{if } \left|R(x,y)-f(x,y,k)\right|>T\\ A_{k-1}(x,y) & \text{otherwise}\end{cases}$$

The accumulator is incremented if the absolute difference between the reference frame and each successive frame is above a threshold. We can also have positive and negative accumulator images.

$$P_k(x,y)=\begin{cases}P_{k-1}(x,y)+1 & \text{if } \left|R(x,y)-f(x,y,t_k)\right|>T\\ P_{k-1}(x,y) & \text{otherwise}\end{cases}$$

$$N_k(x,y)=\begin{cases}N_{k-1}(x,y)+1 & \text{if } \left|R(x,y)-f(x,y,t_k)\right|\le T\\ N_{k-1}(x,y) & \text{otherwise}\end{cases}$$

a b c

FIGURE 10.59 ADIs of a rectangular object moving in a southeasterly direction. (a) Absolute ADI. (b) Positive ADI. (c) Negative ADI.

Image Segmentation

a b c

FIGURE 10.60 Building a static reference image. (a) and (b) Two frames in a sequence. (c) Eastbound automobile subtracted from (a) and the background restored from the corresponding area in (b). (Jain and Jain.)

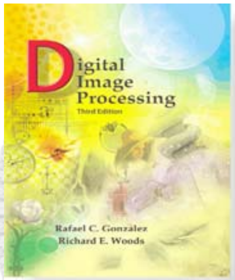


Image Segmentation

There is a small moving object with a 9 pixel Gaussian distribution moving with v_x=0.5 and v_y=0.5 pixel/frame. This is one of 32 frames.

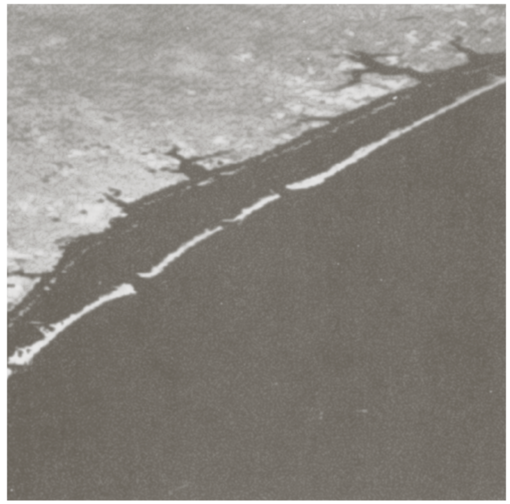


FIGURE 10.61 LANDSAT frame. (Cowart, Snyder, and Ruedger.)

<u>Basic concept:</u> Single one pixel object moving against a uniform background. v_x=1 pixel/frame

1. Project image onto x-axis (sum columns)
2. At t=0 multiply columns of projection array by $e^{j2\pi a x \Delta t}$, x=0,1,2,… where a is a positive integer and Δt is the time interval between frames
3. At t=1 do the same thing except object has moved to x′+1

This gives a accumulator array of zeroes except for the moving object projection.

If the velocity is constant the projection is $e^{j2\pi a(x'+t)\Delta t}=\cos[2\pi a(x'+t)\Delta t]+j\sin[2\pi a(x'+t)\Delta t]$, i.e., projections of moving objects give single frequency sinusoids (for constant velocity)

Image Segmentation

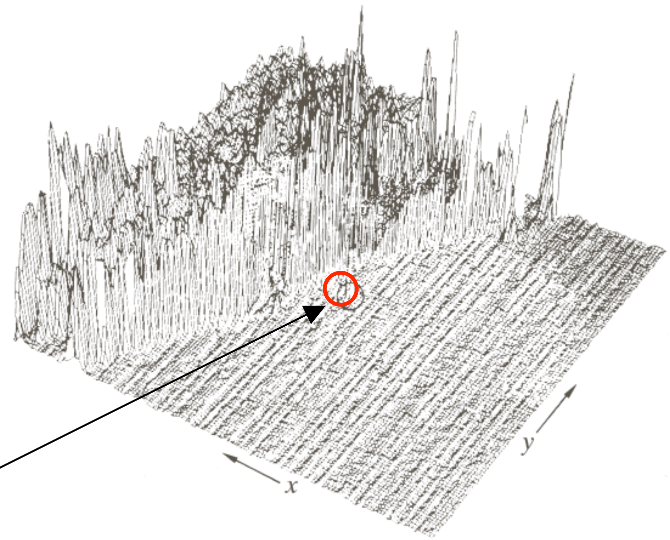


FIGURE 10.62 Intensity plot of the image in Fig. 10.61, with the target circled. (Rajala, Riddle, and Snyder.)

Moving object

Intensity plot of single frame.

Image Segmentation

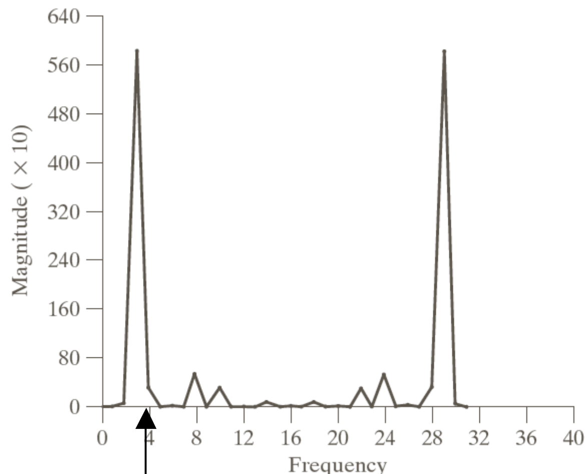


FIGURE 10.63 Spectrum of Eq. (10.6-8) showing a peak at $u_1 = 3$. (Rajala, Riddle, and Snyder.)

$a_1=6$

The a's are selected to prevent aliasing in the frequency domain. A rule of thumb is to select a as the integer closest to u_{max}/v_{max} where v_{max} is the maximum velocity and u_{max} is related to the maximum number of frames/second.

$u_1=3$

x-velocity of unknown object is $v_1 = u_1/a_1=3/6=0.5$ pixel/frame

Image Segmentation

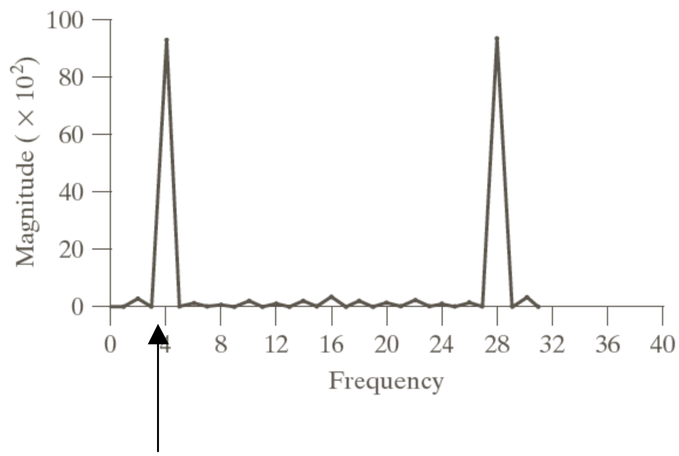


FIGURE 10.64 Spectrum of Eq. (10.6-9) showing a peak at $u_2 = 4$. (Rajala, Riddle, and Snyder.)

$a_2=4$

$u_2=4$

y-velocity of moving object is then
$v_2 = u_2/a_2 = 4/4 = 1$ pixel/frame