

# Image Segmentation

---

Jayanta Mukhopadhyay

Dept. of CSE,  
IIT Kharagpur



# Segmentation

- Partitioning image pixels into meaningful non-overlapping sets.
- Let  $R$  be the entire spatial region occupied by the image.
- Segmentation is a process to partition  $R$  into  $n$  sub-regions  $R_1, R_2, \dots, R_n$  so that
  - $R_i$  is a connected set,  $i=1,2,\dots,n$

$$\bigcup_{i=1}^n R_i = R \quad R_i \cap R_j = \phi, \forall i, j, i \neq j$$

A logical predicate over  $R_i$

$$Q(R_i) = TRUE, \forall i$$

$$Q(R_i \cup R_j) = FALSE, \text{ if } i \text{ \& } j \text{ adjacent}$$



# Segmentation Approaches

- Edge based
  - Checks discontinuities in the functional space
  - Assumption: Boundaries of regions sufficiently different from each other and also from the background.
- Region based
  - Checks similarity among connected set of pixels
    - Similar according to a predefined set of criteria.



# Edge detection: Gradient Operations

Consider the image as a 2D function:  $f(x, y)$



$$\frac{\partial f(x, y)}{\partial x} = f(x+1, y) - f(x, y)$$

$$\frac{\partial f(x, y)}{\partial y} = f(x, y+1) - f(x, y)$$

$$\nabla f(x, y) = \frac{\partial f(x, y)}{\partial x} \hat{i} + \frac{\partial f(x, y)}{\partial y} \hat{j}$$



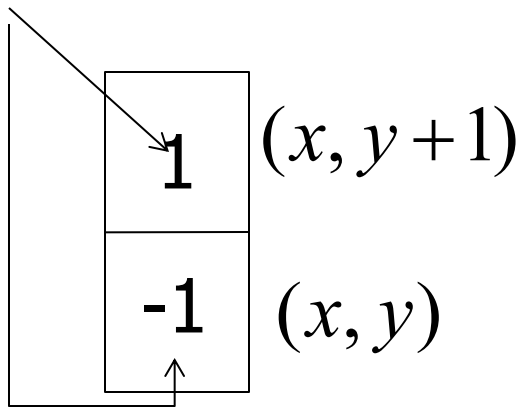


# Computation with mask

-1	1
----	---

$(x, y)$   $(x+1, y)$

Weights



1. Scan the image top to bottom and left to right.
2. At every point  $(x, y)$  place the mask and compute the weighted sum.

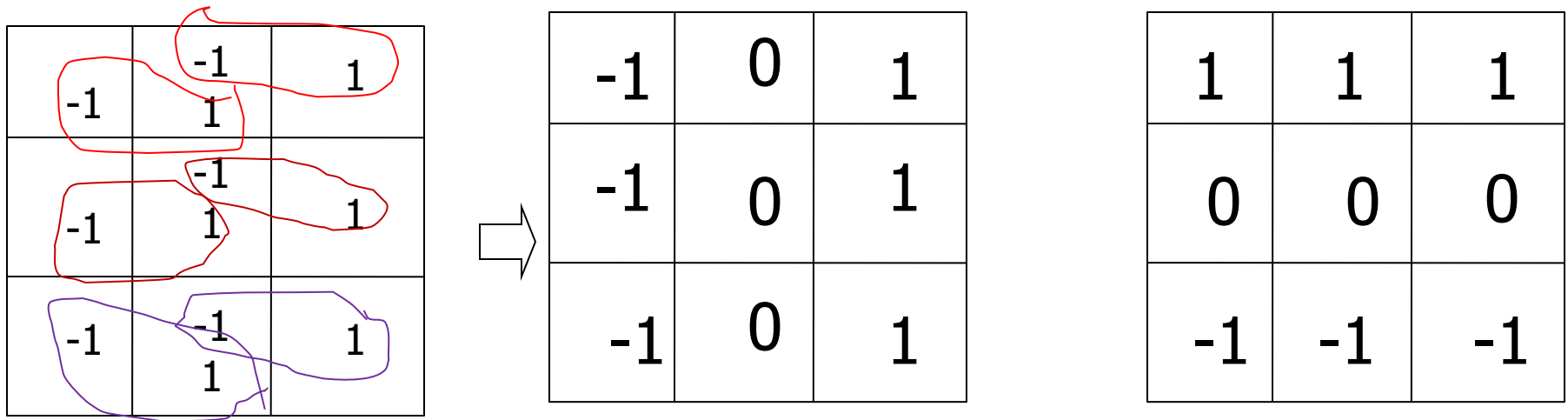
$$g(x, y) = (1) \cdot f(x, y + 1) + (-1) \cdot f(x, y)$$

3. Write the value  $g(x, y)$  at  $(x, y)$  pixel position of the processed image.



# Robust gradient computation

Averaging neighboring gradient values



Prewitt operator



(6 times of the gradient value in any direction)

# Robust gradient computation

Weighted average of neighboring gradient values

1x	-1	$\frac{-1}{1}$	1	⇒	-1	0	1	1	2	1
2x	-1	$\frac{-1}{1}$	1		-2	0	2	0	0	0
1x	-1	$\frac{-1}{1}$	1		-1	0	1	-1	-2	-1

Sobel operator

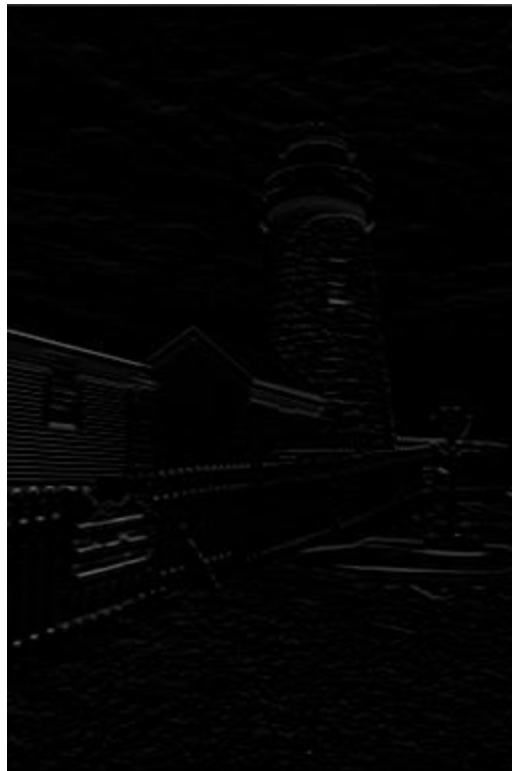
(8 times of the gradient value in any direction)



# Results of gradient operations



Vertical



Horizontal

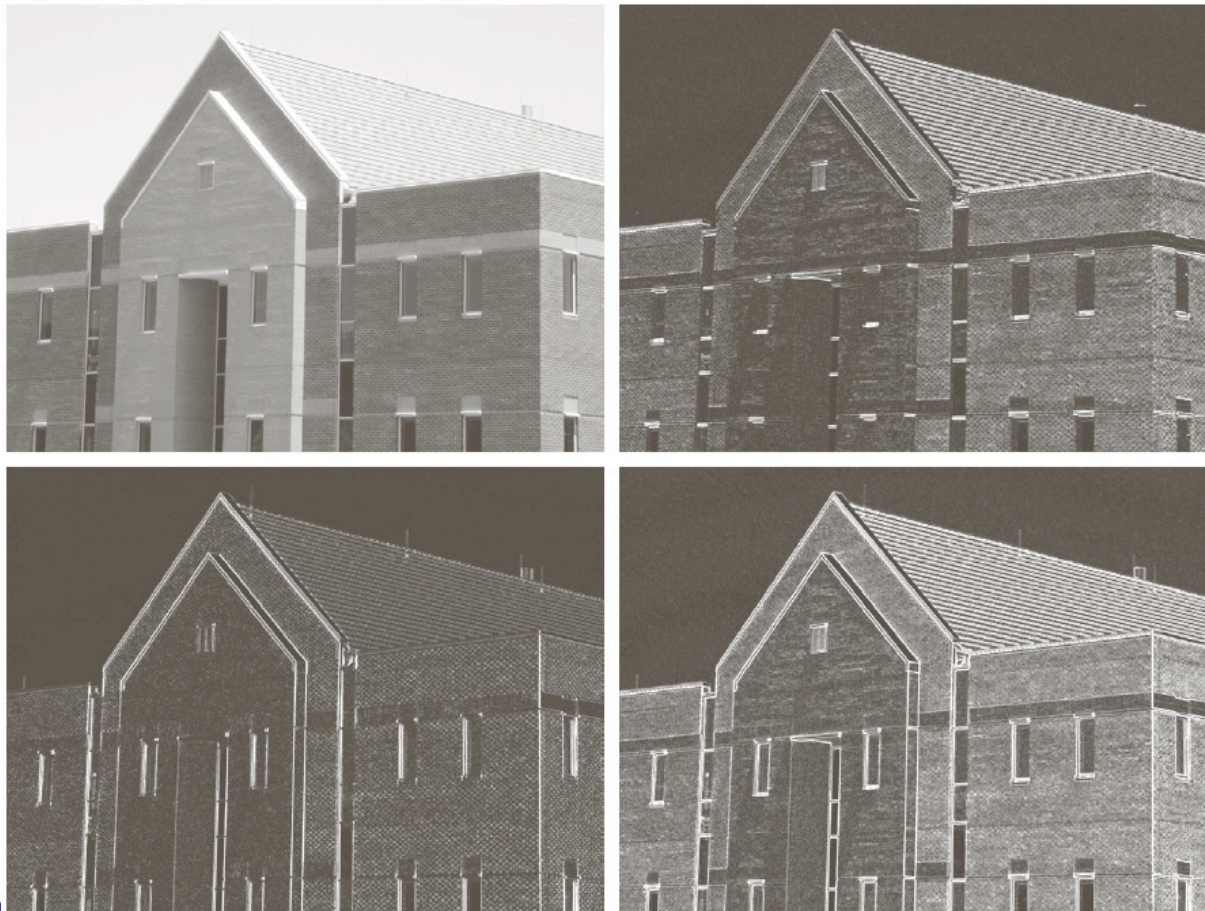


Resultant



# Example

## (Sobel Operator: Horizontal+Vertical)



a	b
c	d

**FIGURE 10.16**

(a) Original image of size  $834 \times 1114$  pixels, with intensity values scaled to the range  $[0, 1]$ .

(b)  $|g_x|$ , the component of the gradient in the  $x$ -direction, obtained using the Sobel mask in Fig. 10.14(f) to filter the image.

(c)  $|g_y|$ , obtained using the mask in Fig. 10.14(g).

(d) The gradient image,  $|g_x| + |g_y|$ .

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1





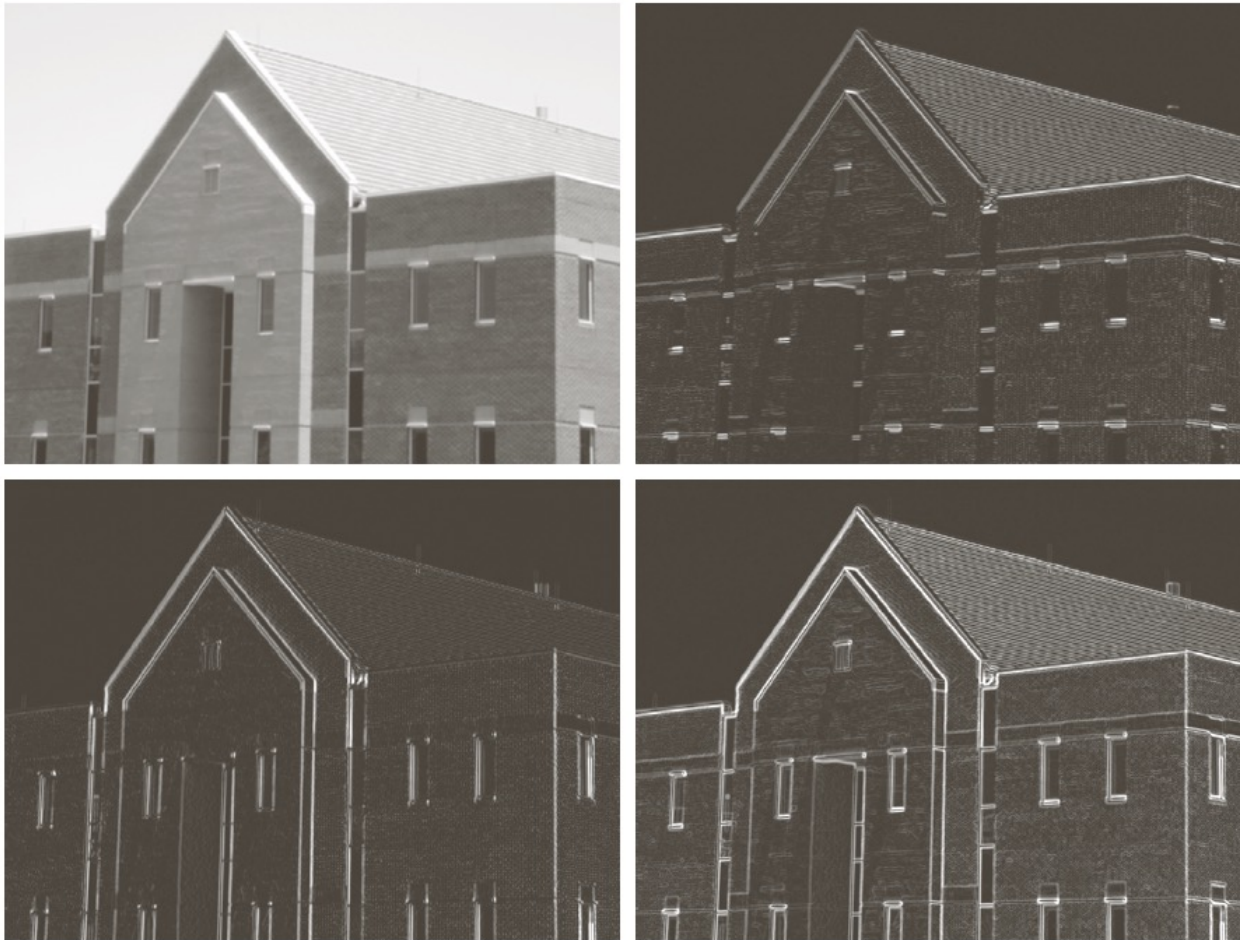
# Example (Gradient Image)



**FIGURE 10.17**  
Gradient angle  
image computed  
using  
Eq. (10.2-11).  
Areas of constant  
intensity in this  
image indicate  
that the direction  
of the gradient  
vector is the same  
at all the pixel  
locations in those  
regions.



# Example (Post Smoothing)



a	b
c	d

**FIGURE 10.18**

Same sequence as in Fig. 10.16, but with the original image smoothed using a  $5 \times 5$  averaging filter prior to edge detection.





# Example (Sobel: Comparison)



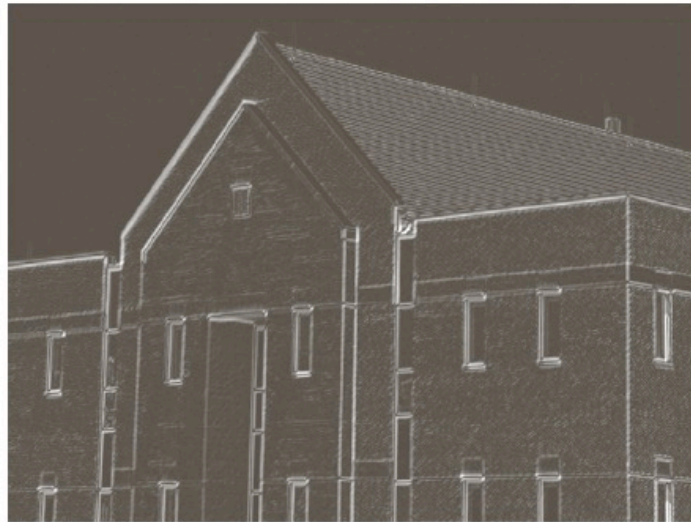
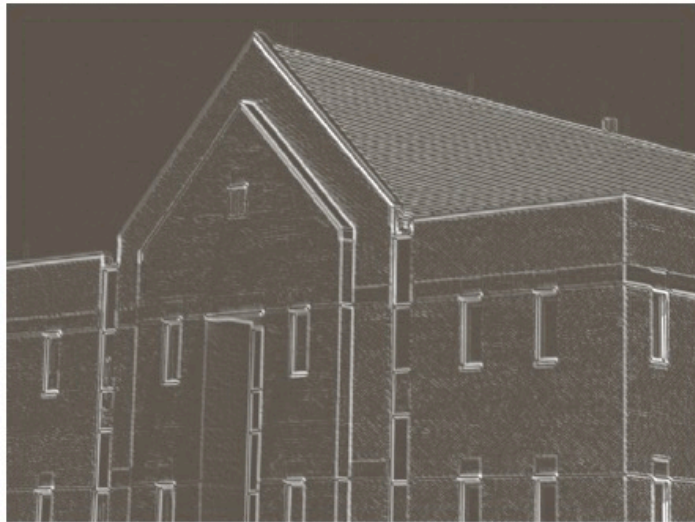
- Without Smoothing

- With Smoothing





# Example (Sobel Diagonal)



a b

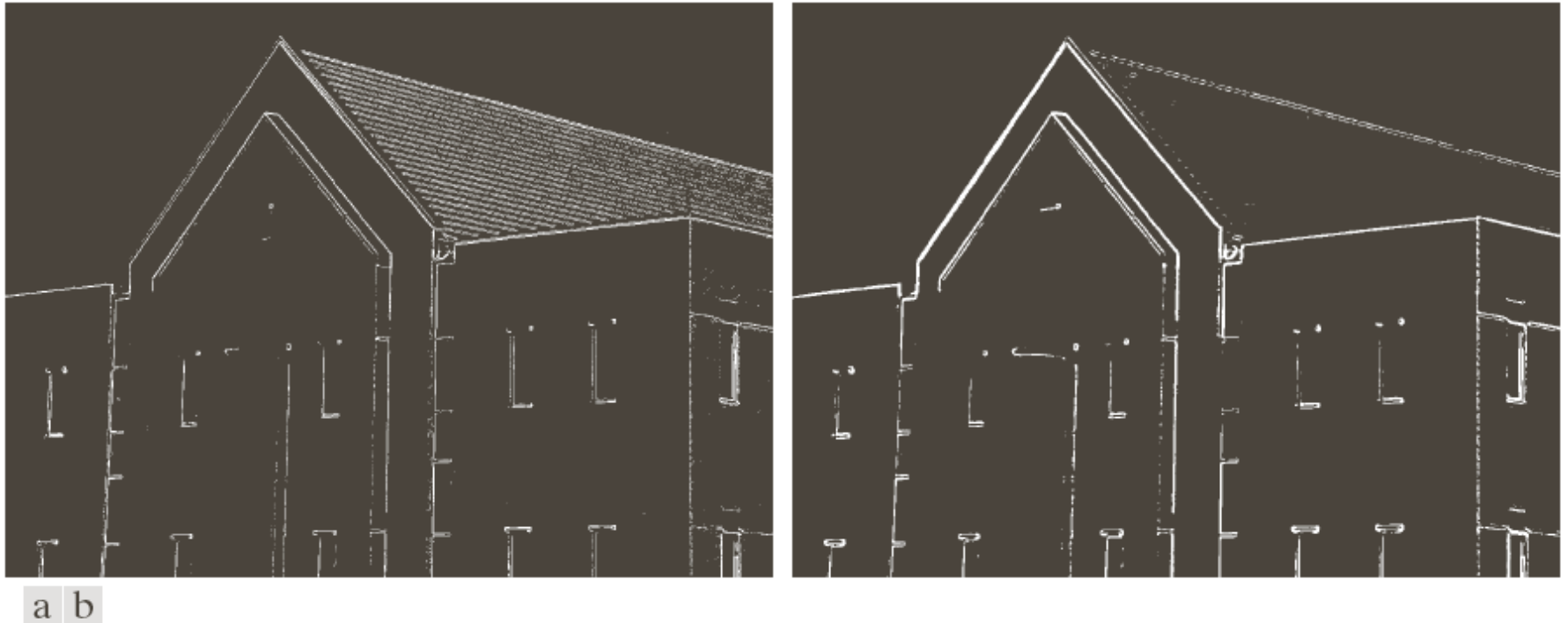
**FIGURE 10.19**  
Diagonal edge detection.  
(a) Result of using the mask in Fig. 10.15(c).  
(b) Result of using the mask in Fig. 10.15(d). The input image in both cases was Fig. 10.18(a).

0	1	2
-1	0	1
-2	-1	0

-2	-1	0
-1	0	1
0	1	2



# Example



**FIGURE 10.20** (a) Thresholded version of the image in Fig. 10.16(d), with the threshold selected as 33% of the highest value in the image; this threshold was just high enough to eliminate most of the brick edges in the gradient image. (b) Thresholded version of the image in Fig. 10.18(d), obtained using a threshold equal to 33% of the highest value in that image.

# Higher order gradients!

$$\frac{\partial f(x, y)}{\partial x} = f(x+1, y) - f(x, y)$$

1	-2	1
(x-1,y)	(x,y)	(x+1,y)

$$\frac{\partial^2 f(x, y)}{\partial x^2} = \frac{\partial f(x, y)}{\partial x} - \frac{\partial f(x-1, y)}{\partial x}$$

$$\approx f(x+1, y) - f(x, y) - (f(x, y) - f(x-1, y))$$

$$= f(x+1, y) - 2f(x, y) + f(x-1, y)$$

$$\frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$



0	1	0
1	-4	1
0	1	0

OR

1	0	1
0	-4	0
1	0	1

Laplacian Operator



# Weighted combination!

Laplacian Operator

$$\frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}$$



1 x

0	1	0
1	-4	1
0	1	0

+

4 x

1	0	1
0	-4	0
1	0	1

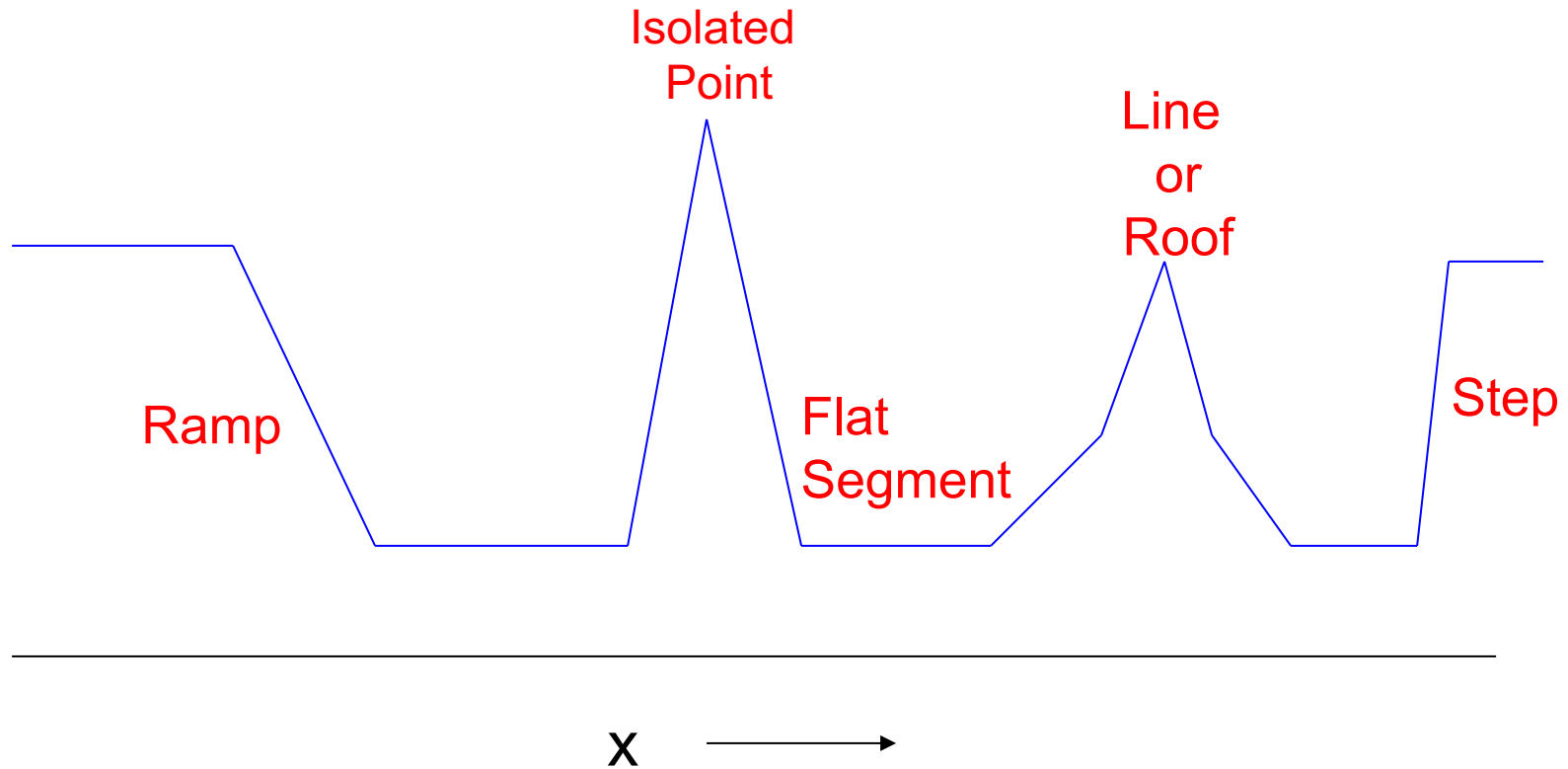


$\frac{1}{5}$  x

4	1	4
1	-20	1
4	1	4



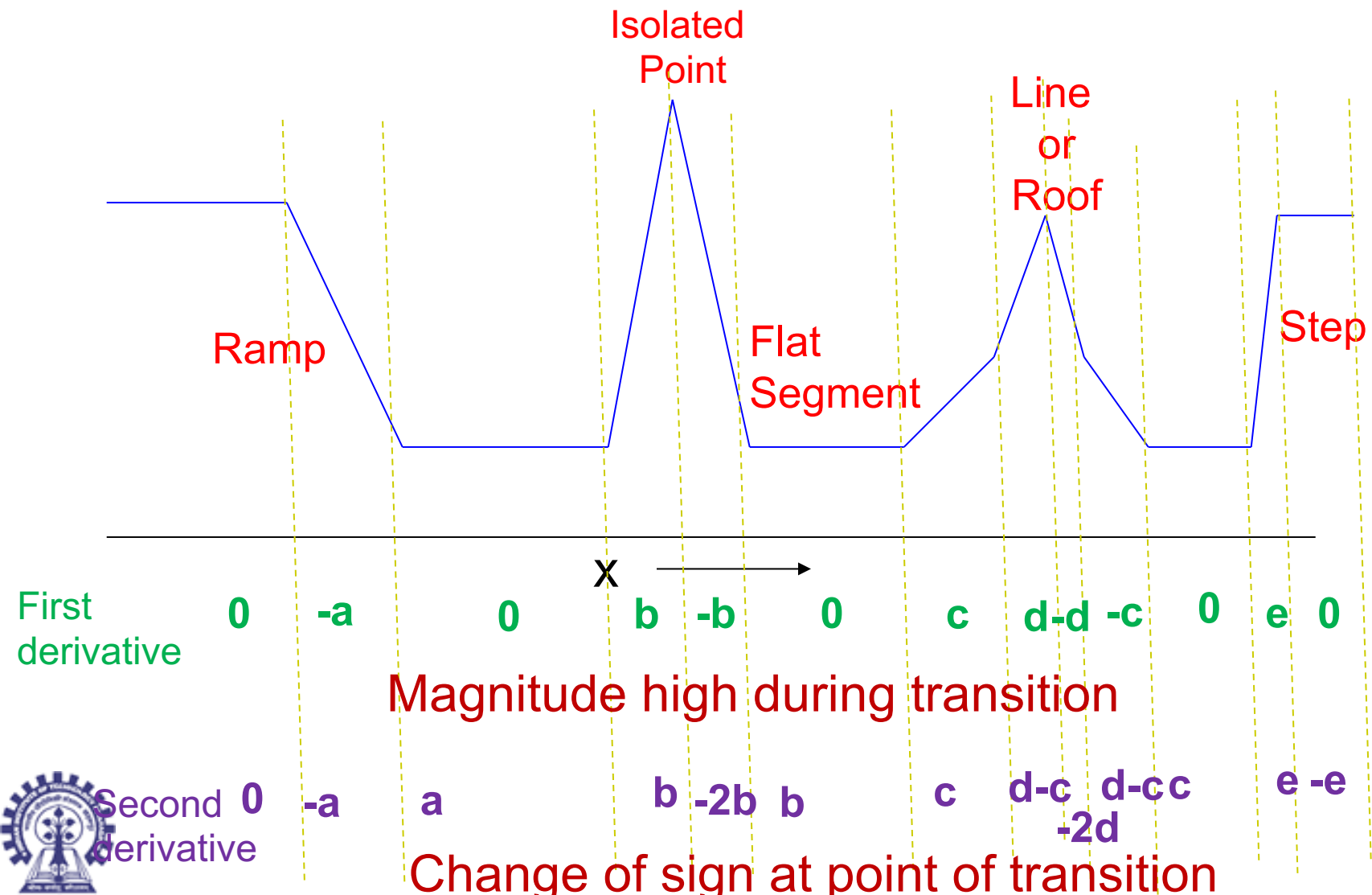
# Types of discontinuities in edges



Horizontal Intensity profile



# First and second derivatives at discontinuities in edges



# Edge detection from first and second order derivatives

- First order derivative
  - Generally produce thicker edges
  - Use magnitude of the derivative.
- Second order derivative
  - Stronger response to fine details, such as, thin lines, isolated points and noise.
    - More sensitive to noise
  - ‘double edge response’ at the point of discontinuity
    - Produces two values for every edge.
    - zero crossing, change of sign, better localization
    - Sign changes could be used to qualify transition (dark to light, or reverse).



# Effect of noise and blur

- Effect of noise
  - Introducing spurious discontinuities
    - Many local spikes
  - Introduces errors in estimating derivatives
    - Affects second derivative more
  - Weak edges may get suppressed
- Effect of blur
  - Step edges tend to become ramp edges
  - Slope of ramp edges decreases
    - Inversely proportional to amount of blur
  - Location of transition shifts.





# Handling noise

- Smooth image
  - Introduces blur due to low pass filtering
- Detect edge points
  - Computes candidate edge points
    - For example thresholding
- Localize edges
  - Select true edge points from the candidates
    - Check the zero-crossings
  - Blur increases with the increase of size of smoothing area (scale)
    - Track edge points in scale space



# The Marr-Hildreth edge detector (1980)

- For robust edge detection in presence of noise.
- Observations
  - Intensity changes are not independent of scale
    - Detection requires edge operator of different sizes
  - Sudden intensity change gives rise to trough or peak in first derivative; or equivalently a zero-crossing in the second derivative
- Two salient features of operators
  - A differential operator
  - Capable of being tuned to act at any desired scale.
    - Large operators for detecting blurry edges
    - Small operators for sharp edges



# The Marr-Hildreth operator (1980)

- Laplacian of Gaussian:

$$G(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

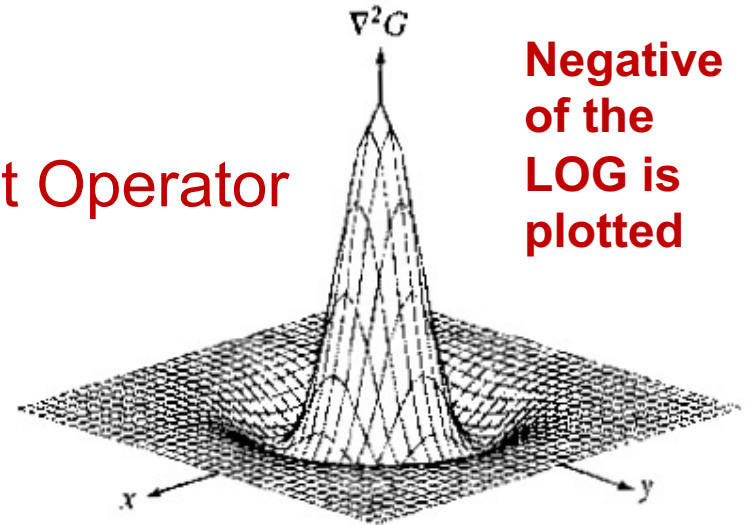
Mexican Hat Operator

$$\nabla^2 G(x, y) = \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2}$$

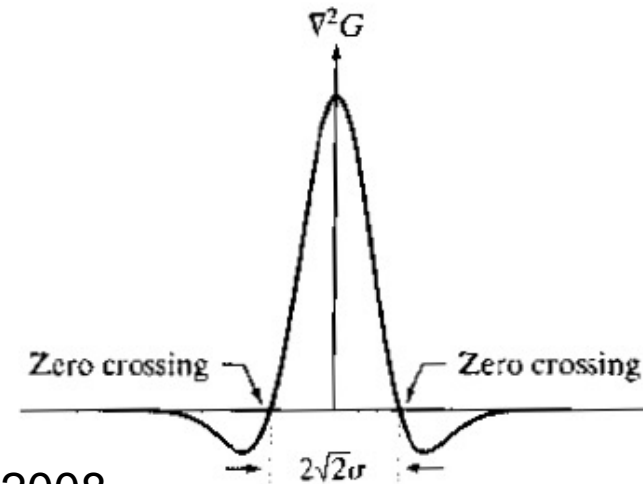
$$\nabla^2 G(x, y) = \left[ \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

Zero crossings occur at  $x^2 + y^2 = 2\sigma^2$

A circle of radius of  $\sqrt{2}\sigma$



Negative  
of the  
LOG is  
plotted



# Typical approximation with integer weight for $\sigma=1$

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0



# Smoothing and differentiation

$$g(x, y) = \nabla^2 [f(x, y) * G(x, y)] = f(x, y) * \nabla^2 G(x, y)$$

Smooth with  $G(x, y)$   
and then apply  
Laplacian



Convolve with  
Laplacian of  
Gaussian

Detect zero-crossings after applying  
the operator to detect edge pixels.



# Marr-Hildreth edge detection algorithm

- Filter the image using  $n \times n$  Gaussian filter.
  - To retain 97% of energy,  $n \geq 6\sigma$
- Compute Laplacian (e.g. using the 3x3 mask for Laplacian operator)
- Detect zero-crossings

OR

- Filter the image using  $n \times n$  LoG filter.
  - To retain the effect of zero-crossing,  $n > 2\sqrt{2}\sigma$
- Detect zero-crossings



# Approximation of LoG by DoG

A kind of BPF.

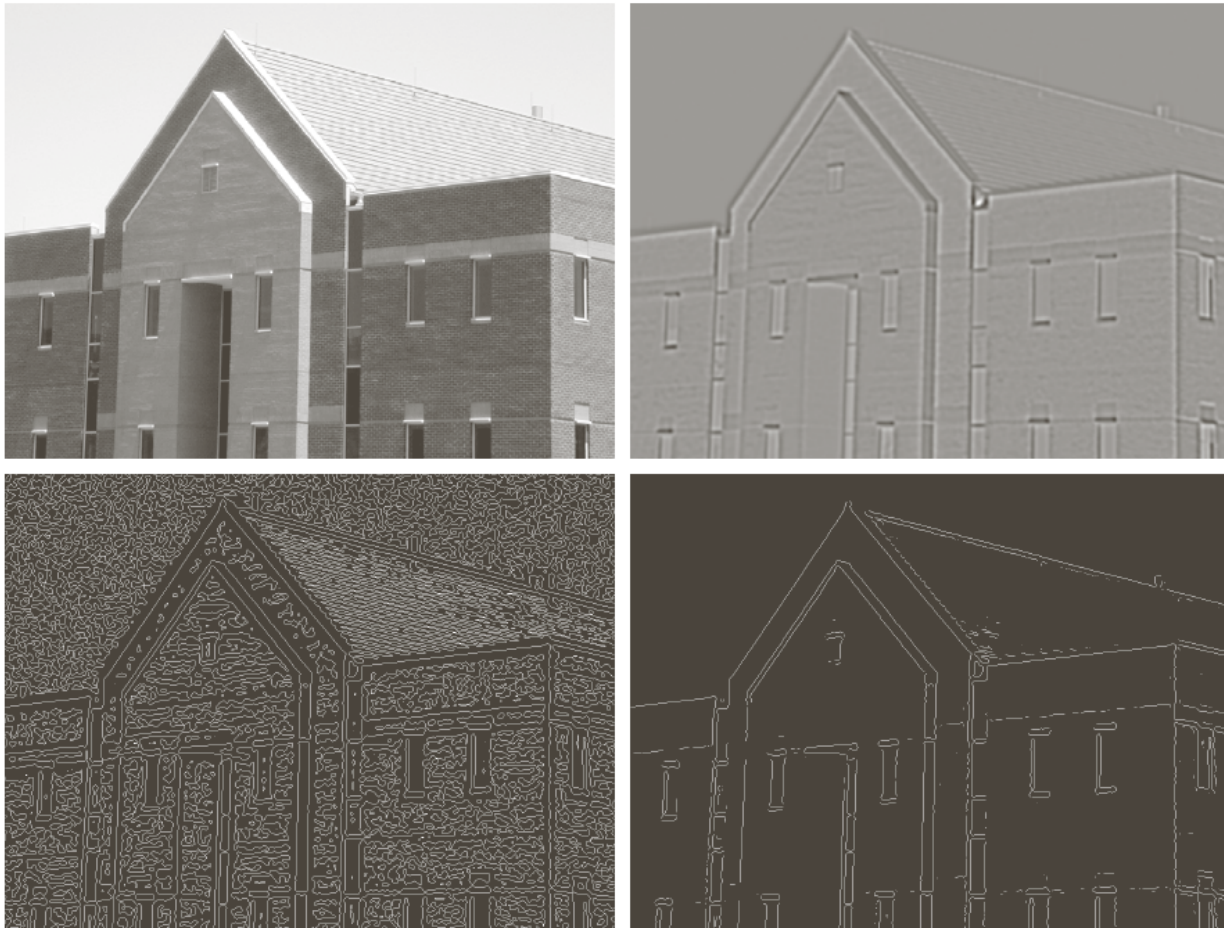
- DoG: 
$$DoG(x, y) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2+y^2}{\sigma_1^2}} - \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}}$$
- Approximates to LoG with the ratio:  $\sigma_1 : \sigma_2 = 1.6:1$
- $\sigma$  of LoG having same zero-crossings:

$$\sigma^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 - \sigma_2^2} \ln \left( \frac{\sigma_1^2}{\sigma_2^2} \right)$$

Certain channels of human vision system are selective w.r.t. freq. and orientation and be modeled by DOG with a ratio of 1.75:1.



# LoG: Example



a	b
c	d

**FIGURE 10.22**

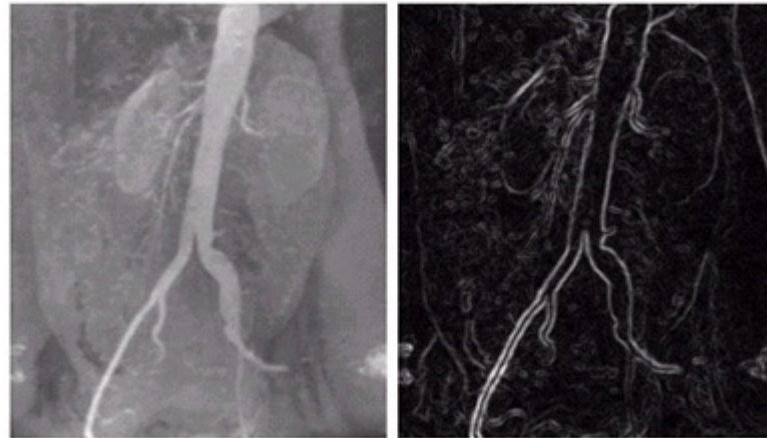
(a) Original image of size  $834 \times 1114$  pixels, with intensity values scaled to the range  $[0, 1]$ . (b) Results of Steps 1 and 2 of the Marr-Hildreth algorithm using  $\sigma = 4$  and  $n = 25$ . (c) Zero crossings of (b) using a threshold of 0 (note the closed-loop edges). (d) Zero crossings found using a threshold equal to 4% of the maximum value of the image in (b). Note the thin edges.



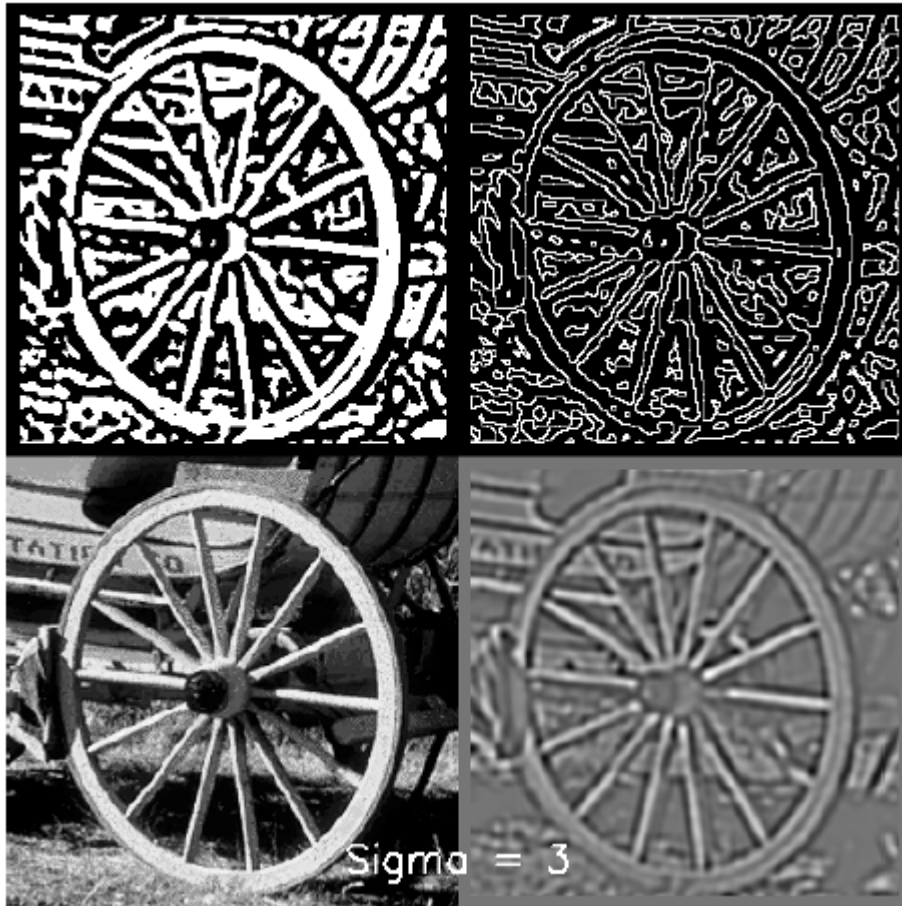


# Example

1. Original image
2. Sobel Gradient
3. LoG
4. Threshold LoG
5. Zero crossing



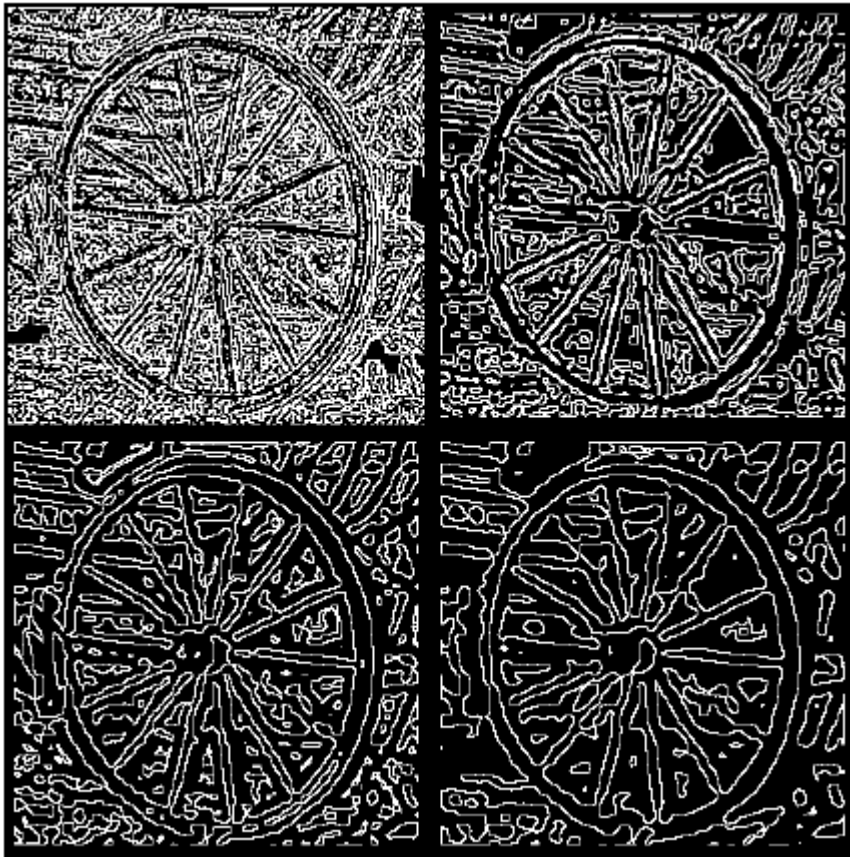
# LoG: Example



- SW: Original
- SE: LoG
- NW: Thresholded
- NE: Zero-Crossings



# LoG: Effect of Sigma



$\sigma = 1$	$\sigma = 2$
$\sigma = 3$	$\sigma = 4$



# The Canny edge detector (1986)

- Three basic objectives
  - Low error rate
    - All edge pixels to be detected
    - No spurious edges
  - Good localization
    - To be as close as possible to true edge points
  - Single edge point response
    - Only one point for each edge point
- Optimization criteria set involving above objectives.
  - Difficult to get closed form optimal solution



# Optimal detector for noisy step edges

- In 1-D
  - Derive responses for an ideal step edge in 1- D contaminated by white Gaussian noise.
  - Optimal numerical solution close to first derivative of a Gaussian.

$$\frac{d e^{-\frac{x^2}{2\sigma^2}}}{dx} = -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$$

- Extension to 2D
  - Analysis required to be repeated along a direction
    - but not known a priori.
  - Compute gradient vector
    - Smooth the image using Gaussian filter
    - Apply Sobel operator
    - Get magnitude  $M(x,y)$  and direction  $\alpha(x,y)$  at each pixel.
  - Edges are thick.

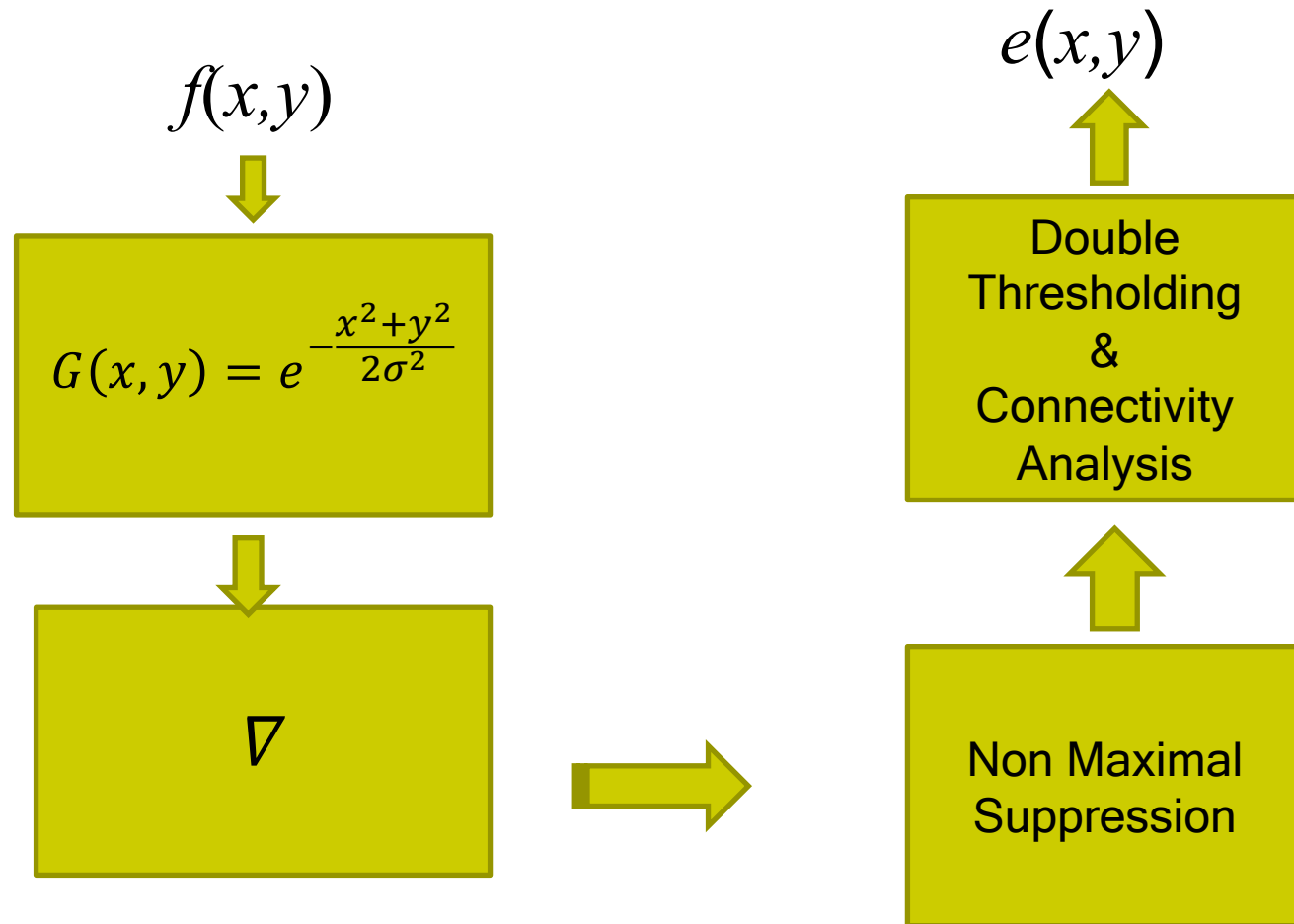


# Localization and thin edges

- Non-maximal suppression
  - Get the discrete direction  $d$  (e.g. 0, 45, 90 and 135 in degrees) closest to  $\alpha(x,y)$ .
  - Set  $M(x,y)=0$ , if it is less than its neighbor along the orthogonal direction of  $d$ .
- Use of two thresholds
  - Higher threshold ( $T_H$ ): low false +ve but high false -ve.
    - Detected edge pixel less erroneous, but misses weaker edges
  - Lower threshold ( $T_L$ ): High false +ve but low false -ve.
    - Detected edge pixel more erroneous, but retains weaker edges
- Hysteresis Thresholding
  - Retain all edge pixels whose  $M(x,y) \geq T_H$ 
    - Category 1
  - Retain an edge pixel with  $T_H > M(x,y) \geq T_L$ , if it is connected to an edge pixel of category 1.



# Canny edge detection pipeline





# Example: Canny



(a) Original



(b) Smoothed



(c) Gradient magnitudes



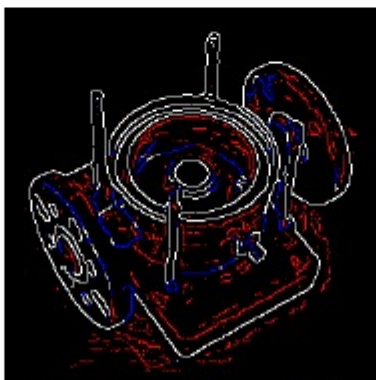
(d) Edges after non-maximum suppression

e)  
White: Strong Edges  
Grey: Weak Edges  
Black: No Edges

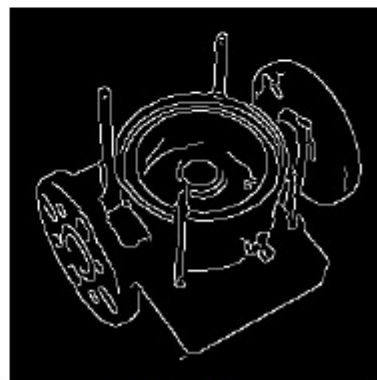


(e) Double thresholding

f)  
White: Valid Edges  
Blue: Chosen Weak Edges  
Red: Rejected Weak Edges



(f) Edge tracking by hysteresis



(g) Final output



14-Sep-11

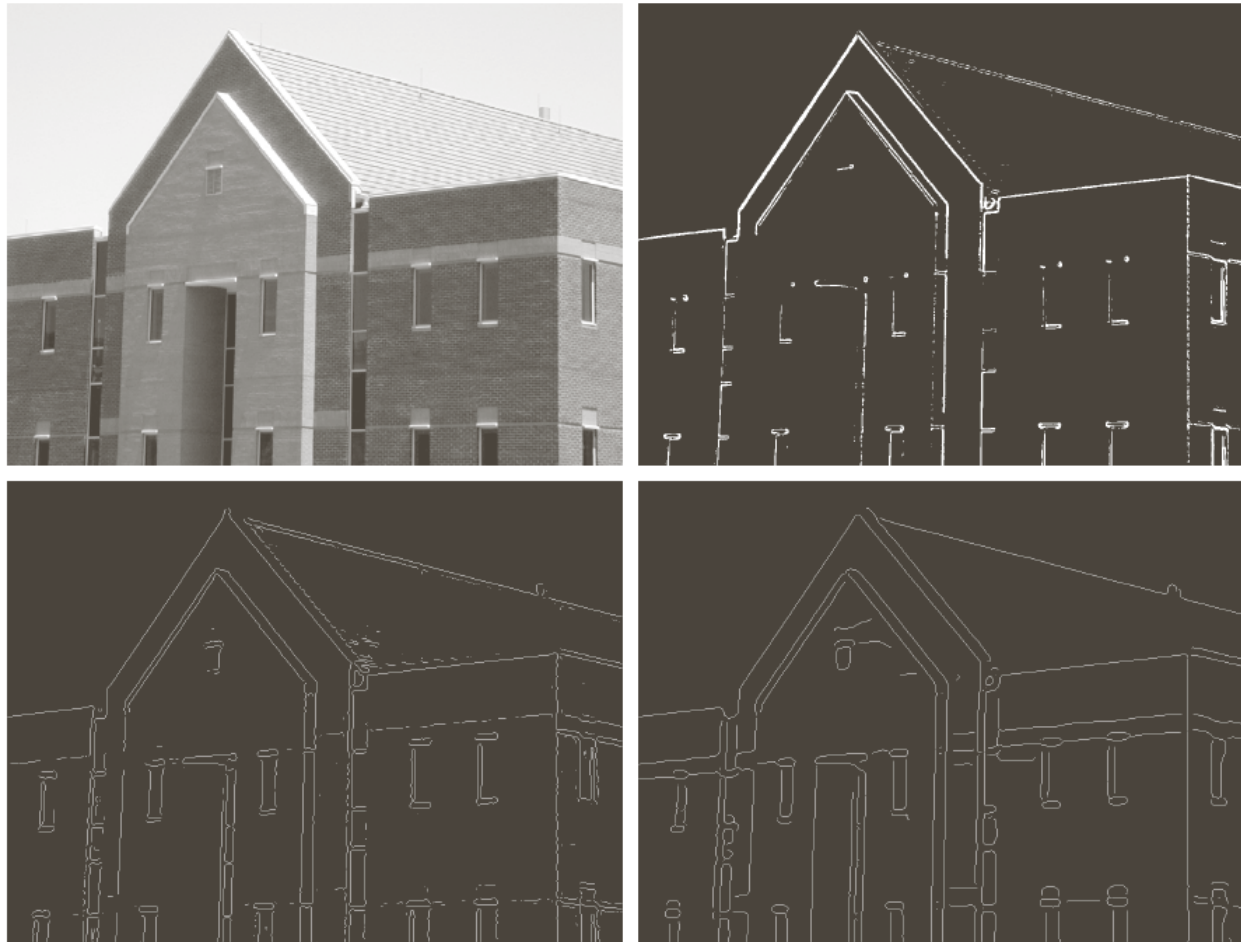
Image Segmentation

36

Courtesy: R.C. Gonzalez and R.E Woods © 1992-2008



# Example: Canny



a	b
c	d

**FIGURE 10.25**

(a) Original image of size  $834 \times 1114$  pixels, with intensity values scaled to the range  $[0, 1]$ .

(b) Thresholded gradient of smoothed image.

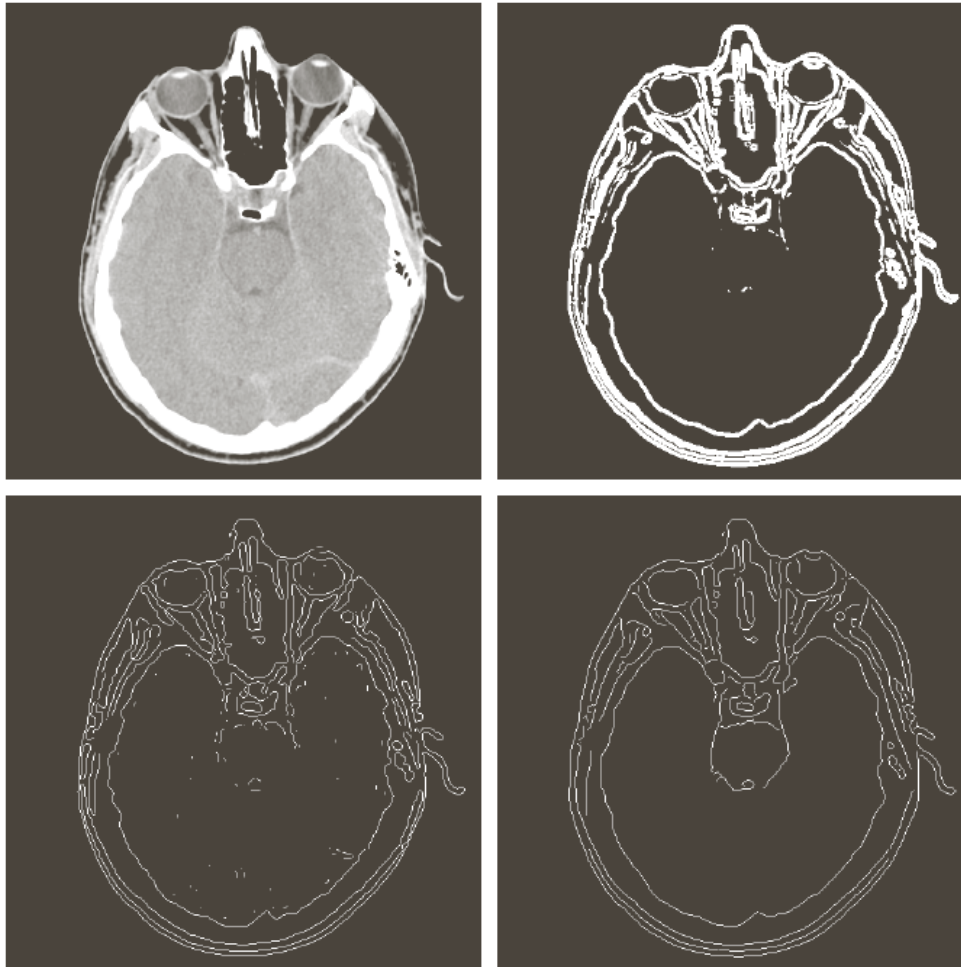
(c) Image obtained using the Marr-Hildreth algorithm.

(d) Image obtained using the Canny algorithm.

Note the significant improvement of the Canny image compared to the other two.



# Example: Canny



a	b
c	d

**FIGURE 10.26**

(a) Original head CT image of size  $512 \times 512$  pixels, with intensity values scaled to the range  $[0, 1]$ .

(b) Thresholded gradient of smoothed image.

(c) Image obtained using the Marr-Hildreth algorithm.

(d) Image obtained using the Canny algorithm.

(Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

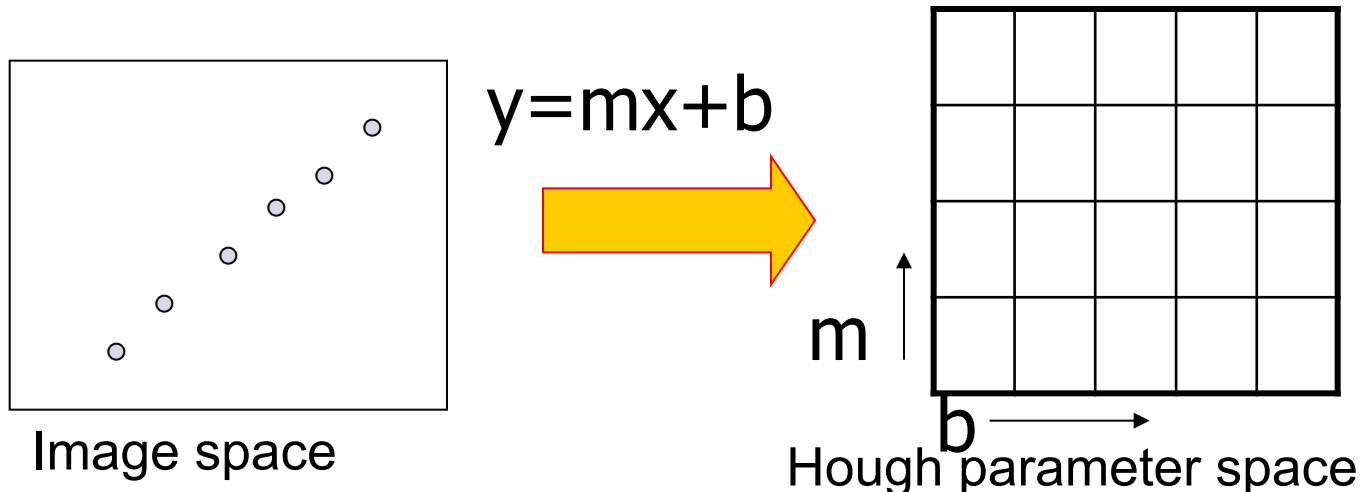
# Edge linking

- Local linking
  - Link an edge pixel with another pixel in its neighborhood having similar magnitude and direction.
    - Computationally expensive
  - Another technique
    - Set an edge pixel of specific direction with a tolerance and having sufficient magnitude to 1, else to 0.
    - Fill small gaps along that direction and compute connected components.
- Global linking
  - Hough Transform
    - Accumulate votes of possible sets of parameters for lines or curves passing through the edge pixel.
    - Local peaks provide the geometric lines or curves linking the edge points.



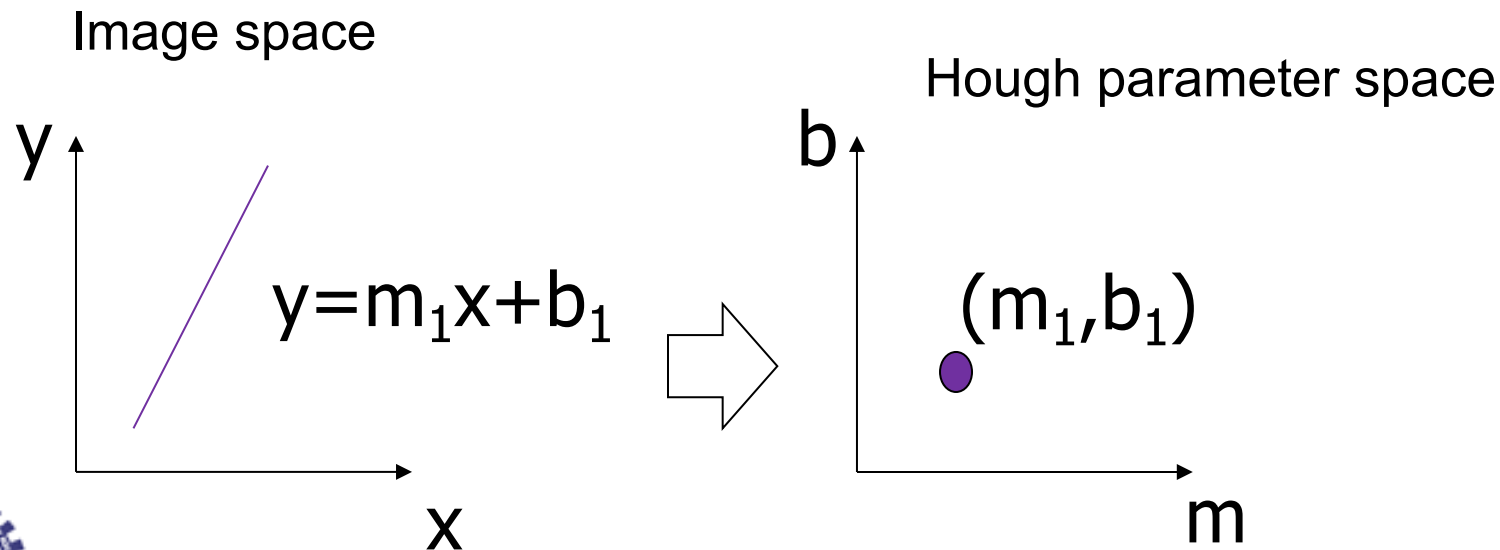
# Hough transform

- ❑ Discretize parameter space into bins
- ❑ For each feature point in the image, put a vote in every bin in the parameter space that could have generated this point.
- ❑ Find bins that have the most votes.



# Parameter space representation

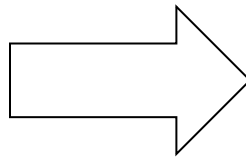
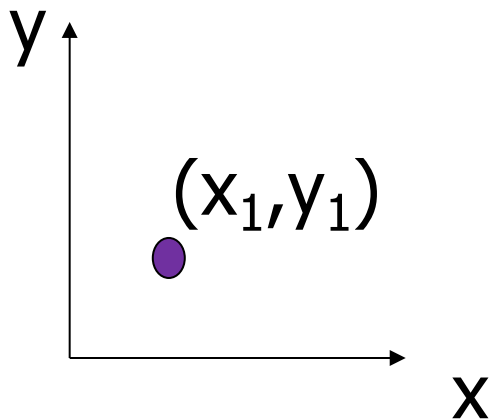
- A straight line in the image corresponds to a point in Hough space.



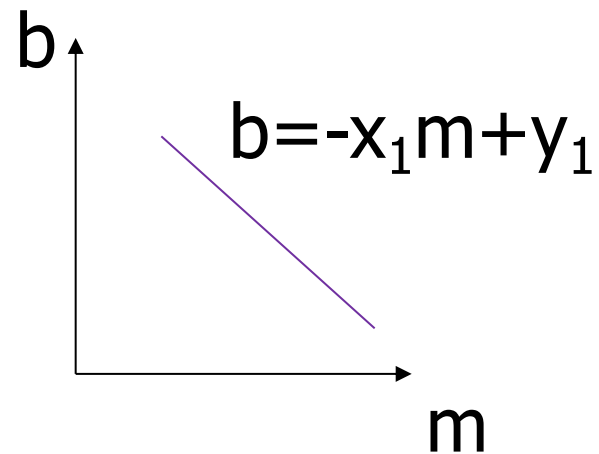
# Parameter space representation

- A point in the image corresponds to a line in the Hough space.

Image space

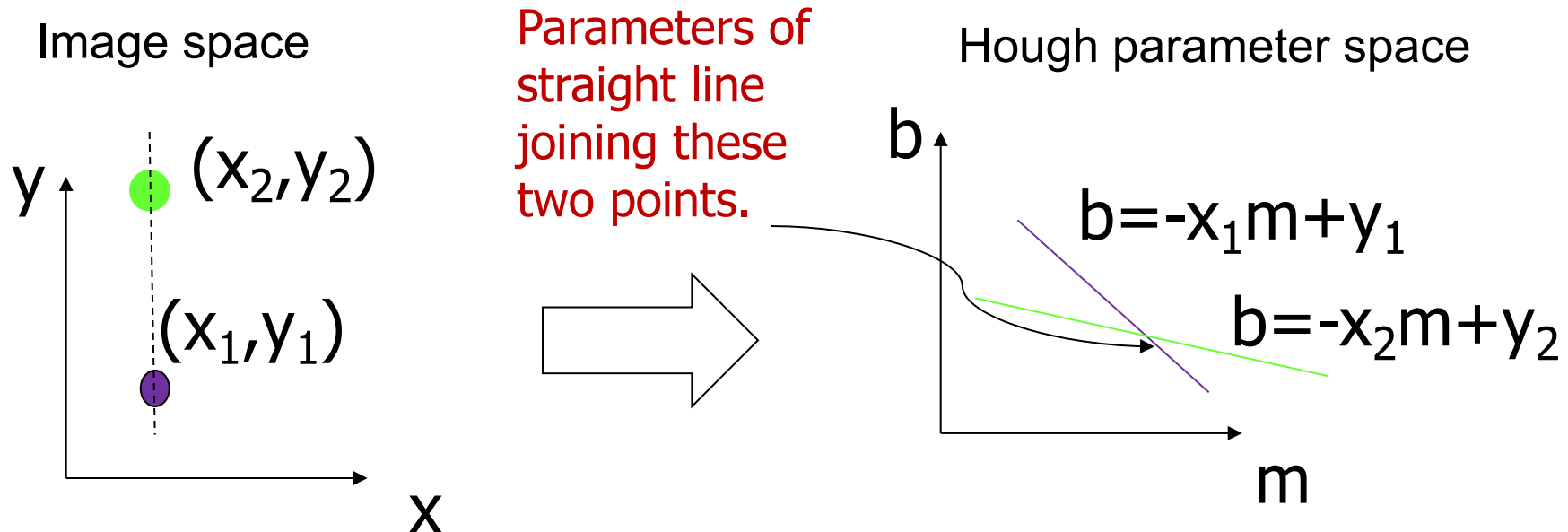


Hough parameter space



# Parameter space representation

- Two points in the image correspond to two lines in the Hough space.

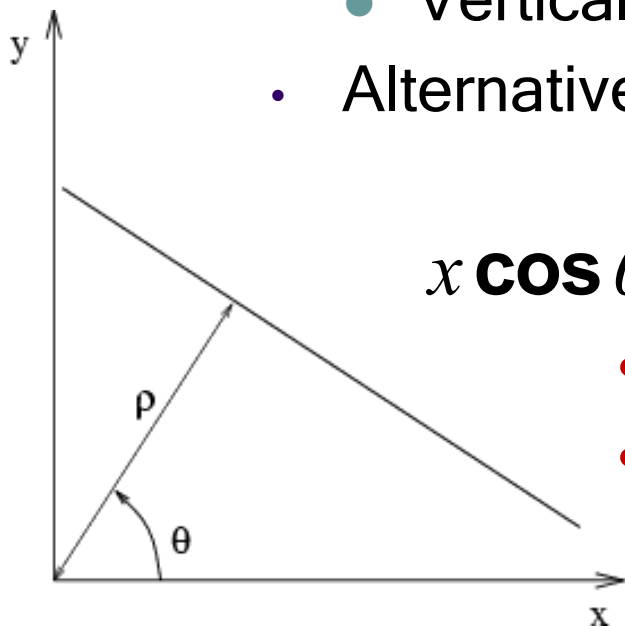


Adapted from slides by S. Seitz



# Parameter space representation

- Problems with the (m,b) space:
  - Unbounded parameter domain
  - Vertical lines require infinite m
- Alternative: polar representation



$$x \cos \theta + y \sin \theta = \rho$$

- $\theta$  varies from 0 to 180°
- $\rho$  varies from 0 to the length of diagonal of the image grid.

Each point will add a sinusoid in the  $(\theta, \rho)$  parameter space

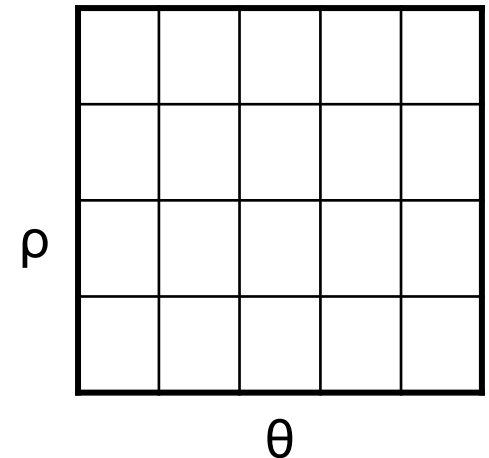




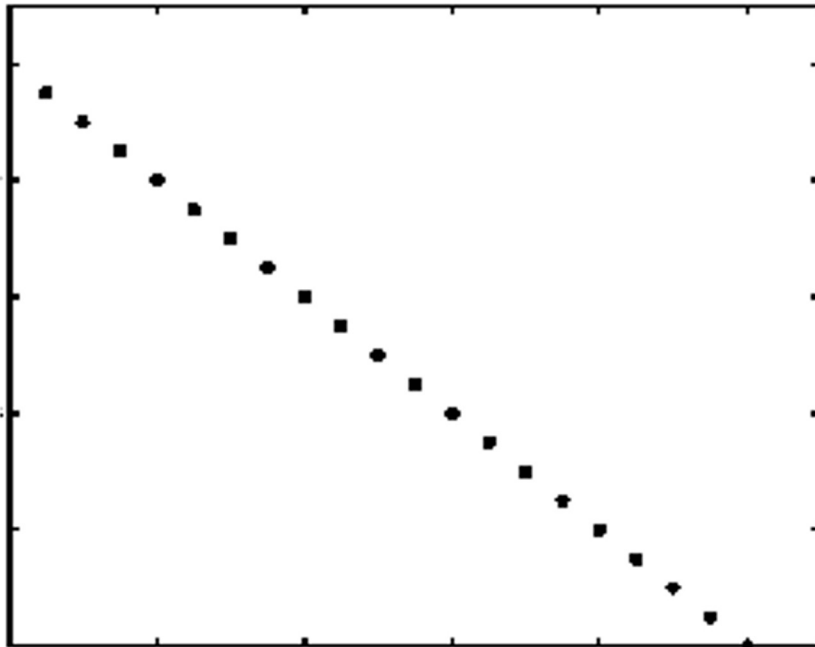
# Algorithm

- Initialize accumulator  $A$  to all zeros
- For each edge point  $(x,y)$  in the image
- {  
    For  $\theta = 0$  to  $180$
- {  
     $\rho = x \cos \theta + y \sin \theta$   
     $A(\theta, \rho) = A(\theta, \rho) + 1$   
    }
- }
- Find the value(s) of  $(\theta, \rho)$  where  $A(\theta, \rho)$  is a local maximum
- The detected line in the image is given by  
     $\rho = x \cos \theta + y \sin \theta$

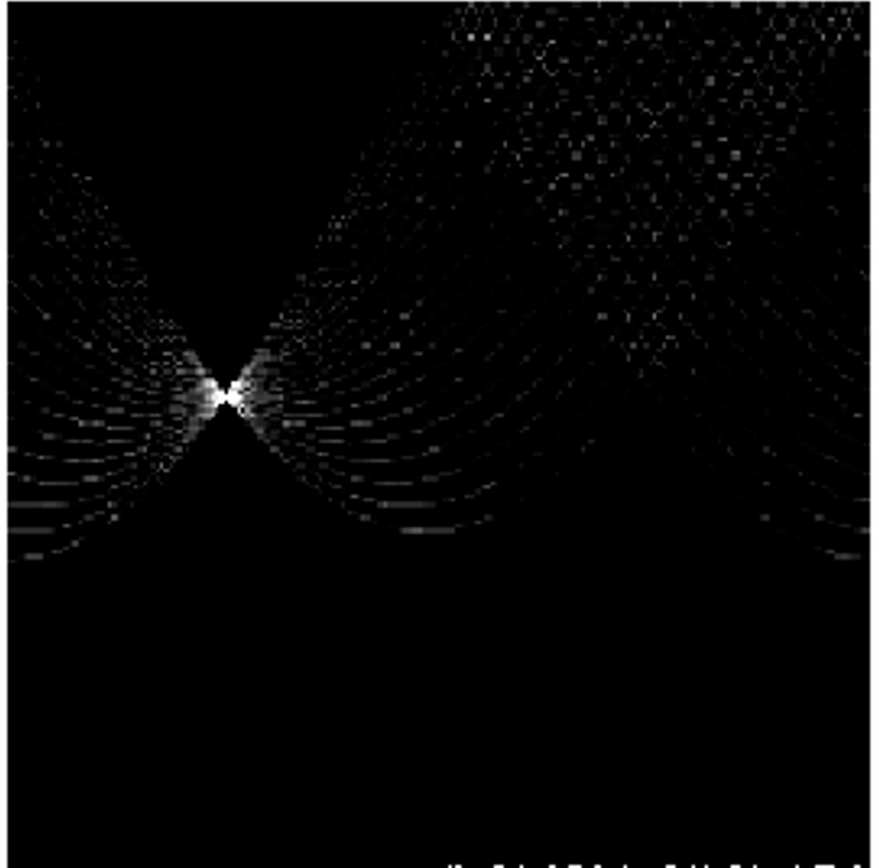
A: Accumulator array



# Basic illustration



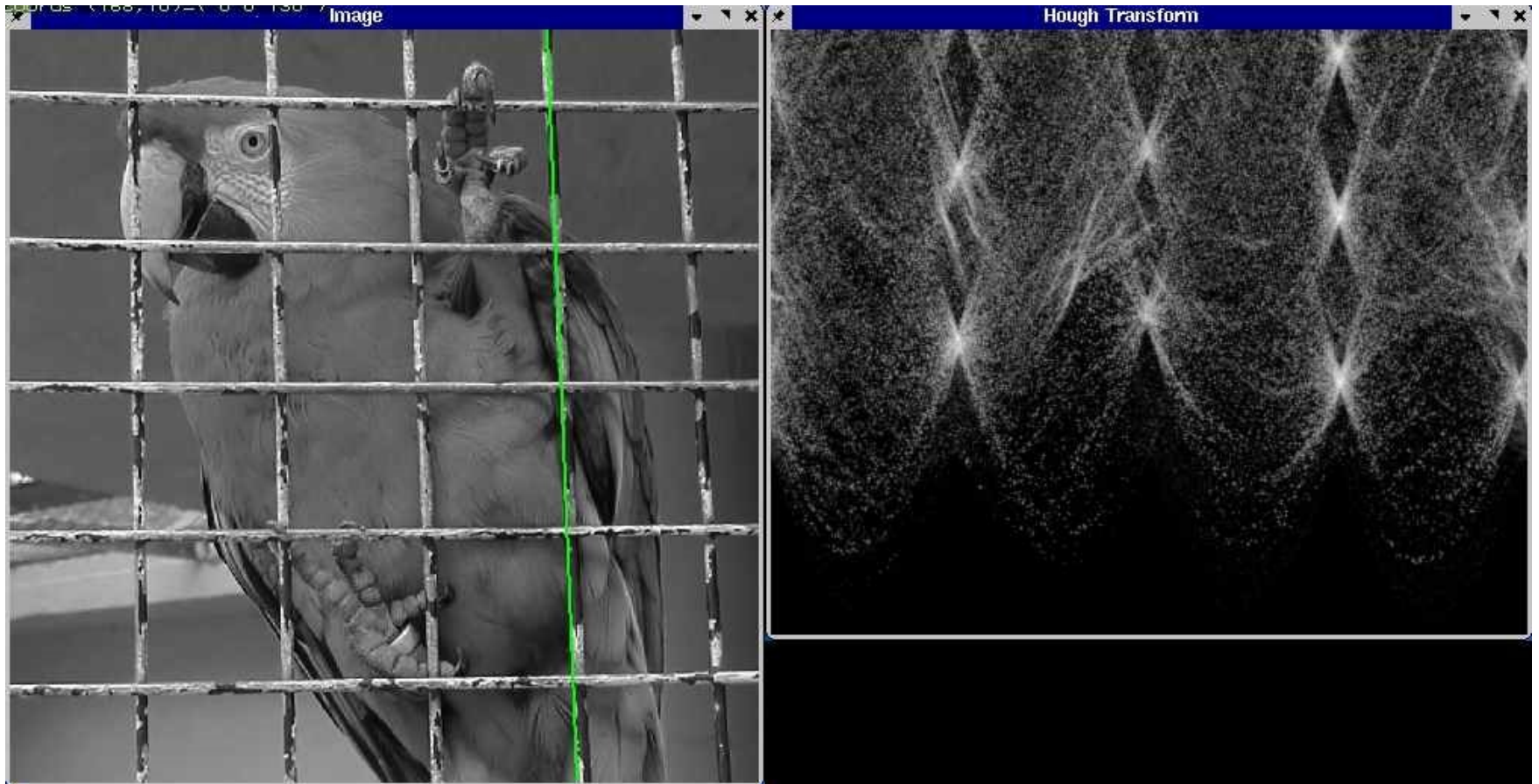
features



votes



# A more complicated image



# Binarization (Thresholding)

- Separation of foreground from background.
  - Foreground: Dark pixels (text)
  - Background: Bright pixels (white region)
- Pixels in a binarized image set to one of the two values (e.g. 255 or 0).
  - 255 for background.
  - 0 for foreground.
- $f: R \text{ (or } Z) \rightarrow \{0, 1\}$  (Or  $\{0, 255\}$ , etc.)
  - A pixel mapping function

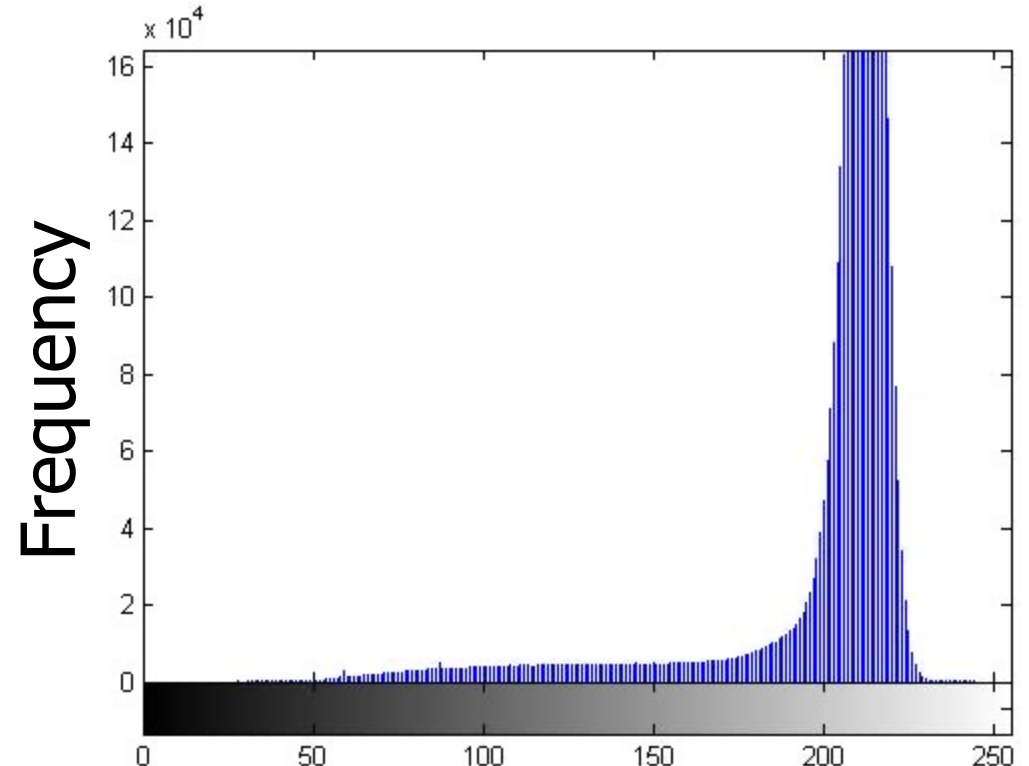
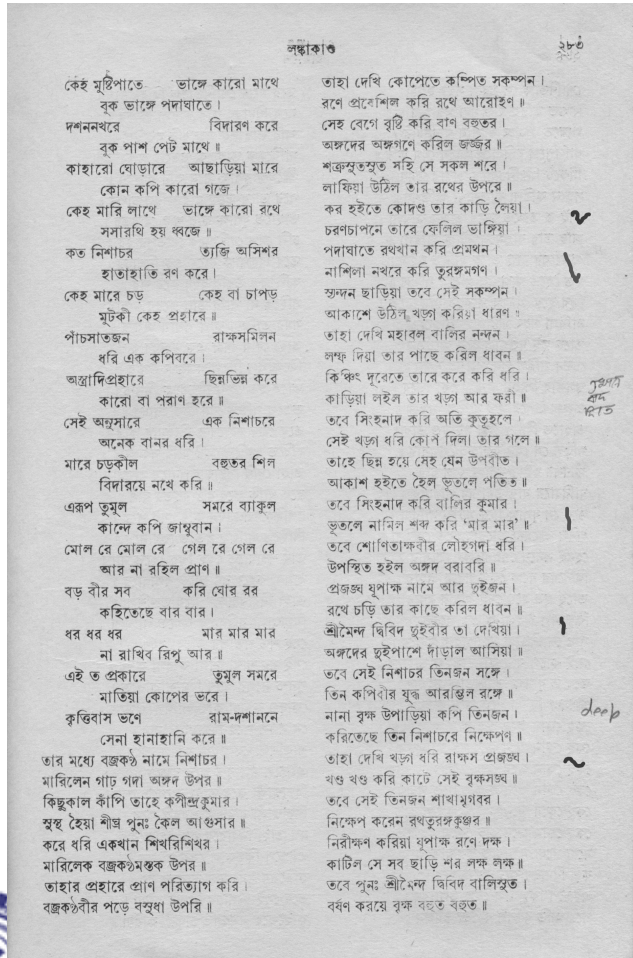


# Binarization (Thresholding)

- A simple algorithm
  - Choose a threshold value  $T$ .
  - A pixel greater than  $T$  is set to 255 (background), otherwise to 0 (foreground).



# Image histogram

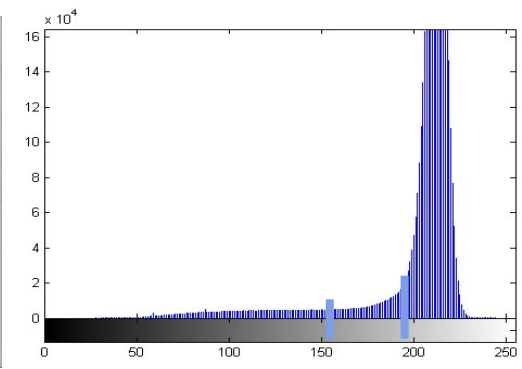
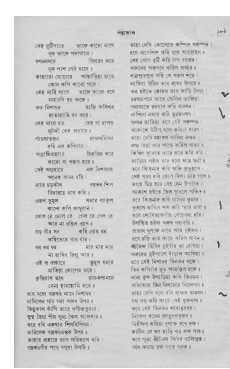


Brightness (x)

Normalized Histogram  $\rightarrow p(x)$



# Thresholding



Can you automate this operation?

লঙ্কাবাত

২৮৬

কেই মুষ্টিপাতে ভাসে কারো মাথে  
বুক ভাসে পদাঘাতে ।  
দর্শনমথরে বিদারণ করে  
বুক পাশ পেট মাথে ॥  
কাহারো বোড়ারে আছাড়িয়া মারে  
কোন কপি কারো গাঙ্গে ।  
কেহ মারি লাগে ভাসে কারো রথে  
সমারথি হয় ধ্বজে ॥  
কত নিশাচর তাজ্জি অসিশর  
হাতাহাতি রণ করে ।  
কেহ মারে চড় কেহ বা চাপড়  
মুটকী কেহ প্রহারে ॥  
পাঁচাত্তজন রাফসমিলন  
ধরি এক কপিবরে ।  
অগ্ন্যগ্নিগ্রহারে ছিন্নভিন্ন করে  
কারো বা পরাণ হরে ॥  
সেই অম্বসারে এক নিশাচরে  
অনেক বানর ধরি ।  
মারে চড়কাল বহুতর শিল  
বিদারয়ে নখে করি ॥  
একপ তুমুল সমরে ব্যাকুল  
কালে কপি জাম্বান ।  
মোল রে মোল রে গেল রে গেল রে  
আর না রহিল প্রাণ ॥  
বড় বীর সব করি যোঁর রর  
কহিতেছে বার বার ।  
ধর ধর ধর মার মার মার  
না রাখিব রিপু আর ॥  
এই ত প্রকারে তুমুল সমরে  
মতিয়া কোপের ভরে ।  
কৃতিবাস ভণে রাম-দশনানে  
সেনা হানাহানি করে ॥

তার মধ্যে বজ্রকণ্ট নামে নিশাচর ।  
মারিলেন গাঢ় গালা অঙ্গন উপর ।  
কিছুকাল কাঁপি তাহে কপীন্দ্রকুমার ।  
সুস্থ হৈয়া শীঘ্র পুনঃ কৈল আগ্রসার ॥  
করে ধরি একখান শিখরিশিখর ।  
মারিলেক বজ্রকণ্টমস্তক উপর ।  
তাহার প্রহারে প্রাণ পরিত্যাগ করি ।  
বজ্রকণ্টবীর পড়ে লম্বা উপরি ॥

তাহা দেবি কোপেতে কণ্ঠিত সৰুস্পন্দন ।  
রণে প্রবেশিল করি রথে আরোহণ ॥  
সেহ বেগে বৃষ্টি করি বাণ বহুতর ।  
অঙ্গদের অঙ্গগণে করিল ছর্জ্জর ॥  
শত্রুস্বত্বহৃত সহি সে সকল শরে ।  
লাফিয়া উঠিল তার রথের উপরে ॥  
কর হইতে কোদণ্ড তার কাড়ি লৈয়া ।  
চরণচাপনে তারে ছেলিল ডাকিয়া ॥  
পদাঘাতে রথখান করি প্রমথন ।  
নাশিলা নথরে করি তুরঙ্গমগণ ॥  
স্বন্দন ছাড়িয়া তারে সেই সৰুস্পন্দন ।  
আকাশে উঠিল খড়্গা করিয়া ধারণ ॥  
তাহা দেখি মহাবল বাণির নন্দন ।  
লক্ষ দিয়া তার পাছে করিল ধাবন ॥  
কিঞ্চিৎ দূরেতে তারে করে করি ধরি ।  
কাড়িয়া লইল তার খড়্গা আর ফরি ॥  
তবে সিংহনাদ করি অতি ক্রুদ্ধহলে ।  
সেই খড়্গা ধরি কোণ দিলা তার গলে ॥  
তাহে ছিন্ন হয়ে সেই বেন উপবীত ।  
আকাশ হইতে হৈল ভূতলে পতিত ॥  
তবে সিংহনাদ করি বাণির কুমার ।  
ভূতলে নামিল শব্দ করি 'মার মার' ॥  
তবে শোণিতাকবীর লৌহগদা ধরি ।  
উপস্থিত হইল অঙ্গদ বরাবরি ॥  
প্রজন্ম যুগ্মক নামে আর দুইজন ।  
রথে চড়ি তার কাছে করিল ধাবন ॥  
ঐন্দ্রমৈদ্র দ্বিবিদ দুইবার তা দেখিয়া ।  
অঙ্গদের দুইপাশে পাড়াল আসিয়া ॥  
তবে সেই নিশাচর ভিন্নজন সঙ্গে ।  
তিন কপিবীর যুদ্ধ আরম্ভিল সঙ্গে ॥  
নানা বৃক্ষ উপাডিয়া কপি ভিন্জন ।  
করিতেছে তিন নিশাচরে নিক্ষেপণ ॥  
তাহা দেখি খড়্গা ধরি রাফস প্রজন্ম ।  
যগু যগু করি কাটে সেই বৃক্ষসজ্জ ॥  
তবে সেই ভিন্নজন শাখাযগবর ।  
নিক্ষেপ করেন রথতুরঙ্গকুঞ্জর ॥  
নিরীক্ষণ করিয়া যুগ্মক রণে দক্ষ ।  
কাটিল সে সব ছাড়ি শর লক্ষ লক্ষ ॥  
তবে পুনঃ ঐন্দ্রমৈদ্র দ্বিবিদ বাসিহুত ।  
বর্ধণ করয়ে বৃক্ষ বহুত বহুত ॥

লঙ্কাবাত

২৮৬

কেই মুষ্টিপাতে ভাসে কারো মাথে  
বুক ভাসে পদাঘাতে ।  
দর্শনমথরে বিদারণ করে  
বুক পাশ পেট মাথে ॥  
কাহারো বোড়ারে আছাড়িয়া মারে  
কোন কপি কারো গাঙ্গে ॥  
কেহ মারি লাগে ভাসে কারো রথে  
সমারথি হয় ধ্বজে ॥  
কত নিশাচর তাজ্জি অসিশর  
হাতাহাতি রণ করে ।  
কেহ মারে চড় কেহ বা চাপড়  
মুটকী কেহ প্রহারে ॥  
পাঁচাত্তজন রাফসমিলন  
ধরি এক কপিবরে ।  
অগ্ন্যগ্নিগ্রহারে ছিন্নভিন্ন করে  
কারো বা পরাণ হরে ॥  
সেই অম্বসারে এক নিশাচরে  
অনেক বানর ধরি ।  
মারে চড়কাল বহুতর শিল  
বিদারয়ে নখে করি ॥  
একপ তুমুল সমরে ব্যাকুল  
কালে কপি জাম্বান ।  
মোল রে মোল রে গেল রে গেল রে  
আর না রহিল প্রাণ ॥  
বড় বীর সব করি যোঁর রর  
কহিতেছে বার বার ।  
ধর ধর ধর মার মার মার  
না রাখিব রিপু আর ॥  
এই ত প্রকারে তুমুল সমরে  
মতিয়া কোপের ভরে ।  
কৃতিবাস ভণে রাম-বশনানে  
সেনা হানাহানি করে ॥

তার মধ্যে বজ্রকণ্ট নামে নিশাচর ।  
মারিলেন গাঢ় গালা অঙ্গন উপর ।  
কিছুকাল কাঁপি তাহে কপীন্দ্রকুমার ।  
সুস্থ হৈয়া শীঘ্র পুনঃ কৈল আগ্রসার ॥  
করে ধরি একখান শিখরিশিখর ।  
মারিলেক বজ্রকণ্টমস্তক উপর ।  
তাহার প্রহারে প্রাণ পরিত্যাগ করি ।  
বজ্রকণ্টবীর পড়ে লম্বা উপরি ॥

তাহা দেবি কোপেতে কণ্ঠিত সৰুস্পন্দন ।  
রণে প্রবেশিল করি রথে আরোহণ ॥  
সেহ বেগে বৃষ্টি করি বাণ বহুতর ।  
অঙ্গদের অঙ্গগণে করিল ছর্জ্জর ॥  
শত্রুস্বত্বহৃত সহি সে সকল শরে ।  
লাফিয়া উঠিল তার রথের উপরে ॥  
কর হইতে কোদণ্ড তার কাড়ি লৈয়া ।  
চরণচাপনে তারে ছেলিল ডাকিয়া ॥  
পদাঘাতে রথখান করি প্রমথন ।  
নাশিলা নথরে করি তুরঙ্গমগণ ॥  
স্বন্দন ছাড়িয়া তারে সেই সৰুস্পন্দন ।  
আকাশে উঠিল খড়্গা করিয়া ধারণ ॥  
তাহা দেখি মহাবল বাণির নন্দন ।  
লক্ষ দিয়া তার পাছে করিল ধাবন ॥  
কিঞ্চিৎ দূরেতে তারে করে করি ধরি ।  
কাড়িয়া লইল তার খড়্গা আর ফরি ॥  
তবে সিংহনাদ করি অতি ক্রুদ্ধহলে ।  
সেই খড়্গা ধরি কোণ দিলা তার গলে ॥  
তাহে ছিন্ন হয়ে সেই বেন উপবীত ।  
আকাশ হইতে হৈল ভূতলে পতিত ॥  
তবে সিংহনাদ করি বাণির কুমার ।  
ভূতলে নামিল শব্দ করি 'মার মার' ॥  
তবে শোণিতাকবীর লৌহগদা ধরি ।  
উপস্থিত হইল অঙ্গদ বরাবরি ॥  
প্রজন্ম যুগ্মক নামে আর দুইজন ।  
রথে চড়ি তার কাছে করিল ধাবন ॥  
ঐন্দ্রমৈদ্র দ্বিবিদ দুইবার তা দেখিয়া ।  
অঙ্গদের দুইপাশে পাড়াল আসিয়া ॥  
তবে সেই নিশাচর ভিন্নজন সঙ্গে ।  
তিন কপিবীর যুদ্ধ আরম্ভিল সঙ্গে ॥  
নানা বৃক্ষ উপাডিয়া কপি ভিন্জন ।  
করিতেছে তিন নিশাচরে নিক্ষেপণ ॥  
তাহা দেখি খড়্গা ধরি রাফস প্রজন্ম ।  
যগু যগু করি কাটে সেই বৃক্ষসজ্জ ॥  
তবে সেই ভিন্নজন শাখাযগবর ।  
নিক্ষেপ করেন রথতুরঙ্গকুঞ্জর ॥  
নিরীক্ষণ করিয়া যুগ্মক রণে দক্ষ ।  
কাটিল সে সব ছাড়ি শর লক্ষ লক্ষ ॥  
তবে পুনঃ ঐন্দ্রমৈদ্র দ্বিবিদ বাসিহুত ।  
বর্ধণ করয়ে বৃক্ষ বহুত বহুত ॥

Th=156

Th=192



# A simple iterative algorithm

- Select an initial estimate of global threshold  $T$ .
- Partition into two sets of background ( $x \geq T$ ) and foreground ( $x < T$ ).
- Compute means of background and foreground  $m_1$  and  $m_2$ , respectively.
- Obtain new estimate of  $T$ ,  $T = (m_1 + m_2) / 2$ .
- Iterate above steps till convergence



# Bayesian Classification of foreground and background pixels

Pixels belonging to two classes:

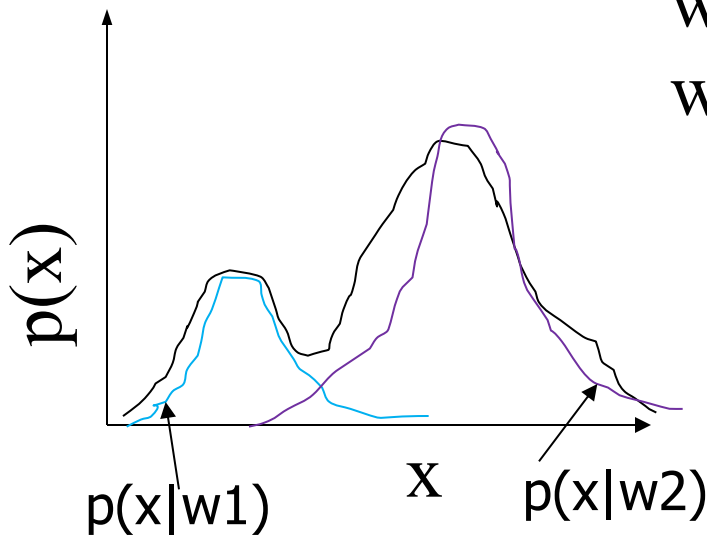
$w_1$ : Foreground

$w_2$ : Background

Compute  $p(w_1|x)$  and  $p(w_2|x)$ .

Bayes' theorem:

$$p(\omega | x) = \frac{p(\omega) p(x | \omega)}{p(x)}$$



Bayes' classification rule:

Assign  $x$  to  $w_1$  if  $p(w_1|x) > p(w_2|x)$ , else to  $w_2$ .



To check whether  $p(w_1)p(x|w_1) > p(w_2)p(x|w_2)$

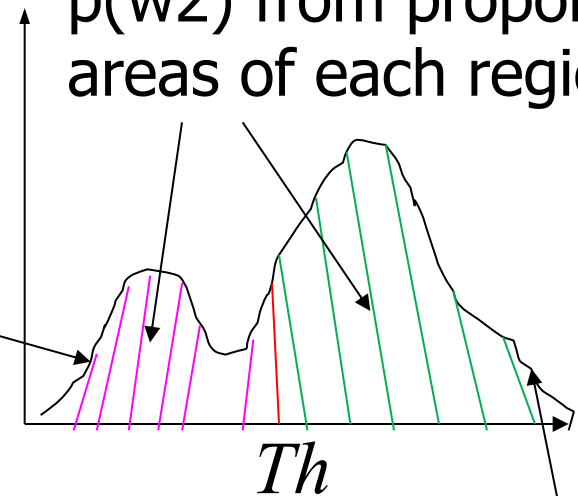


# Expectation-Maximization Algorithm

2. Compute parameters of  $p(x|w_1)$  by assuming it Gaussian.

$\mu_1, \sigma_1$

1. Compute  $p(w_1)$  and  $p(w_2)$  from proportional areas of each region.

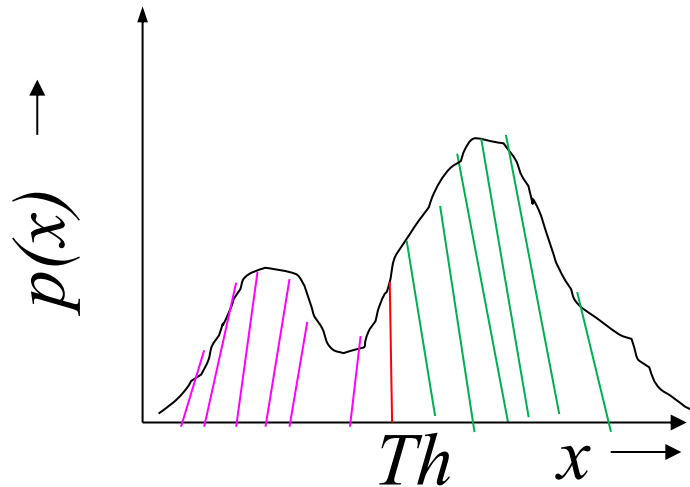


3. Compute parameters of  $p(x|w_2)$  by assuming it Gaussian.  $\mu_2, \sigma_2$

$$N(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



# Expectation-Maximization Algorithm



4. Compute new threshold value so that for  $x < Th$ ,  $p(w1|x) > p(w2|x)$ , and vice versa.

5. Iterate till the value of  $Th$  converges.

$$p(w1) = \sum_{x=0}^{Th} p(x)$$

$$p(w2) = 1 - p(w1)$$

$$\mu_1 = \sum_{x=0}^{Th} x \cdot p(x)$$

$$\sigma_1^2 = \sum_{x=0}^{Th} x^2 \cdot p(x) - \mu_1^2$$

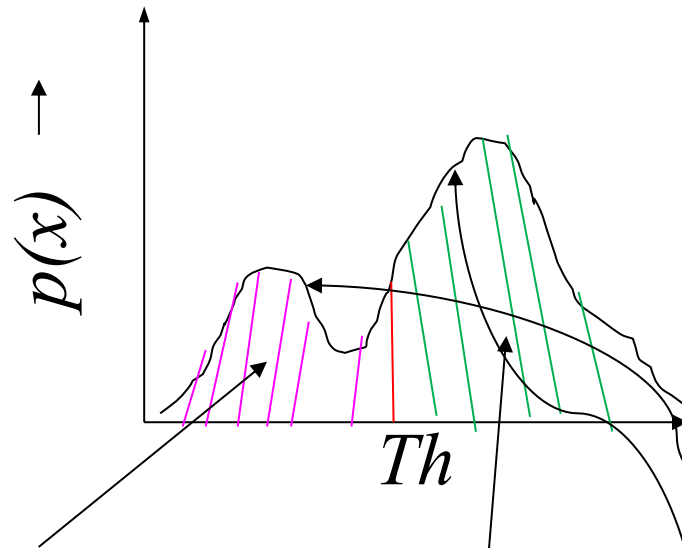
$$\mu_2 = \sum_{x=Th+1}^{255} x \cdot p(x)$$

$$\sigma_2^2 = \sum_{x=Th+1}^{255} x^2 \cdot p(x) - \mu_2^2$$



# Otsu Thresholding

- Choose a threshold value, which maximizes between class variance ( $\sigma_B^2$ ).



$$\sigma_B^2 = p(w1)p(w2)(\mu_2 - \mu_1)^2$$



# An example

Otsu:157

কেই মুষ্টিপাতে ভাঙ্গে কারো মাথে  
বক ভাঙ্গে পদাঘাতে ।  
দশননখরে বিদারণ করে  
বক পাশ পেট মাথে ॥  
কাহারো ঘোড়ারে আছাড়িয়া মারে  
কোন কপি কারো গজে ।  
কেহ মারি লাখে ভাঙ্গে কারো রথে  
সমারথি হয় ধ্বজে ॥  
কত নিশাচর তাজি অসিশর  
হাতাহাতি রণ করে ।  
কেহ মারে চড় কেহ বা চাপড়  
মুটকী কেহ প্রহারে ॥  
পাঁচসাতজন রাফসমিলন  
ধরি এক কপিবারে ।  
অস্ত্রাদিগ্রহায়ে ছিন্নভিন্ন করে  
কারো বা পরাণ হরে ॥  
সেই অমুসারে এক নিশাচরে  
অনেক বানর ধরি ।  
মারে চড়কীল বজ্রতর শিল  
বিদারয়ে নখে করি ॥  
এরূপ তুমুল সমরে ব্যাকুল  
কান্দে কপি জাখুবান ।  
মোল রে মোল রে গেল রে গেল রে  
আর না রহিল প্রাণ ॥  
বড় বীর সব করি ঘোর রব  
কহিতেছে বার বার ।  
ধর ধর ধর মার মার মার  
না রাখিব রিপু আর ॥  
এই ত প্রকারে তুমুল সমরে  
মাতিয়া কোপের ভরে ।  
কৃতিবাস ভণে রাম-দশনানে  
সেনা হানাহানি করে ॥  
তার মধ্যে বজ্রকণ্ট নামে নিশাচর ।  
মারিলেন গাট গালা অঙ্গ উপর ॥  
কিছুকাল কাঁপি তাহে কপীন্দ্রকুমার ।  
স্বস্থ হৈয়া শীঘ্র পুনঃ কৈল আগুসার ॥  
করে ধরি একখান শিখরিশিখর ।  
মারিলেক বজ্রকণ্টমস্তক উপর ॥  
তাহার প্রহারে প্রাণ পরিত্যাগ করি ।  
বজ্রকণ্টবীর পড়ে লম্বা উপরি ॥

লঙ্কা৩

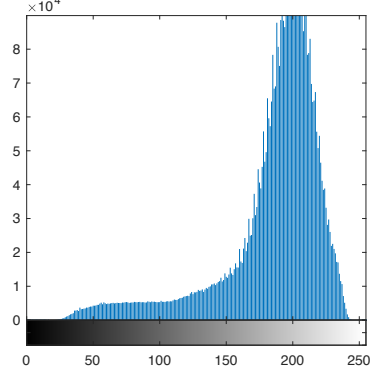
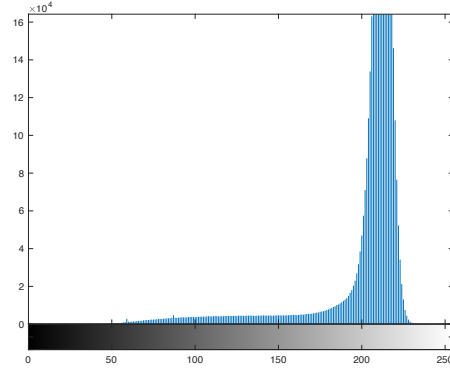
২৮৬

তাহা দেখি কোপেতে কল্লিত সঙ্কম্পন ।  
রণে প্রবেশিল করি রথে আরোহণ ॥  
সেহ বেগে বৃষ্টি করি বাণ বজ্রতর ।  
অঙ্গদের অঙ্গগণে করিল জর্জর ॥  
শক্রমৃত্যুত সহি সে সকল শরে ।  
লাফিয়া উঠিল তার রথের উপরে ॥  
কর হইতে কোদণ্ড তার কাড়ি লৈয়া ।  
চরণচাপনে তারে ফেলিল ভাঙ্গিয়া ॥  
পদাঘাতে রথখান করি প্রমথন ।  
নাশিলা নখরে করি তুরঙ্গমগণ ॥  
স্বন্দন ছাড়িয়া তবে সেই সঙ্কম্পন ।  
আকাশে উঠিল খজা করিধা ধারণ ॥  
তাহা দেখি মহাবল বালির নন্দন ।  
লক্ষ দিয়া তার পাছে করিল ধাবন ॥  
কিঞ্চিৎ দূরেতে তারে করে করি ধরি ।  
কাড়িয়া লইল তার খজা আর ফরি ॥  
তবে সিংহনাদ করি অতি কুতূহলে  
সেই খজা ধরি কোণ দিলা তার গলে ॥  
তাহে ছিন্ন হয়ে সেহ যেন উপবীত ।  
আকাশ হইতে হৈল ভূতলে পতিত ॥  
তবে সিংহনাদ করি বালির কুমার ।  
ভূতলে নামিল শব্দ করি 'মার মার' ॥  
তবে শোণিতাকবীর সৌহগদা ধরি ।  
উপস্থিত হইল অঙ্গদ বরাবরি ॥  
প্রজ্ঞব যুগাক নামে আর দুইজন ।  
রথে চড়ি তার কাছে করিল ধাবন ॥  
শ্রীমৈন্দ দ্বিবিদ দুইবার তা দেখিয়া ।  
অঙ্গদের দুইপাশে দাঁড়াল আসিয়া ॥  
তবে সেই নিশাচর তিনজন সঙ্গে ।  
তিন কপিবার যুদ্ধ আরম্ভিল রঙ্গে ॥  
নানা বৃক্ষ উপাড়িয়া কপি তিনজন ।  
করিতেছে তিন নিশাচরে নিক্ষেপণ ॥  
তাহা দেখি খজা ধরি রাফস প্রজ্ঞব ।  
বগু বগু করি কাটে সেই বৃক্ষসম্ম ॥  
তবে সেই তিনজন শাখামুগবর ।  
নিক্ষেপ করেন রথতুরঙ্গর ॥  
নিরাঙ্কণ করিয়া খুগাক রণে দক্ষ ।  
কাটিল সে সব ছাড়ি শর লক্ষ লক্ষ ॥  
তবে পুনঃ শ্রীমৈন্দ দ্বিবিদ বালিব্রত ।  
বধণ করয়ে বৃক্ষ বজ্রত বজ্রত ॥

১৮৫  
১৮৫

১৮৫

Bayesia  
n:157



লঙ্কা৩

২৮৬

কেই মুষ্টিপাতে ভাঙ্গে কারো মাথে  
বক ভাঙ্গে পদাঘাতে ।  
দশননখরে বিদারণ করে  
বক পাশ পেট মাথে ॥  
কাহারো ঘোড়ারে আছাড়িয়া মারে  
কোন কপি কারো গজে ।  
কেহ মারি লাখে ভাঙ্গে কারো রথে  
সমারথি হয় ধ্বজে ॥  
কত নিশাচর তাজি অসিশর  
হাতাহাতি রণ করে ।  
কেহ মারে চড় কেহ বা চাপড়  
মুটকী কেহ প্রহারে ॥  
পাঁচসাতজন রাফসমিলন  
ধরি এক কপিবারে ।  
অস্ত্রাদিগ্রহায়ে ছিন্নভিন্ন করে  
কারো বা পরাণ হরে ॥  
সেই অমুসারে এক নিশাচরে  
অনেক বানর ধরি ।  
মারে চড়কীল বজ্রতর শিল  
বিদারয়ে নখে করি ॥  
এরূপ তুমুল সমরে ব্যাকুল  
কান্দে কপি জাখুবান ।  
মোল রে মোল রে গেল রে গেল রে  
আর না রহিল প্রাণ ॥  
বড় বীর সব করি ঘোর রব  
কহিতেছে বার বার ।  
ধর ধর ধর মার মার মার  
না রাখিব রিপু আর ॥  
এই ত প্রকারে তুমুল সমরে  
মাতিয়া কোপের ভরে ।  
কৃতিবাস ভণে রাম-দশনানে  
সেনা হানাহানি করে ॥  
তার মধ্যে বজ্রকণ্ট নামে নিশাচর ।  
মারিলেন গাট গালা অঙ্গ উপর ॥  
কিছুকাল কাঁপি তাহে কপীন্দ্রকুমার ।  
স্বস্থ হৈয়া শীঘ্র পুনঃ কৈল আগুসার ॥  
করে ধরি একখান শিখরিশিখর ।  
মারিলেক বজ্রকণ্টমস্তক উপর ॥  
তাহার প্রহারে প্রাণ পরিত্যাগ করি ।  
বজ্রকণ্টবীর পড়ে লম্বা উপরি ॥

তাহা দেখি কোপেতে কল্লিত সঙ্কম্পন ।  
রণে প্রবেশিল করি রথে আরোহণ ॥  
সেহ বেগে বৃষ্টি করি বাণ বজ্রতর ।  
অঙ্গদের অঙ্গগণে করিল জর্জর ॥  
শক্রমৃত্যুত সহি সে সকল শরে ।  
লাফিয়া উঠিল তার রথের উপরে ॥  
কর হইতে কোদণ্ড তার কাড়ি লৈয়া ।  
চরণচাপনে তারে ফেলিল ভাঙ্গিয়া ॥  
পদাঘাতে রথখান করি প্রমথন ।  
নাশিলা নখরে করি তুরঙ্গমগণ ॥  
স্বন্দন ছাড়িয়া তবে সেই সঙ্কম্পন ।  
আকাশে উঠিল খজা করিধা ধারণ ॥  
তাহা দেখি মহাবল বালির নন্দন ।  
লক্ষ দিয়া তার পাছে করিল ধাবন ॥  
কিঞ্চিৎ দূরেতে তারে করে করি ধরি ।  
কাড়িয়া লইল তার খজা আর ফরি ॥  
তবে সিংহনাদ করি অতি কুতূহলে  
সেই খজা ধরি কোণ দিলা তার গলে ॥  
তাহে ছিন্ন হয়ে সেহ যেন উপবীত ।  
আকাশ হইতে হৈল ভূতলে পতিত ॥  
তবে সিংহনাদ করি বালির কুমার ।  
ভূতলে নামিল শব্দ করি 'মার মার' ॥  
তবে শোণিতাকবীর সৌহগদা ধরি ।  
উপস্থিত হইল অঙ্গদ বরাবরি ॥  
প্রজ্ঞব যুগাক নামে আর দুইজন ।  
রথে চড়ি তার কাছে করিল ধাবন ॥  
শ্রীমৈন্দ দ্বিবিদ দুইবার তা দেখিয়া ।  
অঙ্গদের দুইপাশে দাঁড়াল আসিয়া ॥  
তবে সেই নিশাচর তিনজন সঙ্গে ।  
তিন কপিবার যুদ্ধ আরম্ভিল রঙ্গে ॥  
নানা বৃক্ষ উপাড়িয়া কপি তিনজন ।  
করিতেছে তিন নিশাচরে নিক্ষেপণ ॥  
তাহা দেখি খজা ধরি রাফস প্রজ্ঞব ।  
বগু বগু করি কাটে সেই বৃক্ষসম্ম ॥  
তবে সেই তিনজন শাখামুগবর ।  
নিক্ষেপ করেন রথতুরঙ্গর ॥  
নিরাঙ্কণ করিয়া খুগাক রণে দক্ষ ।  
কাটিল সে সব ছাড়ি শর লক্ষ লক্ষ ॥  
তবে পুনঃ শ্রীমৈন্দ দ্বিবিদ বালিব্রত ।  
বধণ করয়ে বৃক্ষ বজ্রত বজ্রত ॥

১৮৫  
১৮৫

১৮৫



# Probabilistic modeling with mixture of densities

Number of components  $\rightarrow K$

Component density  $\rightarrow P(\mathbf{x}|G_i)$

Component proportion  $\rightarrow P(G_i)$

$$P(\mathbf{x}) = \sum_{i=1}^K P(\mathbf{x}|G_i)P(G_i)$$

- $G_i$  defines the  $i$ th segment or cluster.
- $K$  is a hyper-parameter and should be known.
- For multivariate Gaussian distribution:

- $P(\mathbf{x}|G_i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \Sigma_i)$

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})}$$

- To estimate  $\boldsymbol{\mu}_i$ ,  $\Sigma_i$ , and  $P(G_i)$  for all  $i$ . from the set of *iid.* input samples:  $X=\{\mathbf{x}^t\}$ ,  $t=1,2,\dots,N$





# Mixture of Gaussians: Probabilistic inference

- Technique could be refined by computing probabilities of belongingness to a segments.

Parametric PDF:

$$p(x|\{\pi_k, \mu_k, \Sigma_k\}) = \sum_k \pi_k N(x|\mu_k, \Sigma_k)$$

Mixing coefficients

Mean vector

Covariance matrix



# Expectation Maximization (EM) Algorithm

$$z_{ik} = \frac{1}{Z_i} \pi_k N(x_i | \mu_k, \Sigma_k)$$

Normalizing  
factor

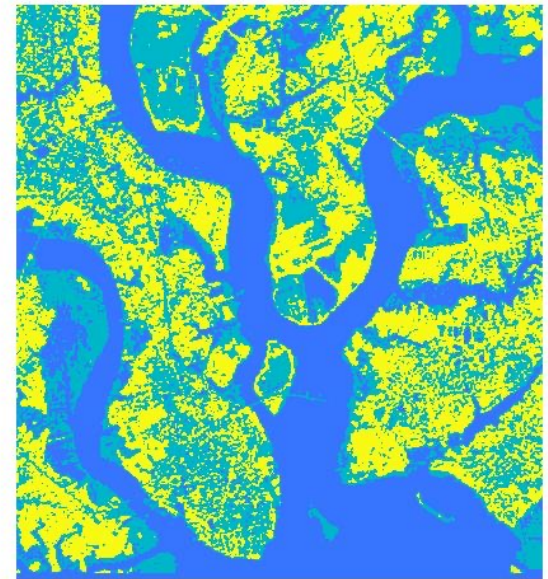
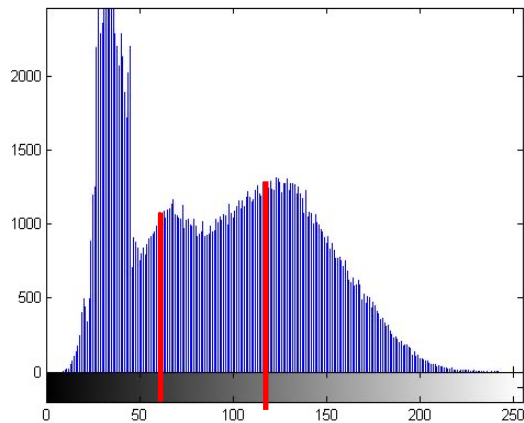
$$Z_i = \sum_k \pi_k N(x_i | \mu_k, \Sigma_k)$$

- Start with initial set :  $\{\pi_k, \mu_k, \Sigma_k\}$ .
- E-Step (Expectation stage)
  - Compute probability ( $z_{ik}$ ) of  $x$  belonging to  $k$ th Gaussian cluster.
  - Assign  $x$  to the  $m$ th cluster whose probability is maximum.
- M-Step (Maximization Stage)
  - Re-estimate parameters ( $\{\pi_k, \mu_k, \Sigma_k\}$ ) from class distribution
- Iterate above two steps till it converges.

Optional step.  
Decision to be  
taken at the end. →



# EM thresholding: An example



Intervals:  $[0, 60]$ ,  $[61, 119]$ ,  $[120, 255]$

Blue

Green

Yellow



# K-means clustering: Segmentation through estimating means

- Given  $N$   $d$ -dimensional data points,
  - compute  $K$  partitions (clusters) in them
    - so that it minimizes the sum of square of distances between a data point and the center of its respective partition (cluster).

## Optimization problem

### Minimization of Sum of Squared Errors (SSE)

$$E = \sum_k \sum_{\forall x \in C_k} \|x - c_k\|^2$$

where

$$c_k = \frac{1}{|C_k|} \sum_{\forall x \in C_k} x$$

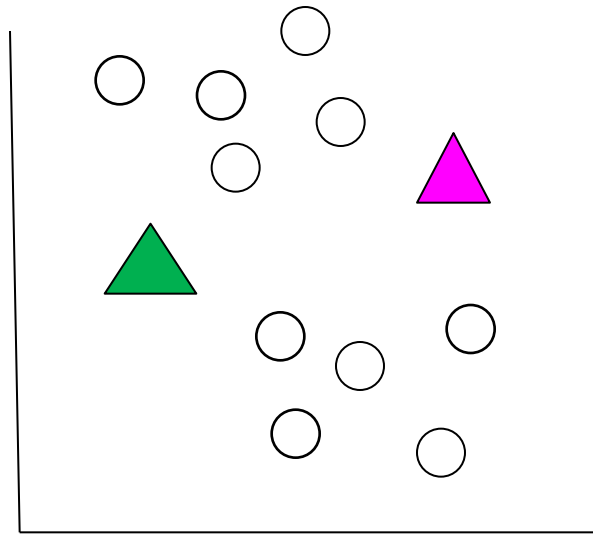


# The Lloyd algorithm (1957) (Batch K-Means)

- Given  $K$  initial centers, assign a point to the cluster represented by its center, if it is the closest among them.
- Update the centers.
- Iterate above two steps, till the centers do not change their positions.



# K-means: example ( $k=2$ )

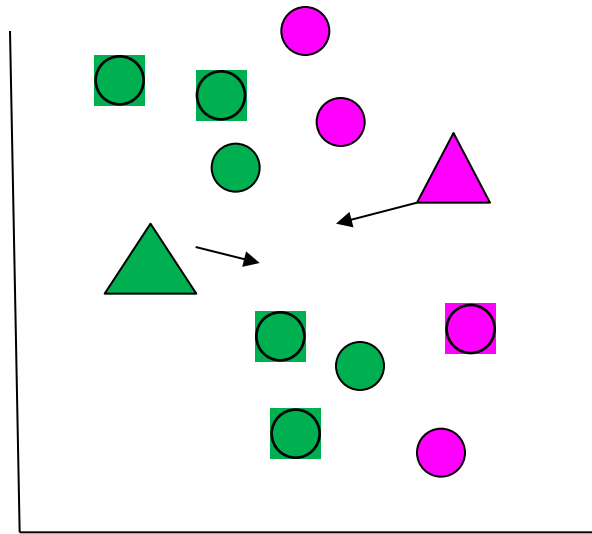


Choose initial centers.

Compute partitions.



# K-means: example (k=2)



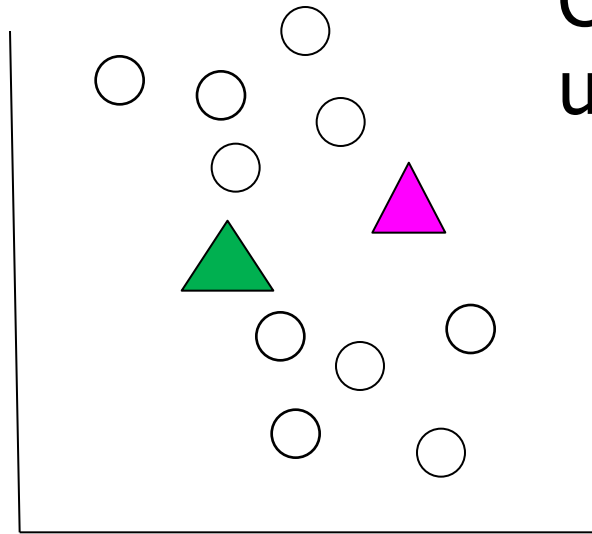
Compute partitions.

Update centers.





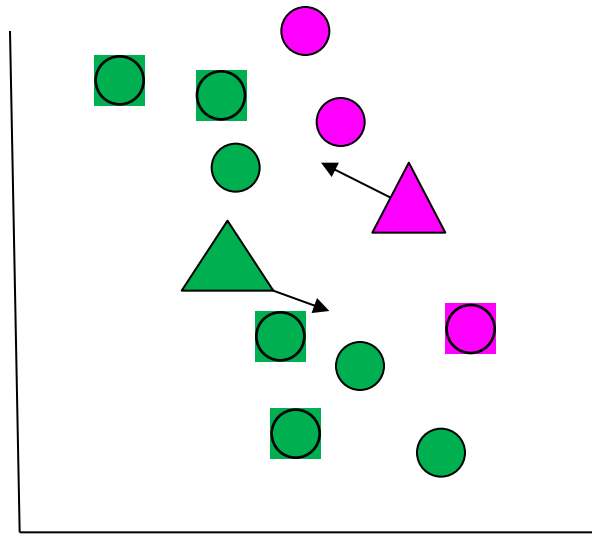
# K-means: example (k=2)



Compute new partitions with updated centers.



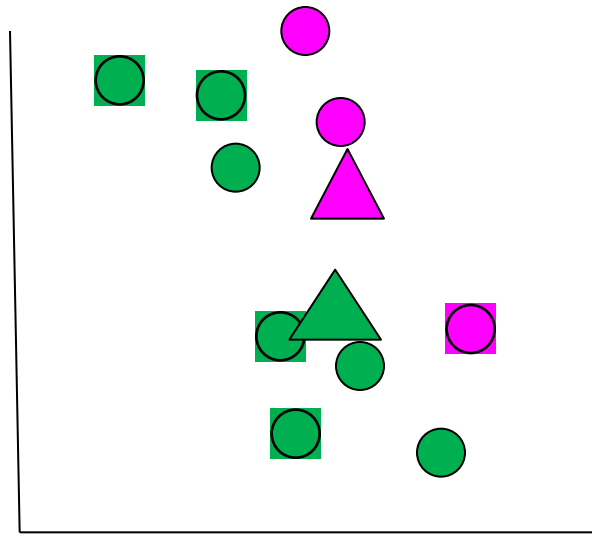
# K-means: example (k=2)



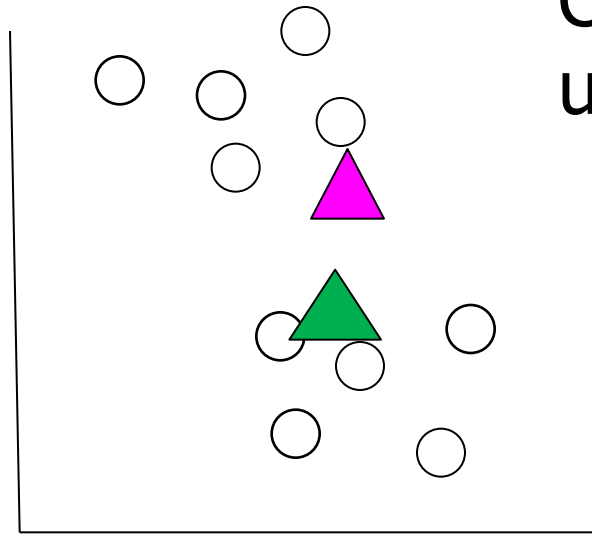
Update centers.



# K-means: example (k=2)



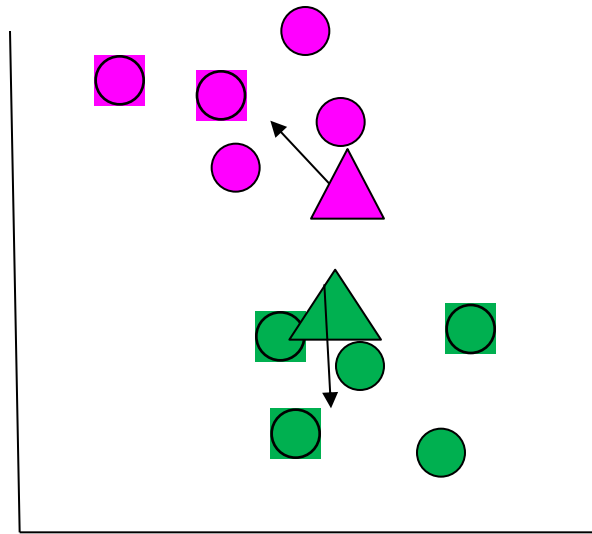
# K-means: example ( $k=2$ )



Compute new partitions with updated centers.



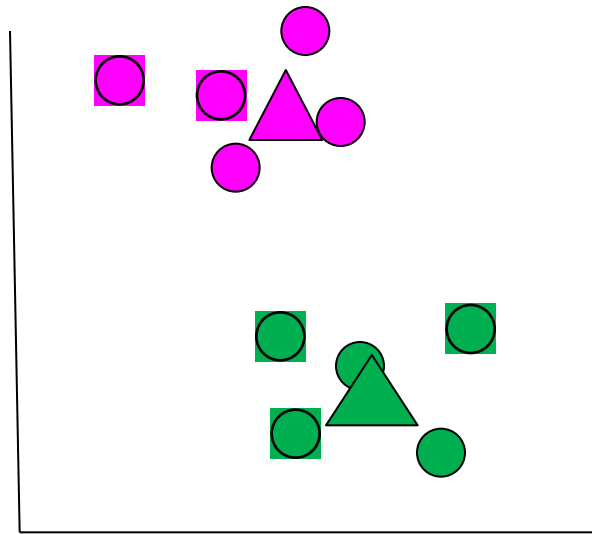
# K-means: example (k=2)



Update centers.



# K-means: example ( $k=2$ )



Stop at no change  
(or a very little  
change in cluster  
centers).



# Histogram analysis using K-Means clustering

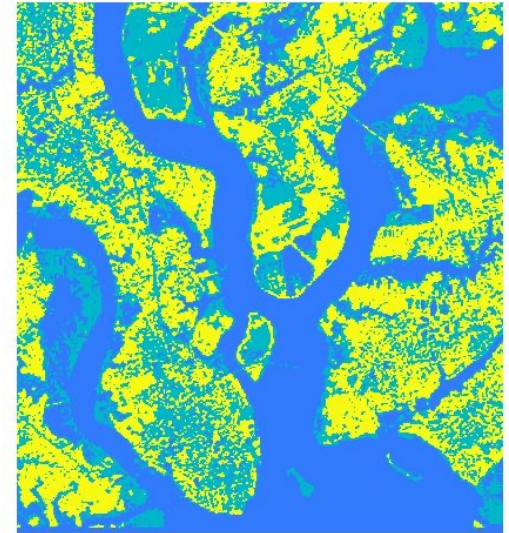
- Compute the histogram  $h(x)$ ,  $x=0,1,2,...N$
- Choose initial  $k$  brightness levels for the set of means, say  $m_1, m_2, ..., m_k$ , such that  $0 < m_1 < m_2 < ... < m_k < N$
- Update the  $i$ th means, for all  $i$

$$m_i = \frac{\sum_{(m_{i-1}+m_i)/2}^{(m_i+m_{i+1})/2} xh(x)}{\sum_{(m_{i-1}+m_i)/2}^{(m_i+m_{i+1})/2} h(x)}$$

- Iterate till convergence



# K-Means Segmentation: An example

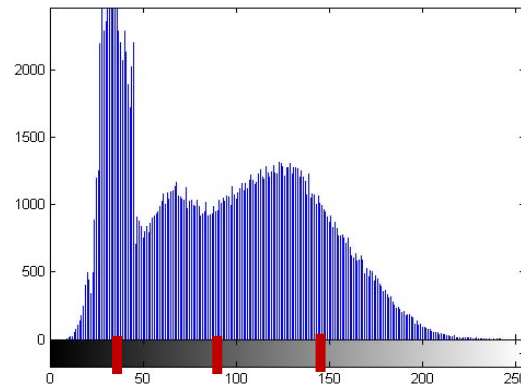


k-means for  
 $k=3$

$m_1=39,$

$m_2=90,$

$m_3=147$



Blue: 0-65

green: 66-119

yellow: 120-255





# Mean Shift Algorithm:

## Segmentation through estimating modes (peaks)

- A probability density estimation method
  - Non-parametric
  - Does not require the number of components ( $K$ ) as in K-Means.
- Identify peaks of distribution.
- Each peak defines a cluster / segment.
  - Pixels / elements having shortest path to a peak defines a segment



# Probability density estimation

- Probability distribution using Parzen Windows
  - A function of some finite number of data points  $x_1 \dots x_n$

$$P(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n K(\mathbf{x} - \mathbf{x}_i)$$

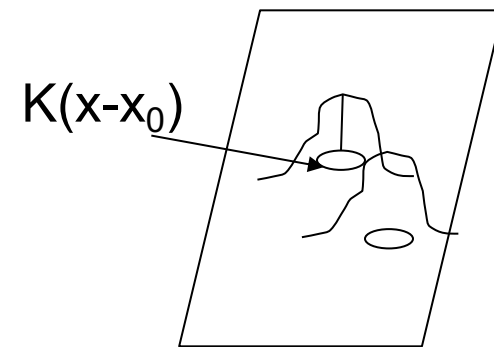
- Kernel Properties:

- Normalized
- Symmetric

$$\int_{R^d} K(\mathbf{x}) d\mathbf{x} = 1$$
$$\int_{R^d} \mathbf{x} K(\mathbf{x}) d\mathbf{x} = 0$$

- Exponential weight Decay

$$\lim_{\|\mathbf{x}\| \rightarrow \infty} \|\mathbf{x}\|^d K(\mathbf{x}) = 0$$



# Examples of kernels

- Epanechnikov 
$$K_e(\mathbf{x}) = \begin{cases} c(1 - \|\mathbf{x}\|^2) & \|\mathbf{x}\| \leq 1 \\ 0 & \textit{otherwise} \end{cases}$$

- Uniform 
$$K_u(\mathbf{x}) = \begin{cases} c & \|\mathbf{x}\| \leq 1 \\ 0 & \textit{otherwise} \end{cases}$$

- Gaussian 
$$K_g(\mathbf{x}) = ce^{-\frac{1}{2}\|\mathbf{x}\|^2}$$



# Mode selection

- Compute the gradient of distribution

$$\nabla P(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \nabla K(\mathbf{x} - \mathbf{x}_i) \quad K(\mathbf{x} - \mathbf{x}_i) = ck \left( \left\| \frac{\mathbf{x} - \mathbf{x}_i}{h} \right\|^2 \right)$$

$$\nabla P(\mathbf{x}) = \frac{c}{n} \sum_{i=1}^n \nabla k_i = \frac{c}{n} \left[ \sum_{i=1}^n g_i \right] \cdot \left[ \frac{\sum_{i=1}^n \mathbf{x}_i g_i}{\sum_{i=1}^n g_i} - \mathbf{x} \right]$$

Size of window  $h$

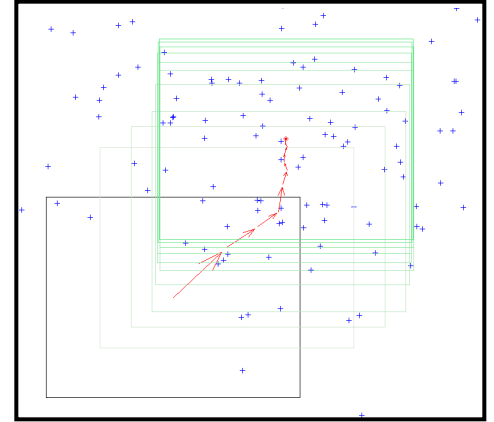
$g(\mathbf{x}) = -k'(\mathbf{x})$

$$\nabla P(\mathbf{x}) = 0 \quad \Rightarrow \quad \mathbf{m}(\mathbf{x}) = \frac{\sum_{i=1}^n \mathbf{x}_i g \left( \frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{h} \right)}{\sum_{i=1}^n g \left( \frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{h} \right)} - \mathbf{x} = 0 \quad \Rightarrow \quad \mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + \mathbf{m}(\mathbf{x}^{(i)})$$



# Mean shift algorithm

- Searches a *mode* or local maximum of density of a given distribution from a point
- Choose a search window (width and location)
- Compute the mean of the data in the search window
- Center the search window at the new mean location
- Repeat until convergence
  - Converged point is a mode.
- From every point do the same.
- Set of points arriving at the same mode forms a segment

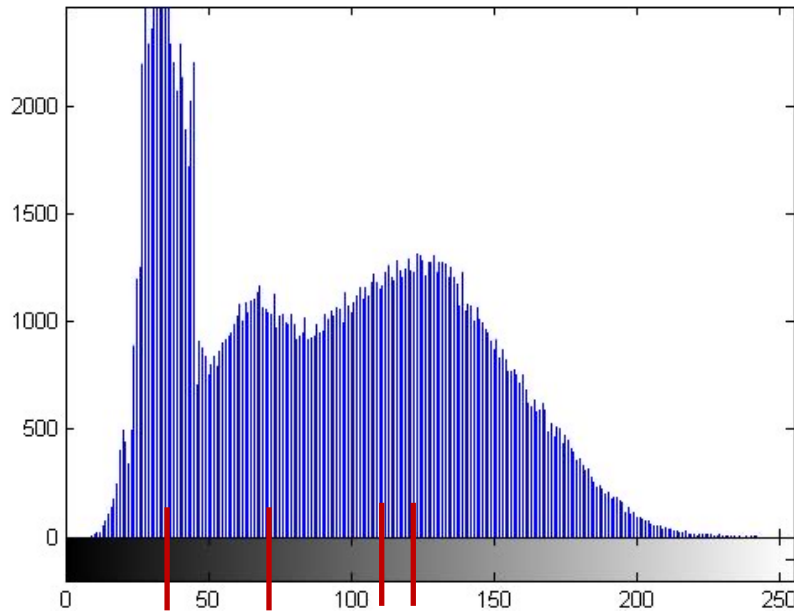


# Mean shift analysis of histogram

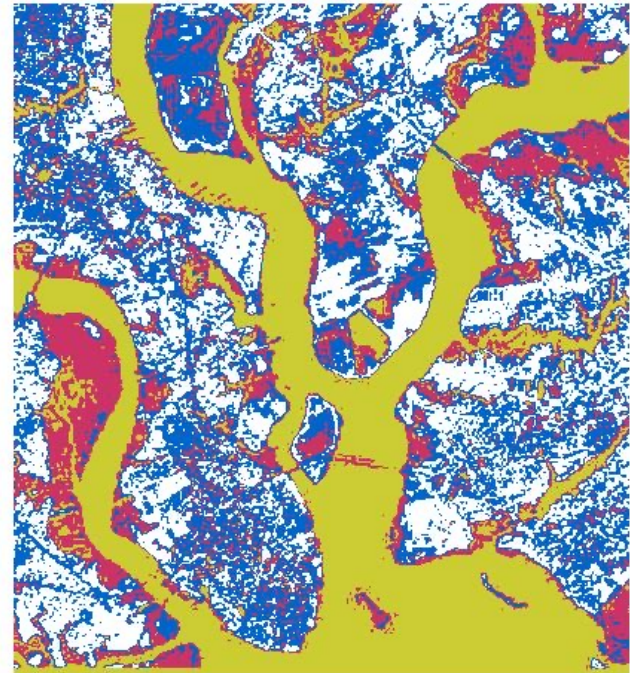
- Compute the histogram  $h(x)$ ,  $x=0,1,2,...N$
- For each  $x$  determine the mode  $m(x)$  in the histogram.
- Ensure monotonicity in  $m(x)=0,1,2,...,N$ 
  - $m(x_1) \leq m(x_2)$ , for  $x_1 < x_2$
- Prune spurious modes
  - sufficient support (brightness interval),
  - sufficient gap between adjacent distinct pair, and
  - sufficient strength (number of pixels).
- Get brightness interval for each mode



# Mean shift analysis of histogram and segmentation



modes:  
32,67,117,130

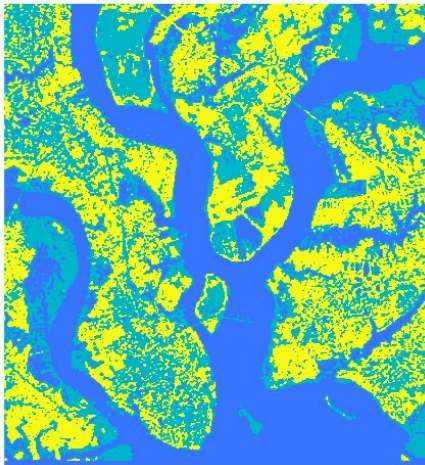
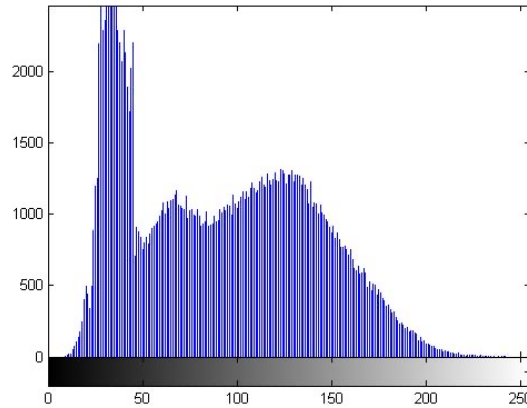


yellow:[1,51]  
red:[52,82]  
blue:[83,128]  
white:[129,256]

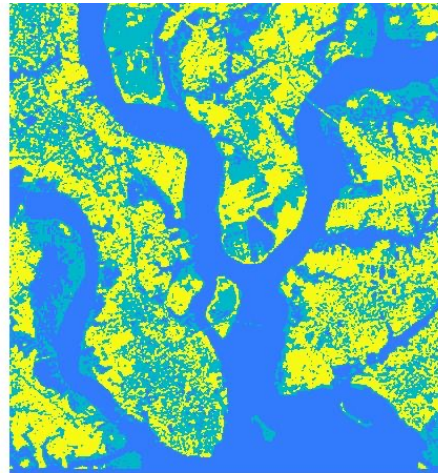




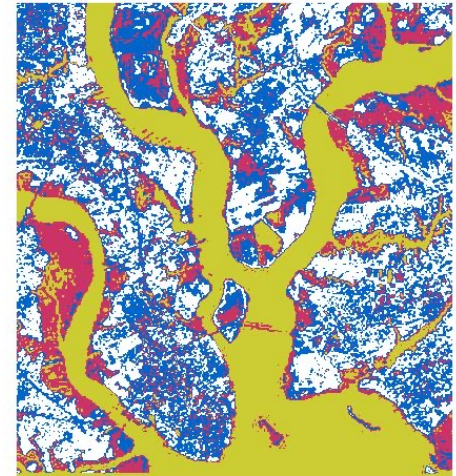
# Segmentation results from various approaches



GMM



K-Means

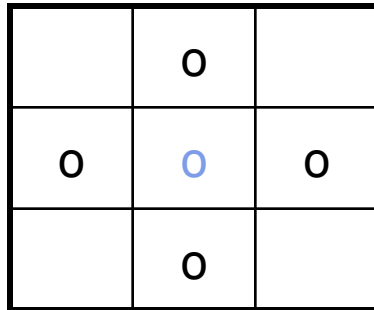


Mean Shift

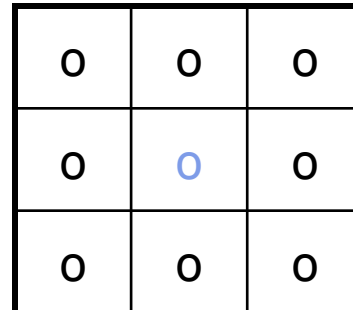


# Component Labeling (Region growing)

- Partitioning **connected** image pixels into meaningful non-overlapping sets.
  - Neighborhood definition.
  - 4-neighbor, 8-neighbor



4-neighbors



8-neighbors



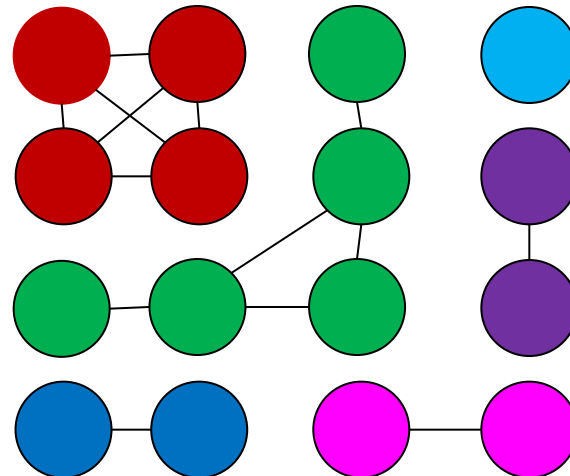


0	0	0
0	0	0
0	0	0

# Component Labeling

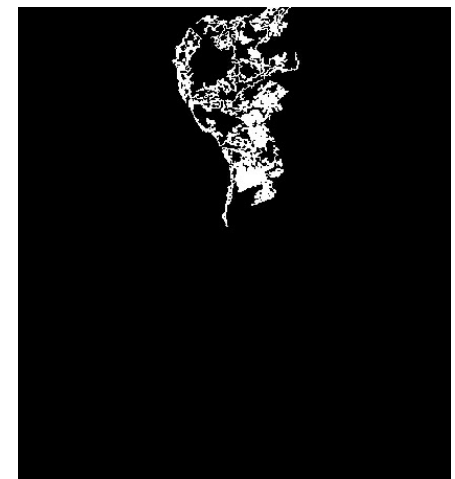
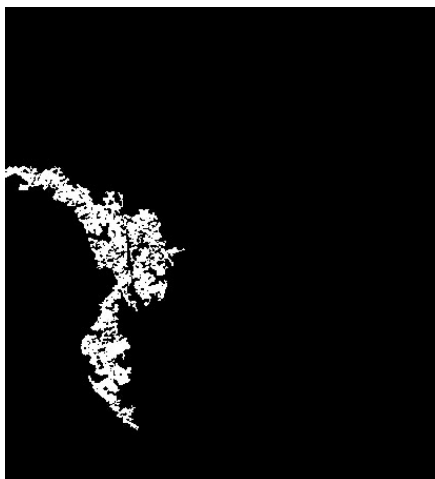
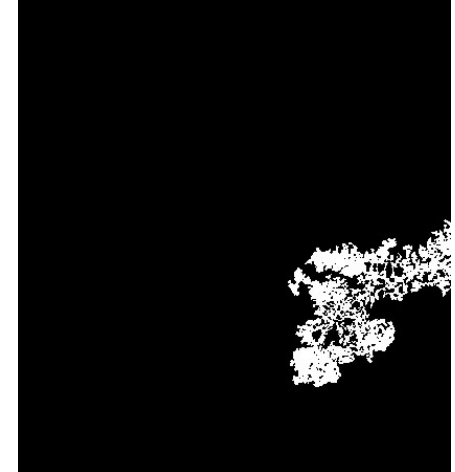
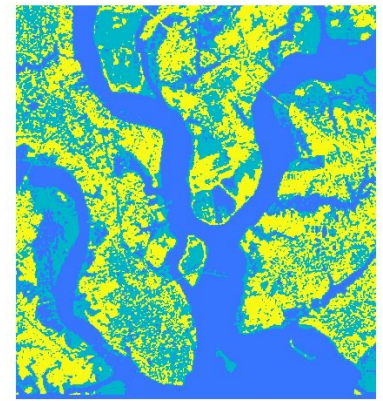
- Form graph with edges between neighboring pixels having same labels.
- Compute connected components.
  - Graph traversal algorithms

20	20	50	20
20	20	50	100
50	50	50	100
100	100	20	20

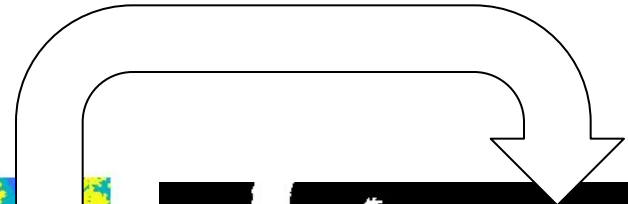
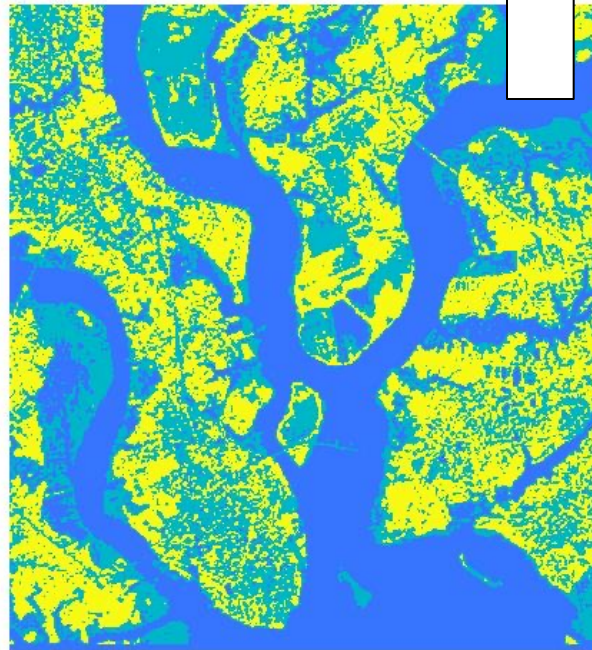


Do you require  
an explicit  
graph  
representation?  
Can you  
compute using  
only the image  
array?

# Examples of components



?



Why a part of river channel is missing?



# Morphological watersheds

- Topographic interpretation of a point in the functional space
- Three types of points
  - Points belonging to a regional minimum
  - Points from which a water-drop will fall to a minimum
    - The set of such points corresponding to the regional minimum forms the catchment basin (a segment)
      - Connected
  - Points from which a water-drop is equally likely to fall toward multiple minimum
    - Points are near the boundaries of the catchment basins and adjacent basins need to be kept separated by building dams at their meeting points.
      - Core catchment area incrementally dilated from both side to detect the meeting points.



# Simulating flooding process

- $T(n)$ : Set of points having brightness value  $\leq n$
- $X_{\min}$ ,  $X_{\max}$ : Minimum and maximum brightness values
- $\{M_1, M_2, \dots, M_k\}$ : The set of regional minima identified for formation of catchment basin.
  - Each as a pit for forming catchment basin with flooded water.
- $C_i(M_j)$ : Catchment basin of  $M_j$  at level  $i$ 
  - set of points connected to  $M_j$  having brightness value  $\leq i$
  - $C_{\max}(M_i)$ : Full catchment basin of  $M_i$
- $C(n)$ : Union of catchment basins at level  $n$ ;
  - Union of  $C_n(M_i)$ 's
  - The same as  $T(n)$  - {set of points building dam}



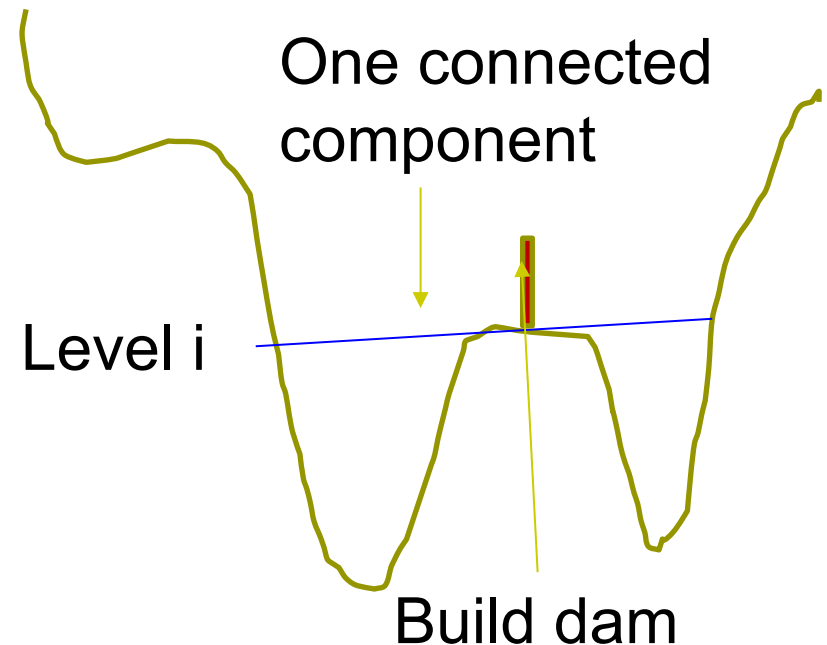
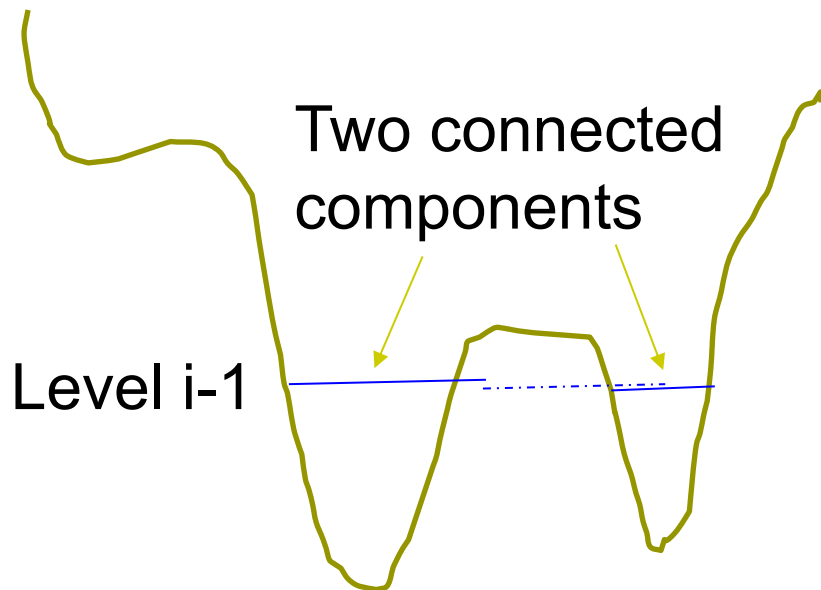
# Watershed segmentation

- The process to keep the catchment basins separated by building dams
  - at the meeting points of catchment basins while water level rises from the bottom to top
  - from  $X_{\min}$  to  $X_{\max}$





# Flooding process and dam building



- $T(i-1)$ : Number of connected components: 2

- $T(i)$ : Number of connected component: 1

Decrease of number of connected component in consecutive levels indicates the event of dam building.



# Watershed segmentation

- $C(i-1)$ : Union of catchment basins at level  $i-1$  having
- $T(i)$ : Set of pixels having having brightness value  $\leq I$
- $C(X_{\min})=T(X_{\min})$  and  $C(X_{\max})=U C_{\max}(M_i)$
- Approach: Incrementally compute  $C(i)$  from  $C(i-1)$ 
  - Start from  $C(X_{\min})$  and Stop at  $C(X_{\max})$ .
- Perform connected component analysis from  $T(i)$ 
  - Check each component  $q$  of  $T(i)$
  - Perform  $q \cap C(i-1)$ 
    - If it is NULL  $\rightarrow$  New catchment basin, include in  $C(i)$
    - If it is one component  $\rightarrow$  No spill over, include  $q$  in  $C(i)$
    - If it is multiple components  $\rightarrow$  Spill over takes place.
      - Expand each through dilation and build dam at meeting points of expansion, and include those modified components..



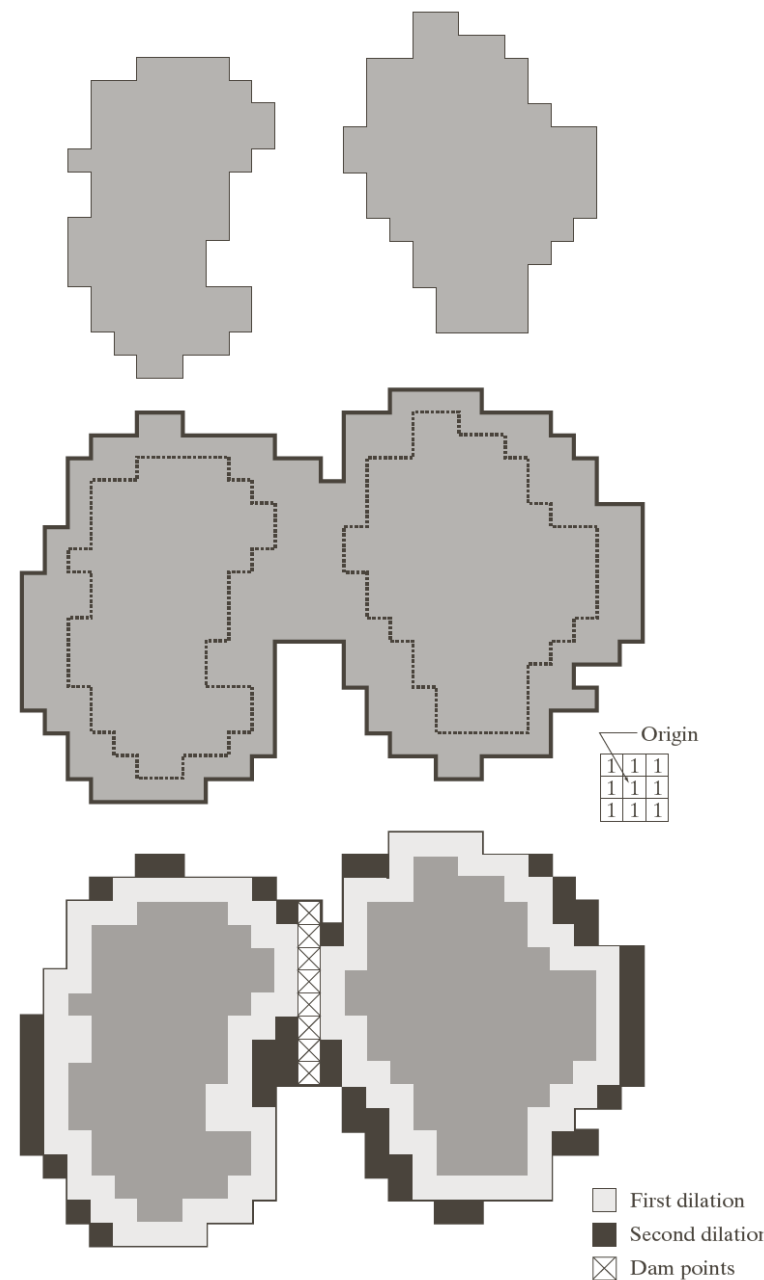
# Building dam at meeting points of expansion

- Dilate each component of  $q \cap C(i-1)$  in steps with a constraint that only additional points from  $q$  are included in those expanded components.
- SE: Disk of 8-Neighbor distance of radius 1, Size: 3x3
- The common points of multiple components are not included in the expanded components and are marked for the separating dam.
- One pixel width
- Stop when  $q$  is completely covered
- A point belongs to either one of the components of  $C(i-1)$  or to the dam.



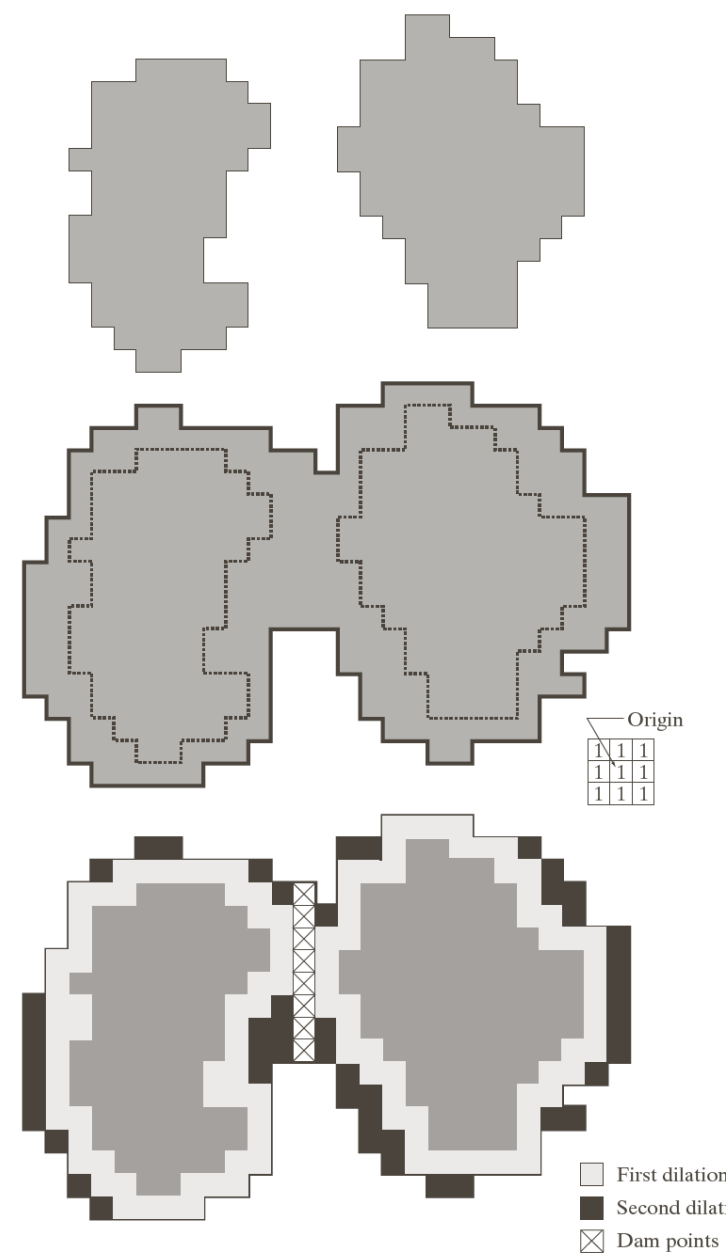
# Dam Construction

- $M_1, M_2$ :
  - Sets of coordinates of points in the two regional minima
- $C_{n-1}(M_1), C_{n-1}(M_2)$ 
  - Sets of coordinates of points in the catchment basins associated with  $M_1, M_2$  at stage  $n-1$  of flooding (catchment basins **up to the flooding level**)
- $C[n-1] = C_{n-1}(M_1) \cup C_{n-1}(M_2)$



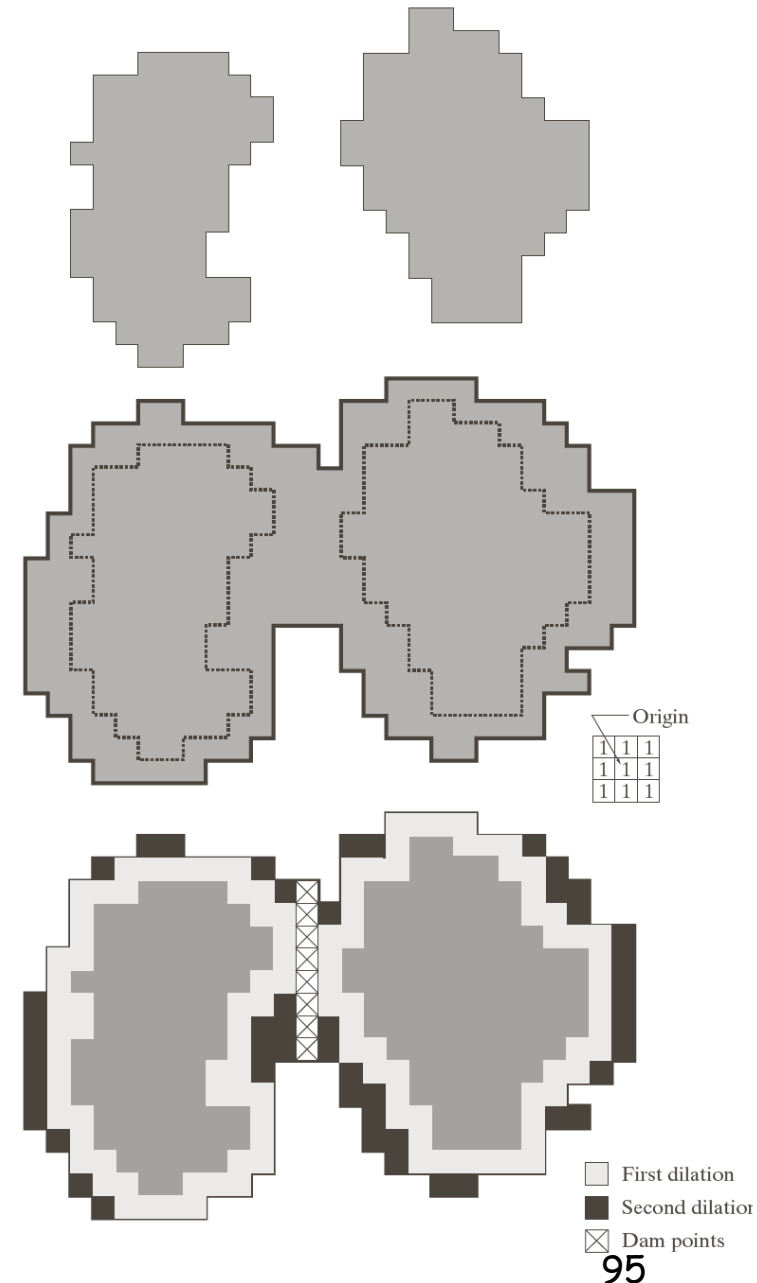
# Dam Construction

- At flooding step  $n-1$ , there are two connected components. At flooding step  $n$ , there is only one connected component
- The water between the two catchment basins has merged at flooding step  $n$
- Use “ $q$ ” to denote the single connected component
- Steps
  - Dilate  $C_{n-1}(M_1)$ ,  $C_{n-1}(M_2)$  by the  $3 \times 3$  SE, subject to:
    - Constrain SE to  $q$
    - Dilation cannot merge the sets



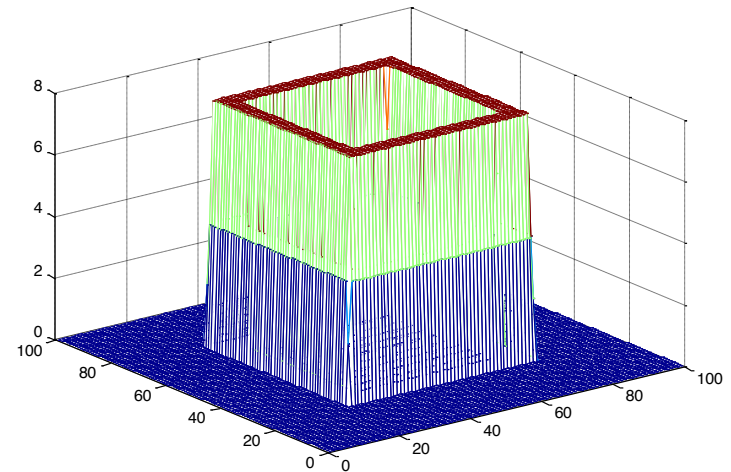
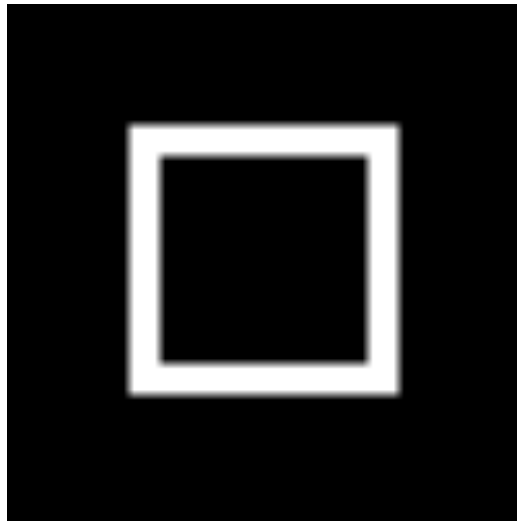
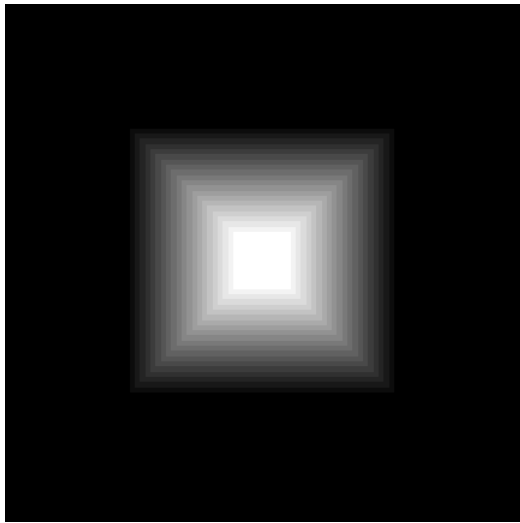
# Dam Construction

- The dam is constructed by the points on which the dilation would cause the sets being dilated to merge.
- Resulting one-pixel thick connected path
- Setting the gray level at each point in the resultant path to  $\max+1$

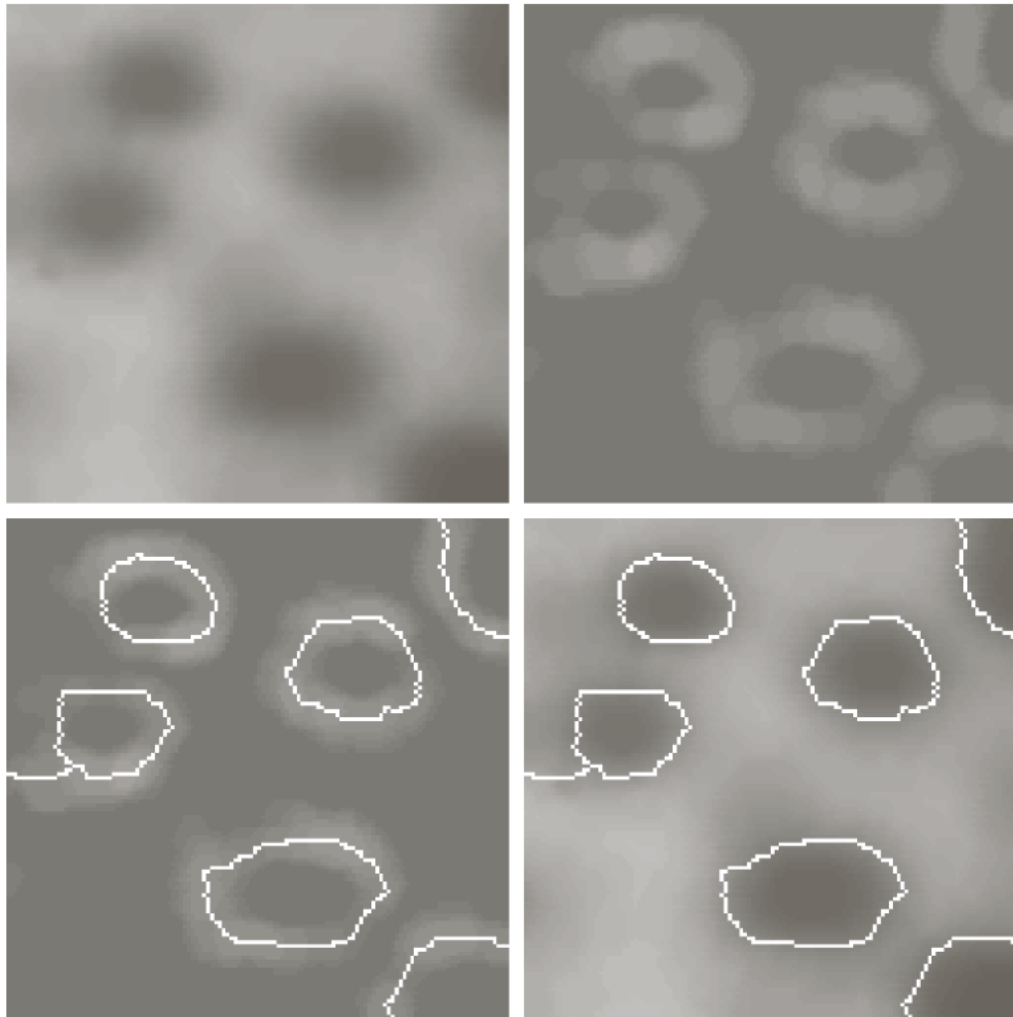


# Watershed segmentation: A few tips and issues

- Instead of working on an image itself, this technique is often applied on its **gradient image**.
  - Each object is distinguished from the background by its up-lifted edges



# Watershed segmentation: Typical results



a	b
c	d

**FIGURE 10.56**

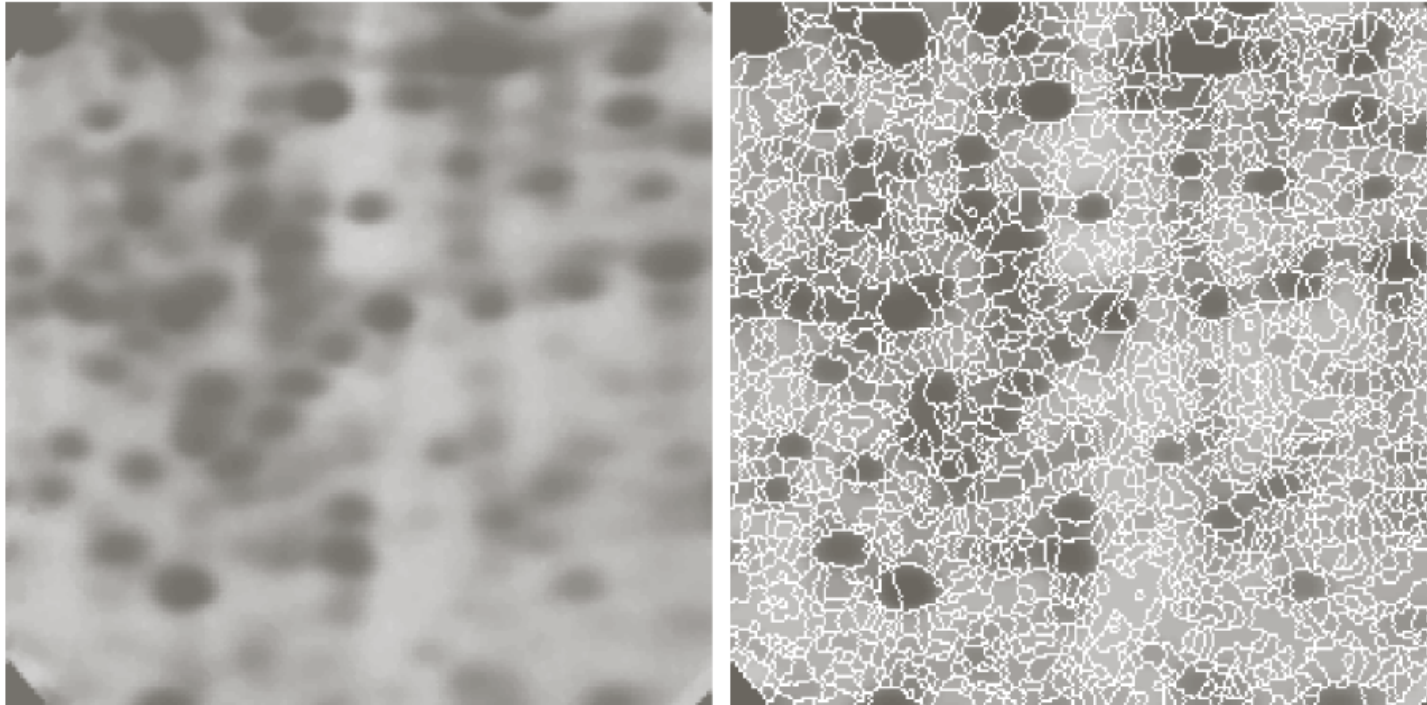
(a) Image of blobs.  
(b) Image gradient.  
(c) Watershed lines.  
(d) Watershed lines superimposed on original image.  
(Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)





# Watershed Segmentation: Use of markers

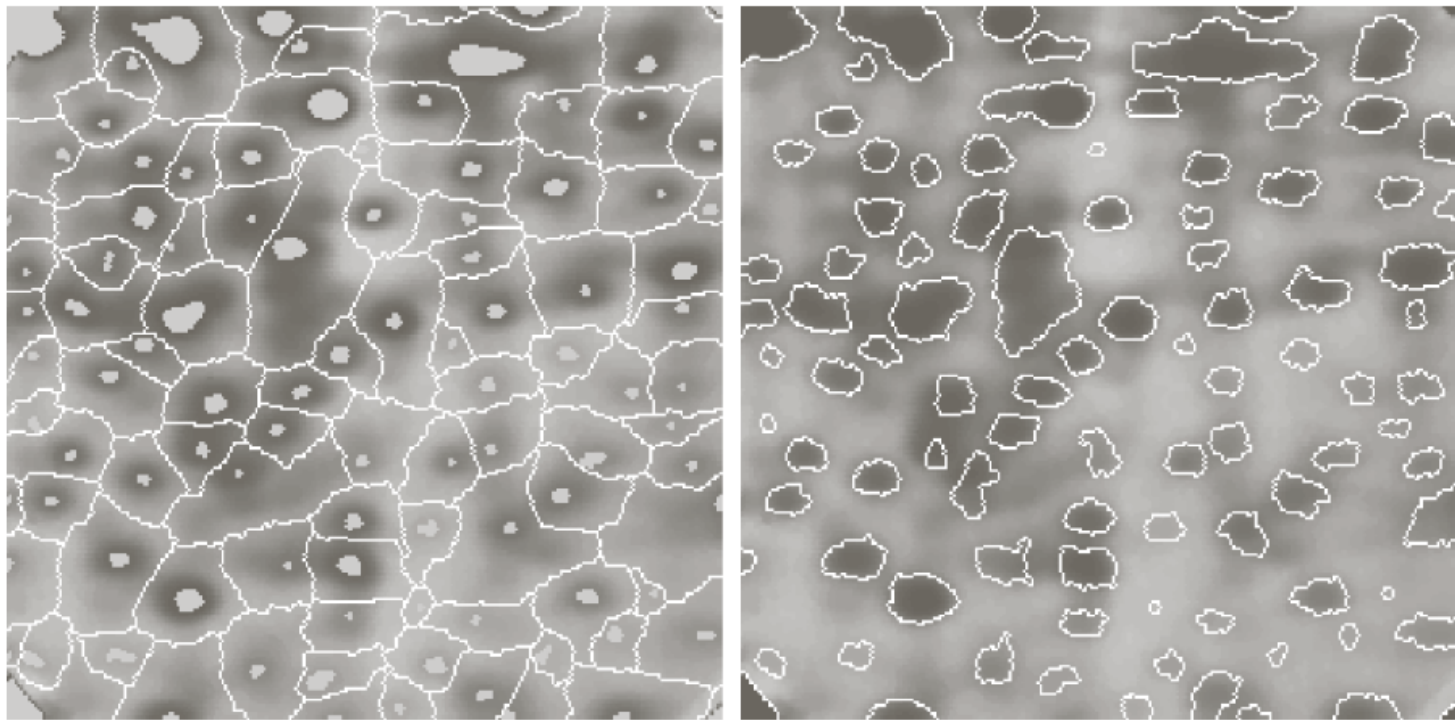
Over segmentation



- Use of marker
  - A connected region defining the pit or the dam.
  - Internal marker within object: Pit
  - External marker in background: Dam



# Example results using markers



a b

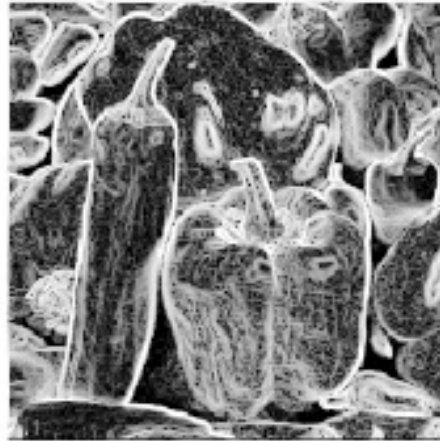
**FIGURE 10.58** (a) Image showing internal markers (light gray regions) and external markers (watershed lines). (b) Result of segmentation. Note the improvement over Fig. 10.47(b). (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)



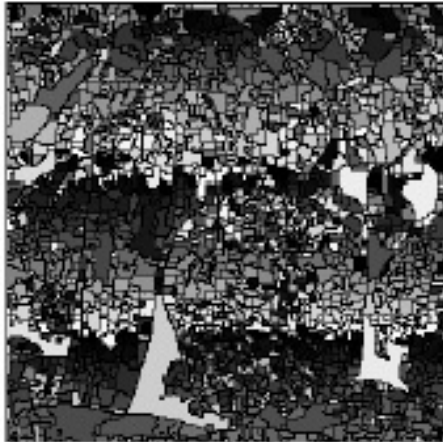
# Another Example



(a)



(b)



(c)



(d)

- a) Original Image
- b) Gradient Image:  
3X3 Sobel Edge  
Detector
- c) Raw Watershed
- d) Controlled  
Watershed with  
Region Marking



# Summary

- Meaningful partitioning of image pixels
  - Edges, Regions
- Edge operators: First order and Second order derivatives (Laplacian)
  - Marr-Hildreth, Canny
- Analysis of histogram
  - Binarization through classification
    - Bayes' classification rule applied
  - Binarization through maximizing interclass variances.
    - Otsu thresholding
  - By finding peaks / valleys and declare intervals of brightness value for a segment.
    - Gaussian mixture model
    - Mean shift algorithm



# Summary

- K-means clustering in the feature space, if the cluster number known.
- Component labeling for connected components.
- Watershed segmentation algorithm



Thank You

