CSCI 435/MCS9435 Computer Vision

Image Segmentation (I)

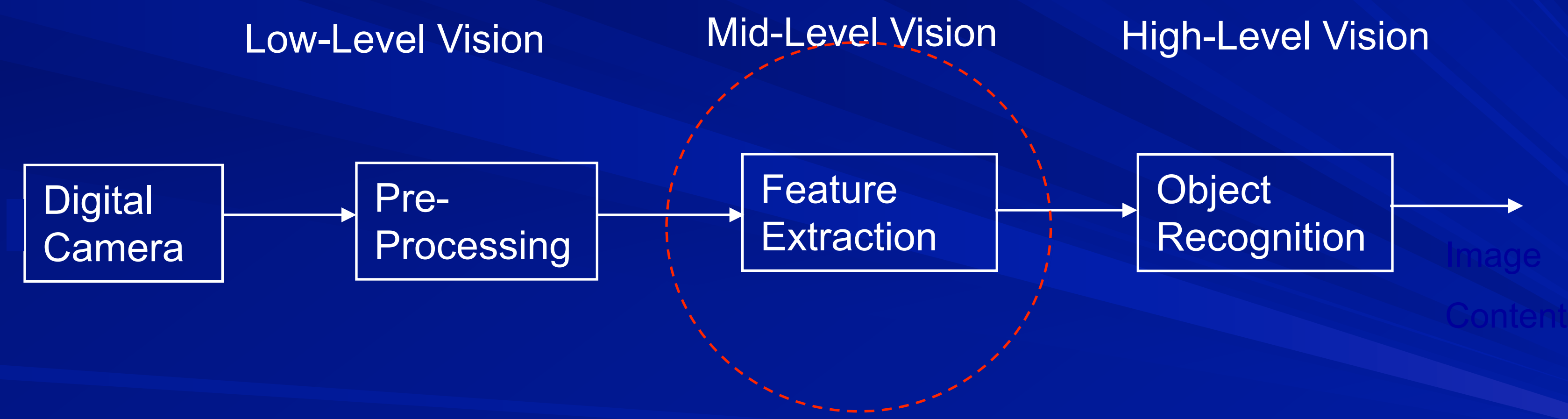
Lecturer: Wanqing Li, PhD
Room: 3.101
email: wanqing@uow.edu.au
http://www.uow.edu.au/~wanqing

Outline

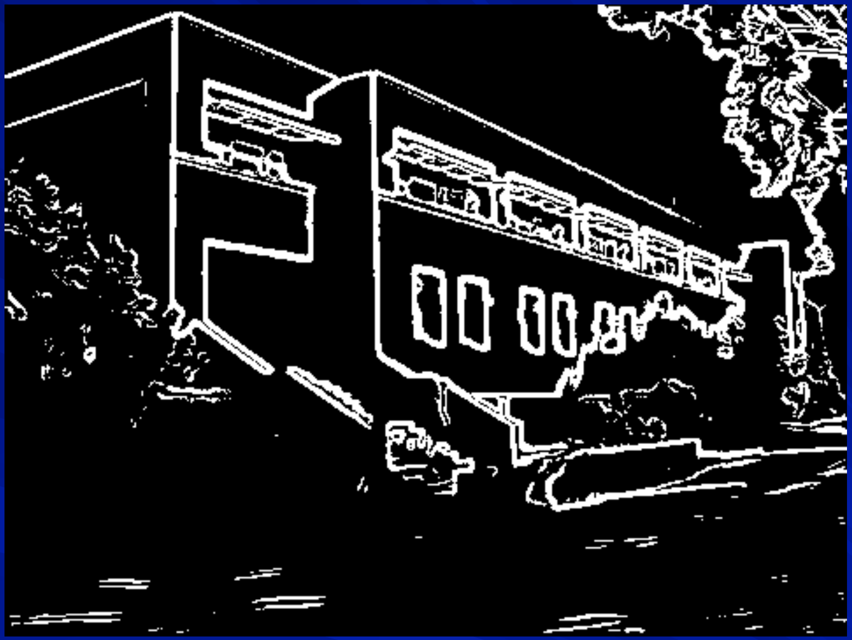
- What is segmentation
- Region segmentation
  - Clustering based approach
  - Contour based approach
- Applications of segmentation
- CV & IP tools

Machine Vision Concept (review)

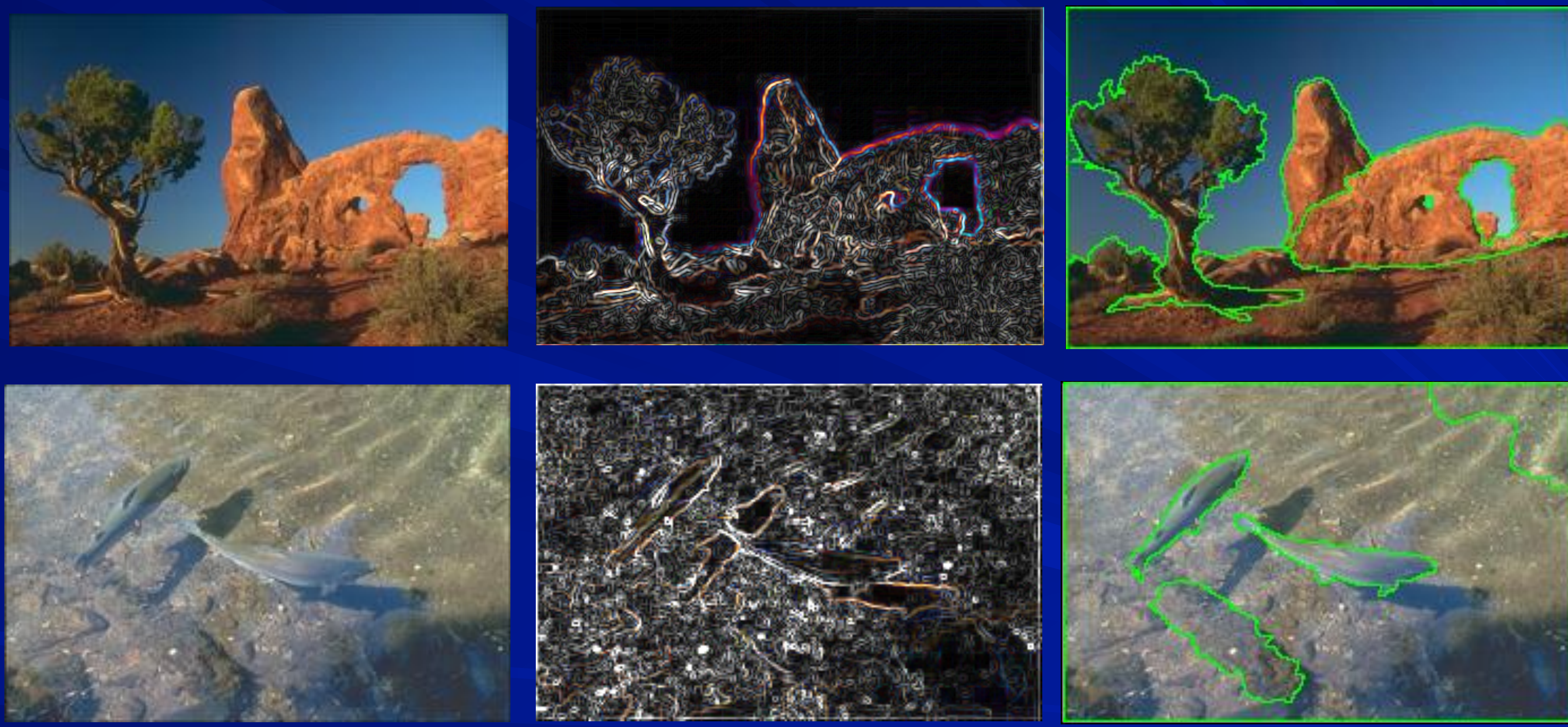
Machine Vision is a multistage process where each previous stage affects performance of all following stages



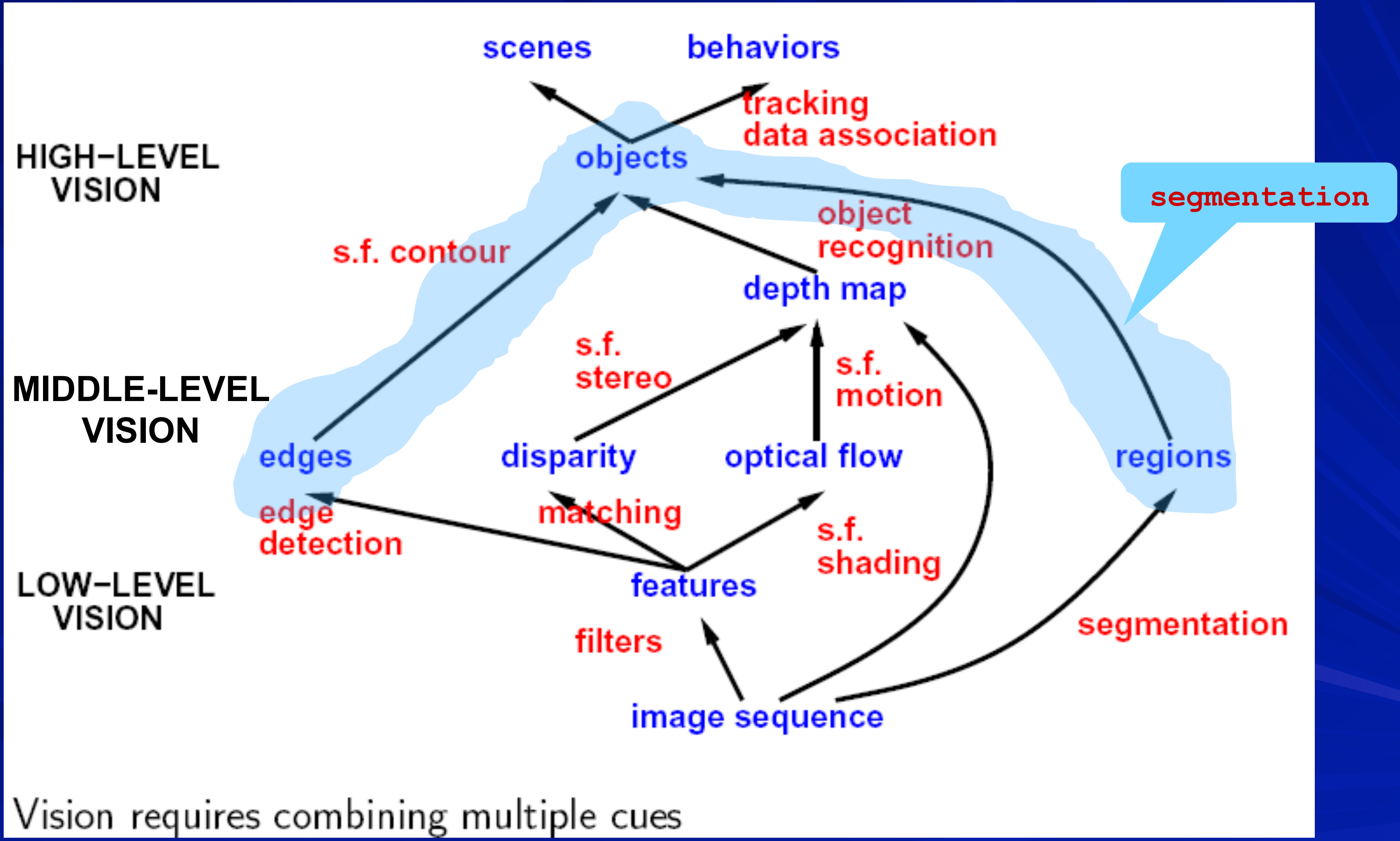
Middle-Level Vision



Middle-Level Vision



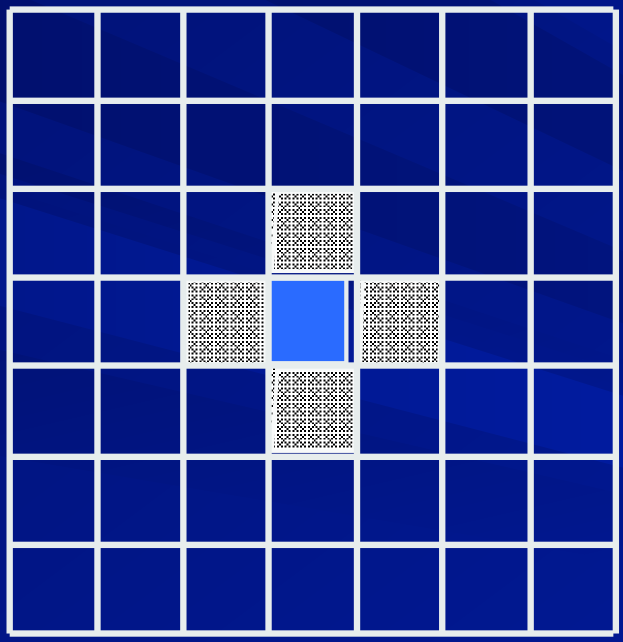
Vision Subsystems



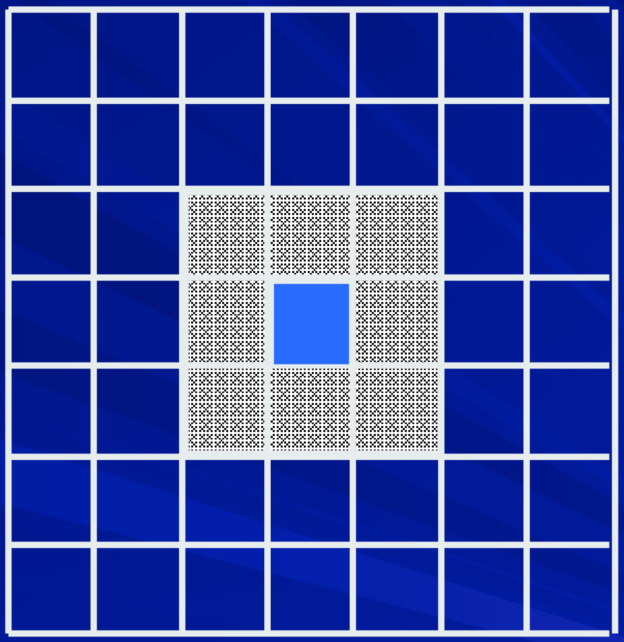
What is Segmentation

- Segmentation is defined as a process to decompose an image into segments, regions or objects.
  - The segments are meaningful in the context of the environment in which the segmentation output is to be used.
  - The process stops when the segments of interest in an application have been isolated
- Segments are usually non-overlapping
- Pixels in a segment
  - have similar visual properties.
  - are spatially connected

Spatial Connectivity



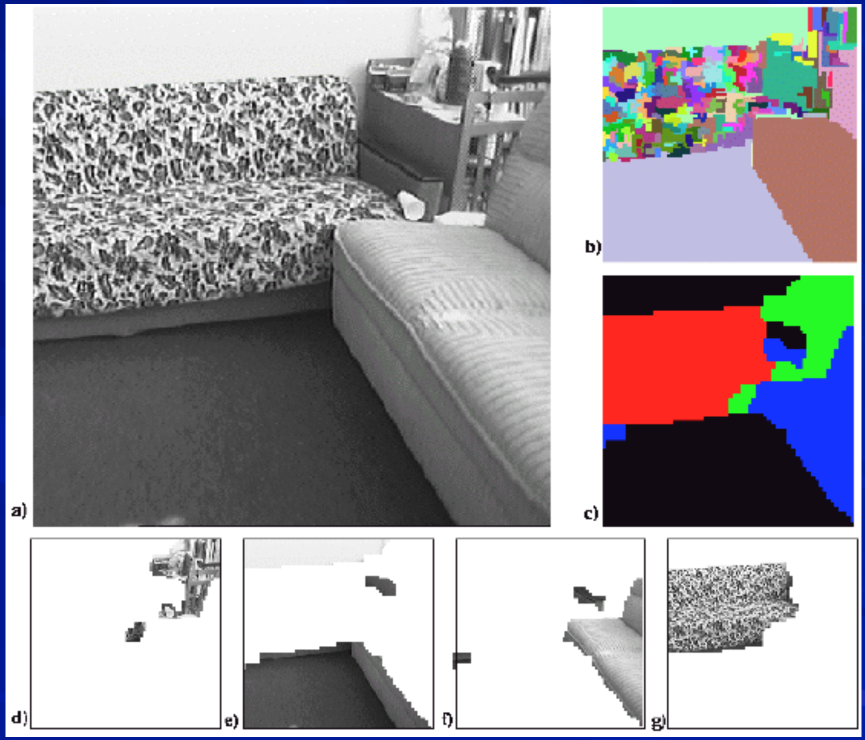
Four Connected



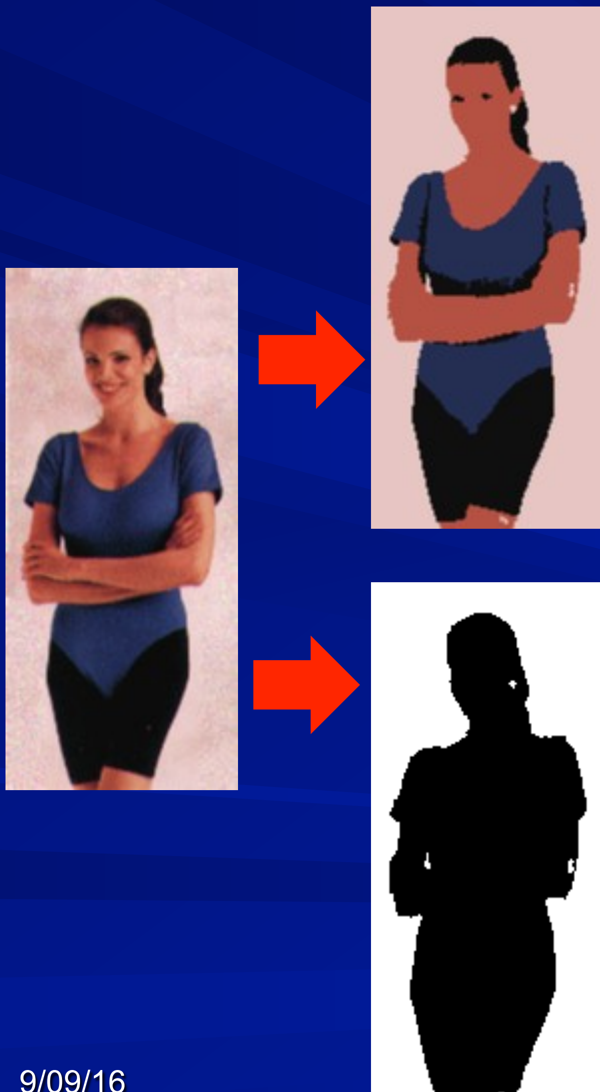
Eight Connected

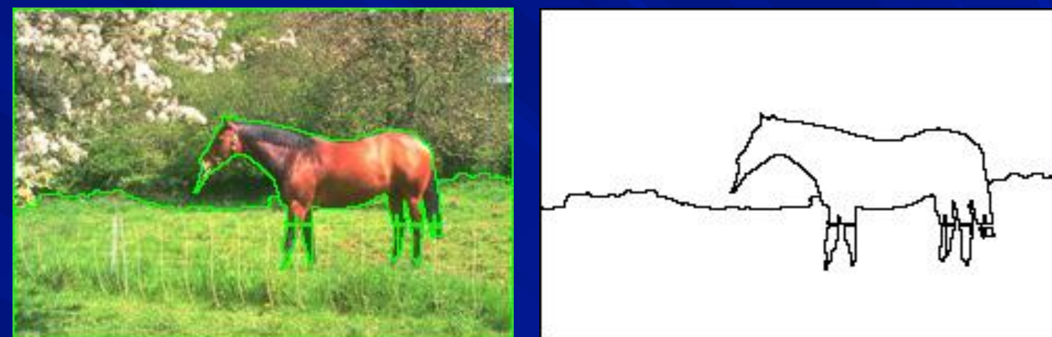
Visual Properties - Features

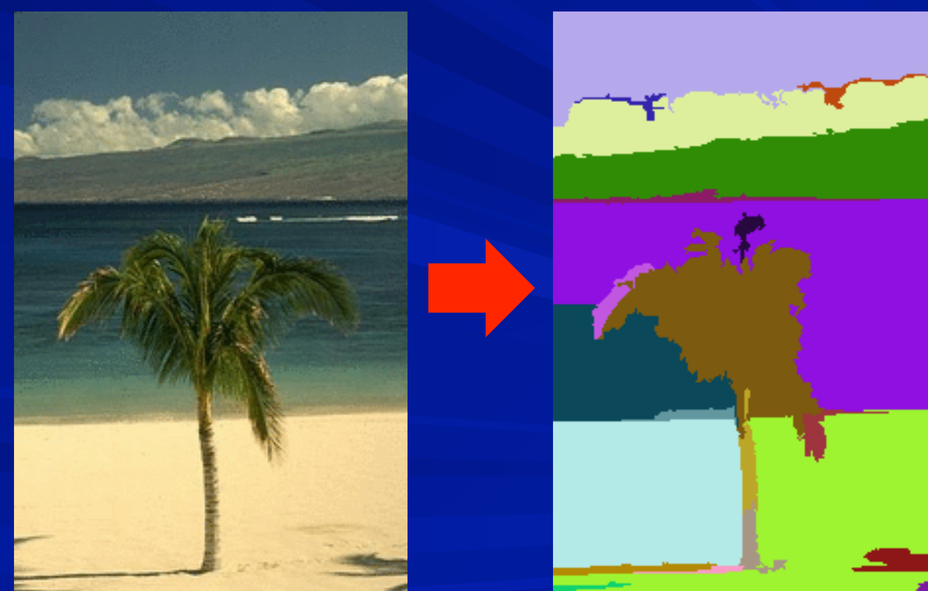
- Measurable features
  - Intensities or color
  - *Texture*
  - *Motion*
  - *Depth*



Segmentation – How?







General Approaches

- Clustering-based
  – group all pixels with similar visual property together
- Contour-based
  – Also known as edge-based
  – Detect edge between segments and form closed contours of segments
- Hybrid-approach

Clustering-based Image Segmentation

Clustering-based methods

Info from image:
- **luminance/color**
- **texture**
- **motion**

Feature Extraction → **Grouping pixels** → **Post Processing** → **Regions**

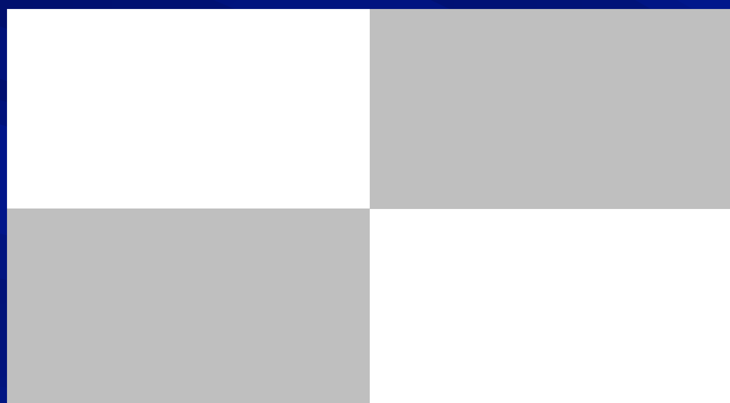
Contextual Info:
- **spatial/temporal coherence**

Sufficient information: most time, depending on applications

Grouping Pixels

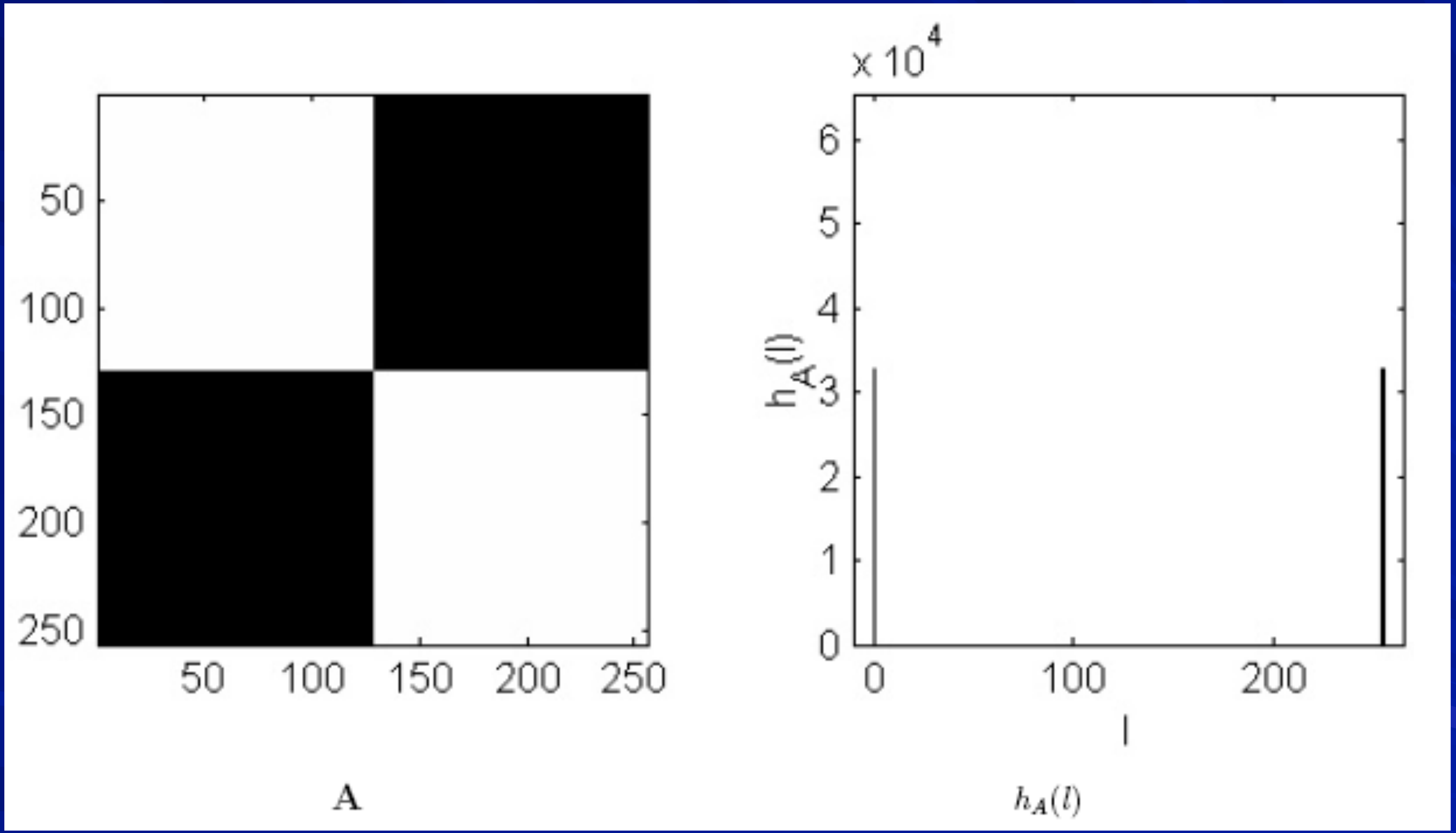
- Grouping the pixels into a K clusters such that
  - Pixels within a cluster have more similar feature than pixels across clusters
  - The spatial connectivity is not considered
- Questions
  - How to measure the similarity ?
  - What should the value of K be ?
  - How many segments or regions ?

A Simple Example



- How many clusters?
- How many regions?
- How to group the pixels?

Thresholding



Thresholding

Image with dark background and a light object

Image with dark background and two light objects

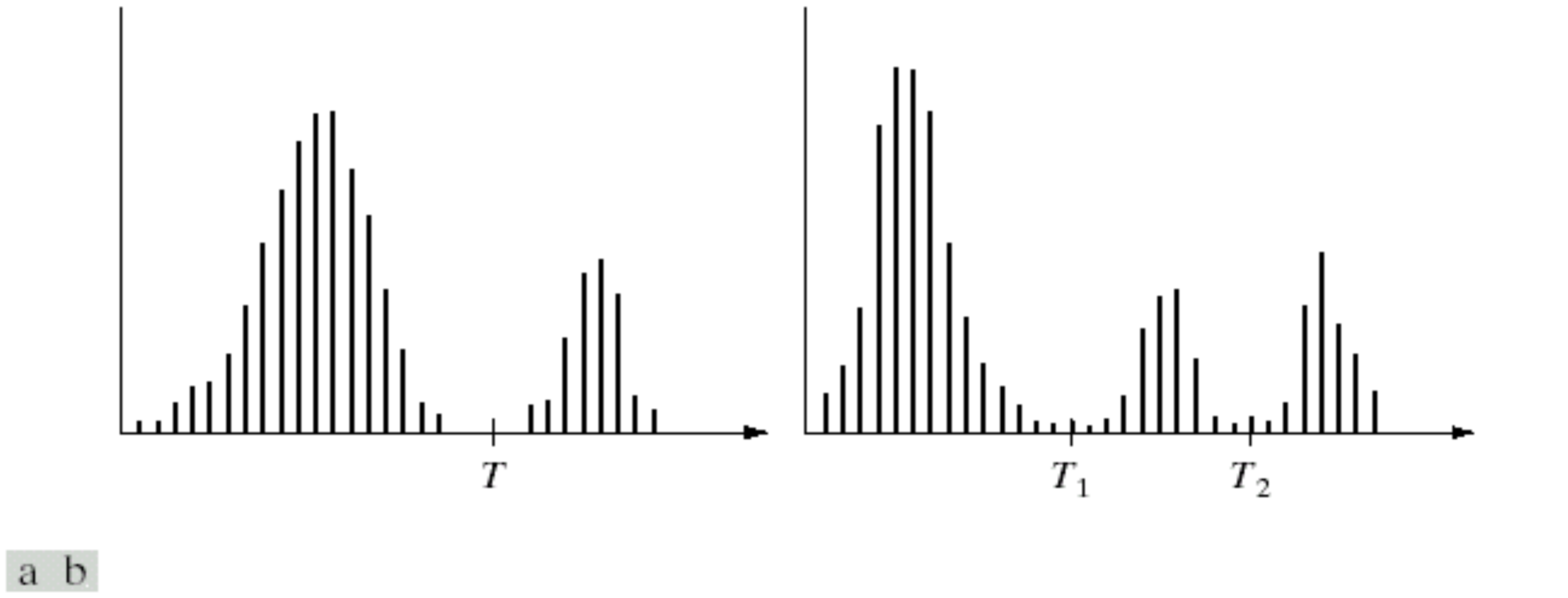
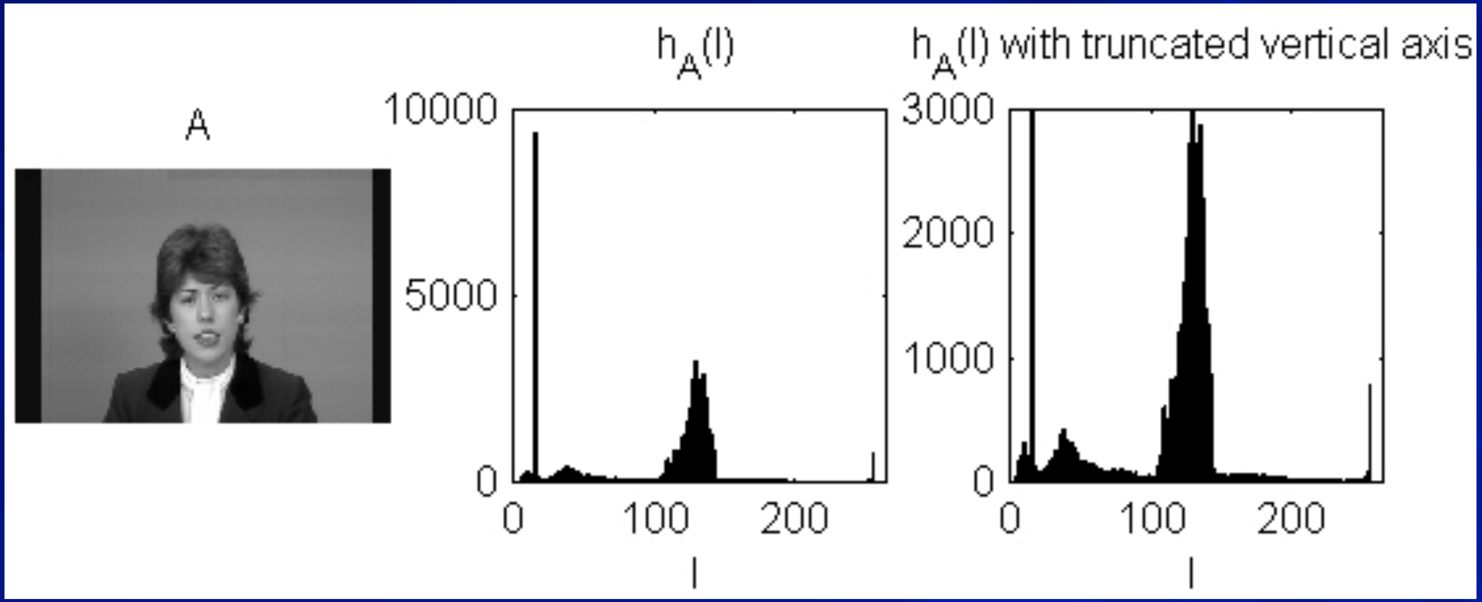


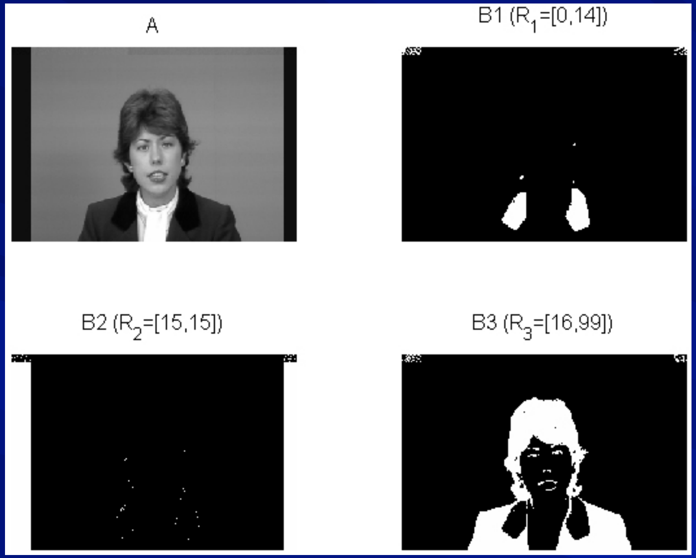
FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

An Example

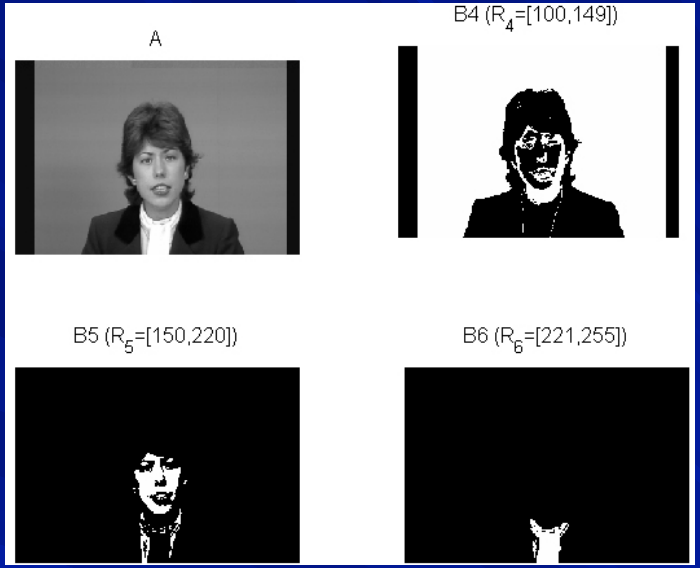


$R1 = [0; 14]; R2 = [15; 15]; R3 = [16; 99];$
$R4 = [100; 149]; R5 = [150; 220]; R6 = [221; 255].$

Example…



A
B1 (R_1=[0,14])
B2 (R_2=[15,15])
B3 (R_3=[16,99])



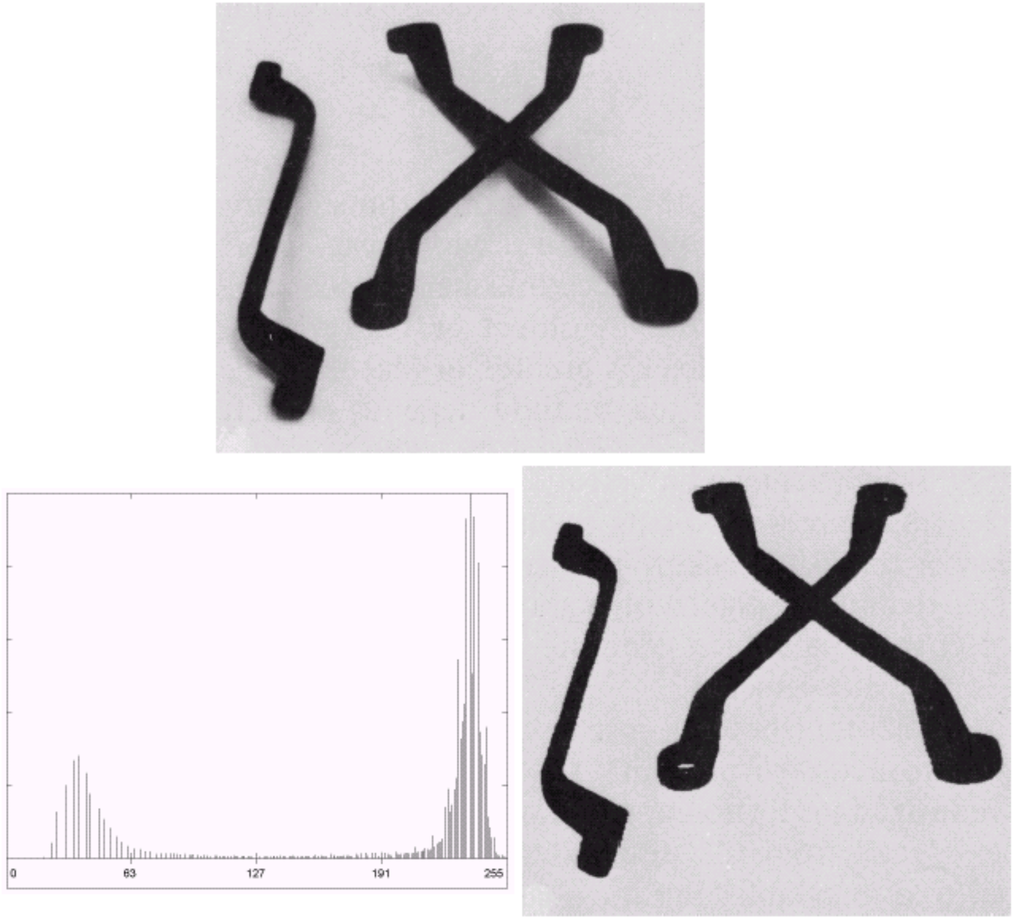
A
B4 (R_4=[100,149])
B5 (R_5=[150,220])
B6 (R_6=[221,255])

A
B (Histogram Segmented)

Selection of Thresholds

- Extensively researched questions
  - – No absolute "optimal" solution
- Histogram based approaches
  - – Interactive threshold
    - Visual inspection of the histogram
  - – Heuristic
  - – Minimisation methods
    - Otsu's method
    - Method by Dong et al
    - Global optimal
  - – Adaptive threshold

Basic Global Thresholding



a
b c

FIGURE 10.28 (a) Original image. (b) Image histogram. (c) Result of global thresholding with *T* midway between the maximum and minimum gray levels.

Selection of Global Threshold

- Heuristic method
  - select the initial estimate for T (mid value)
  - Group the pixels using T producing G_1 (>=T) and G_2.
  - compute average μ_1 and μ_2
  - $T = 1/2\ (\mu_1 + \mu_2)$
  - repeat until the difference in T is small

Example: Heuristic method

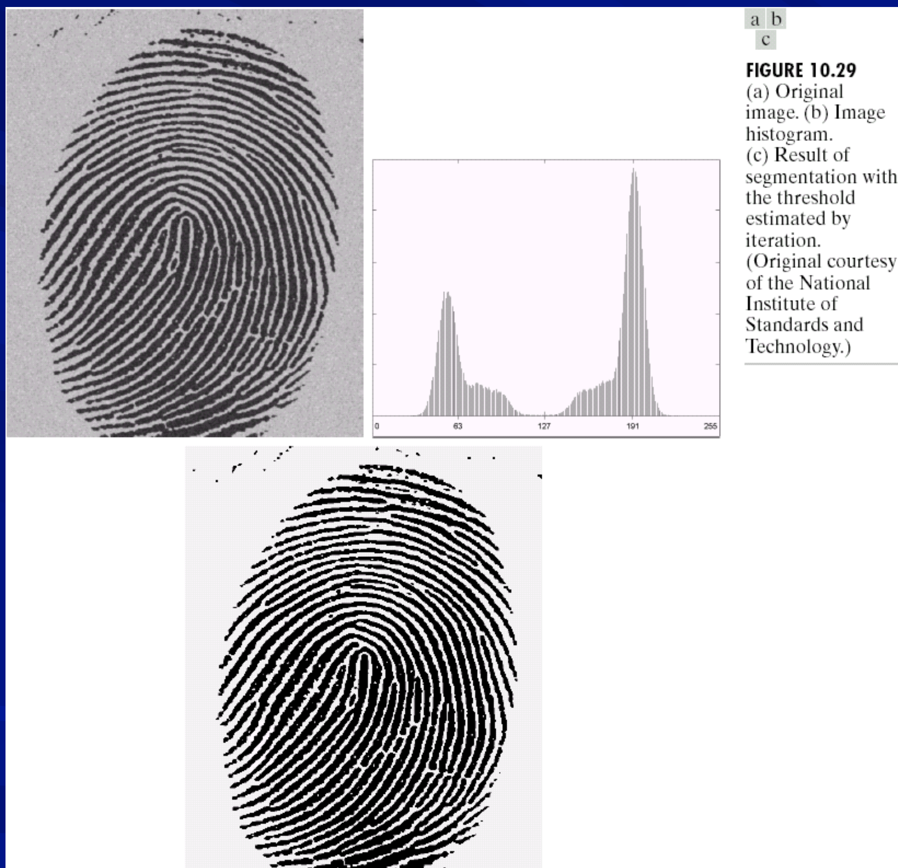


FIGURE 10.29 (a) Original image. (b) Image histogram. (c) Result of segmentation with the threshold estimated by iteration. (Original courtesy of the National Institute of Standards and Technology.)

Note: the clear valley of the histogram and the effective of the segmentation between object and background

3 iterations
with result T = 125

Otsu's Thresholding Method

(1979)

- Based on a very simple idea: Find the threshold that *minimizes the weighted within-class variance.*
- This turns out to be the same as *maximizing the between-class variance.*
- Operates directly on the gray level histogram [*e.g.* 256 numbers, P(i)], so it's fast (once the histogram is computed).

Otsu: Assumptions

- Histogram (and the image) are *bimodal.*
- No use of *spatial coherence,* nor any other notion of object structure.
- Assumes stationary statistics, but can be modified to be locally adaptive. (exercises)
- Assumes uniform illumination (implicitly), so the bimodal brightness behavior arises from object appearance differences only.

The *weighted within-class variance* is:

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

Where the class probabilities are estimated as:

$$q_1(t) = \sum_{i=1}^{t} P(i) \qquad q_2(t) = \sum_{i=t+1}^{I} P(i)$$

And the class means are given by:

$$\mu_1(t) = \sum_{i=1}^{t} \frac{iP(i)}{q_1(t)} \qquad \mu_2(t) = \sum_{i=t+1}^{I} \frac{iP(i)}{q_2(t)}$$

The individual class variances are:

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

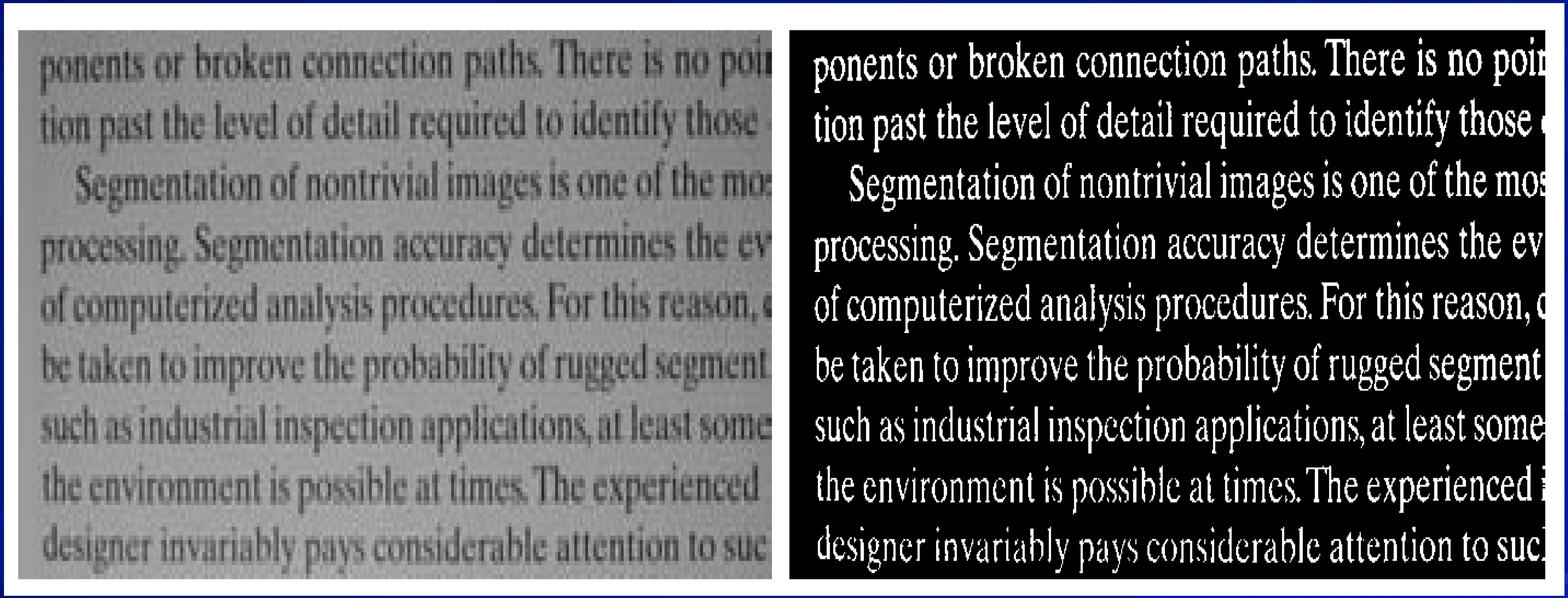
$$\sigma_2^2(t) = \sum_{i=t+1}^{I} [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)}$$

Now, we could actually stop here. All we need to do is just run through the full range of *t* values [1,256] and pick the value that minimizes $\sigma_w^2(t)$

But the relationship between the within-class and between-class variances can be exploited to generate a recursion relation that permits a much faster calculation.

Otsu's method…

- Text segmentation using Otsu’s method



Extension (Dong et al.)

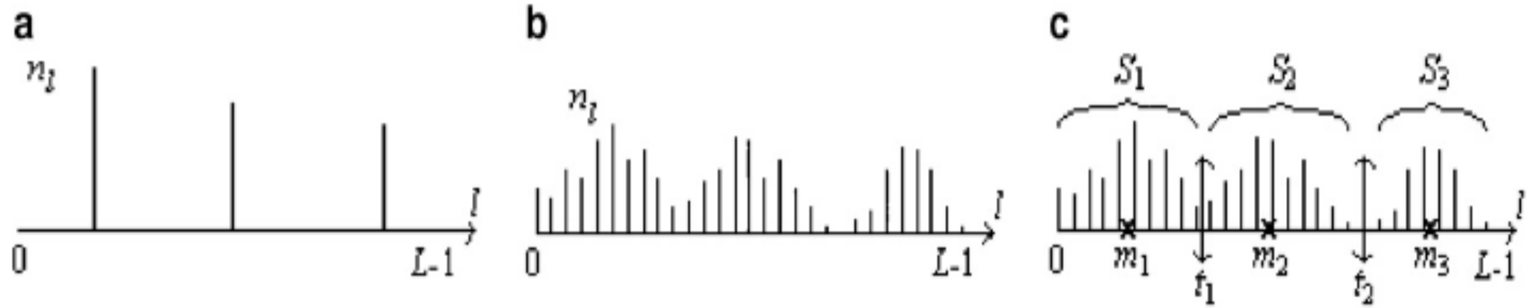
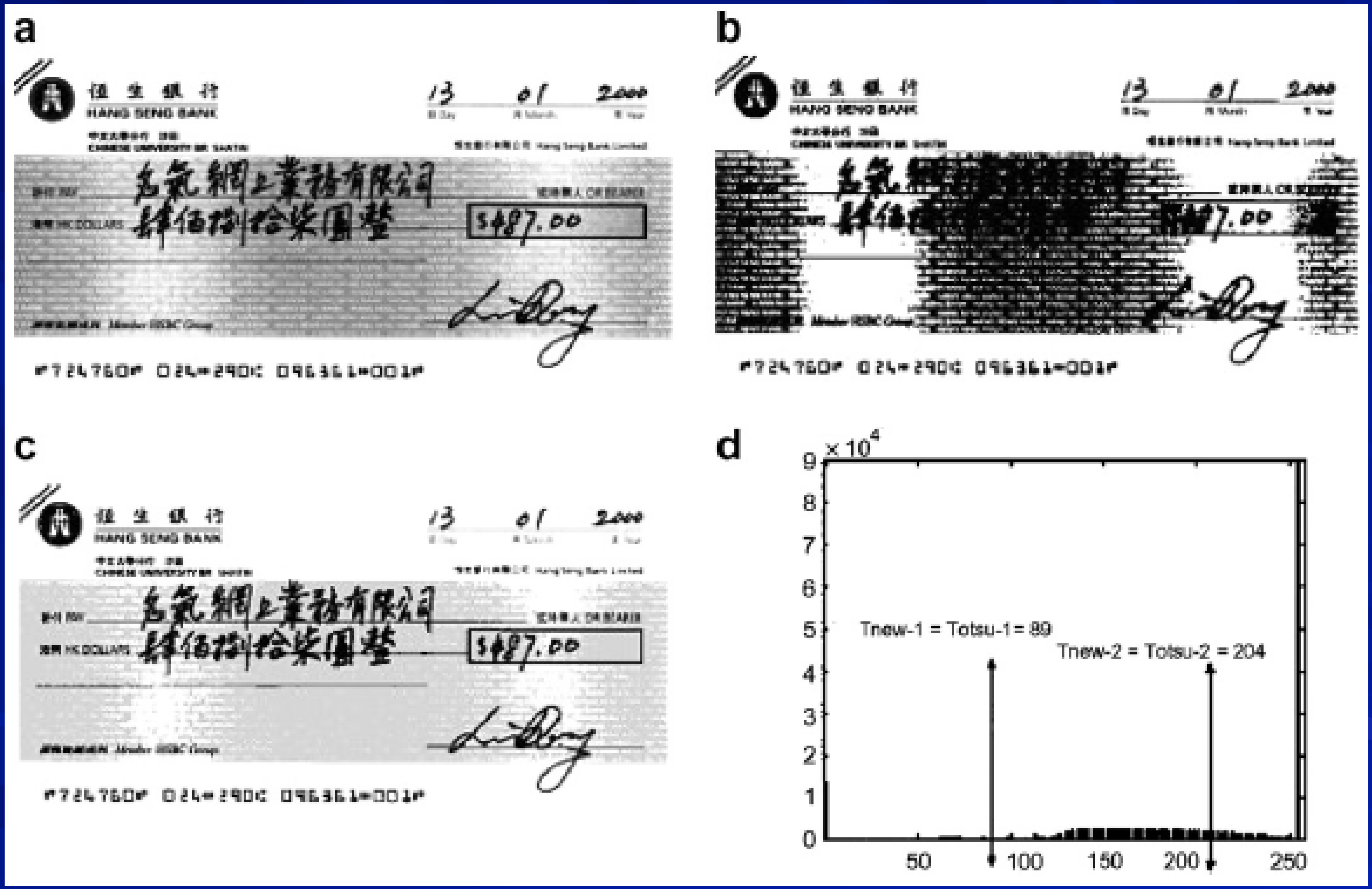


Fig. 1. (a) An ideal histogram with only three non-zero gray levels. (b) A practical histogram with three clusters. (c) One partition of three clusters of the gray levels, S_1, S_2, S_3, where the centroids of the clusters are m_1, m_2, and m_3, and the two thresholds are t_1 and t_2.

$$f(m_1, m_2, \ldots, m_c) = \sum_{i=1}^{c} f_i = \sum_{i=1}^{c} \sum_{l \in S_i} n_l (l - m_i)^2.$$

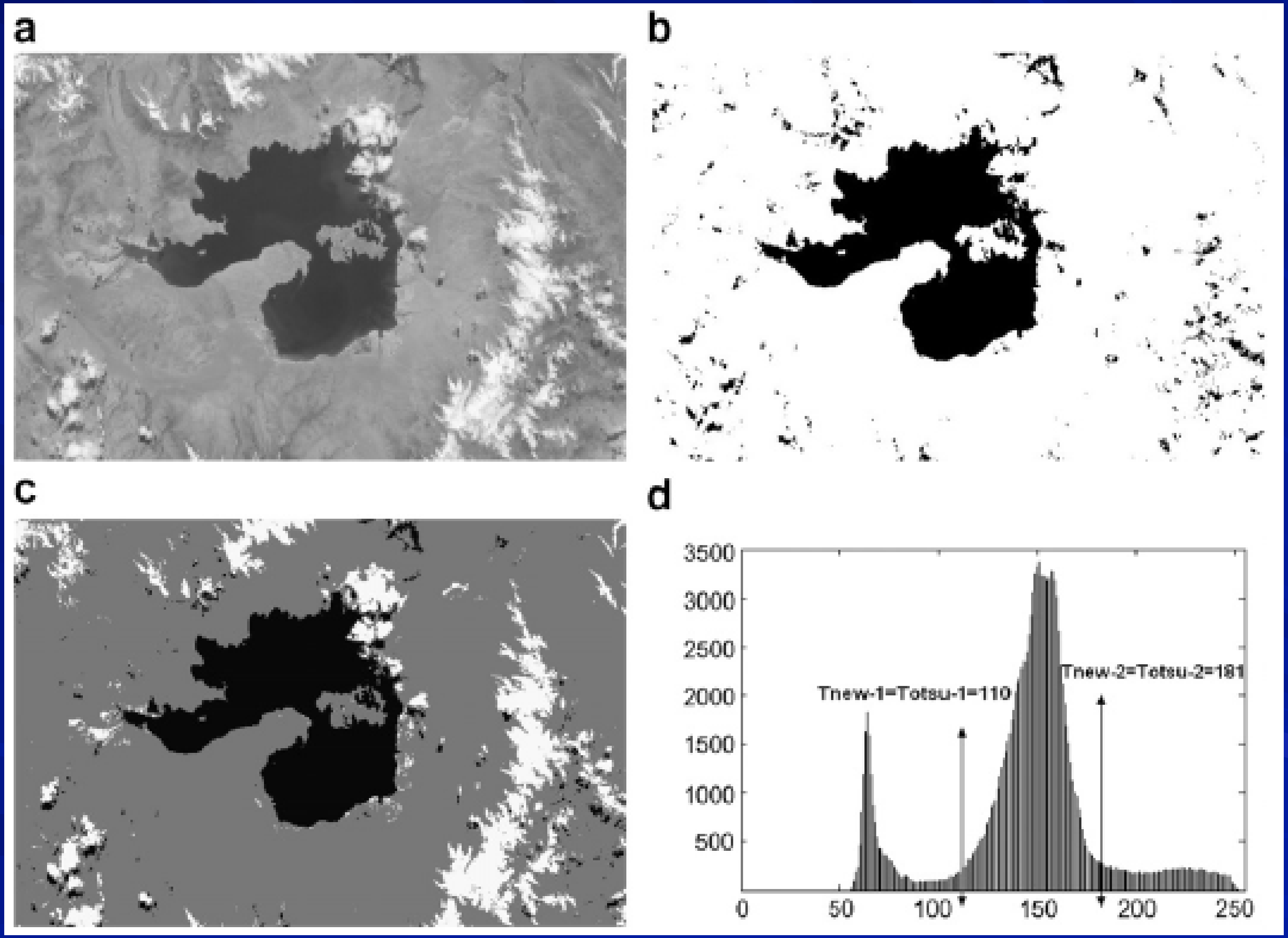
L. Dong, G. Yu, P. Ogunbona and W. Li, *An efficient iterative algorithm for image thresholding*, Pattern Recognition Letter, 29 (2008) 1311–1316

Examples

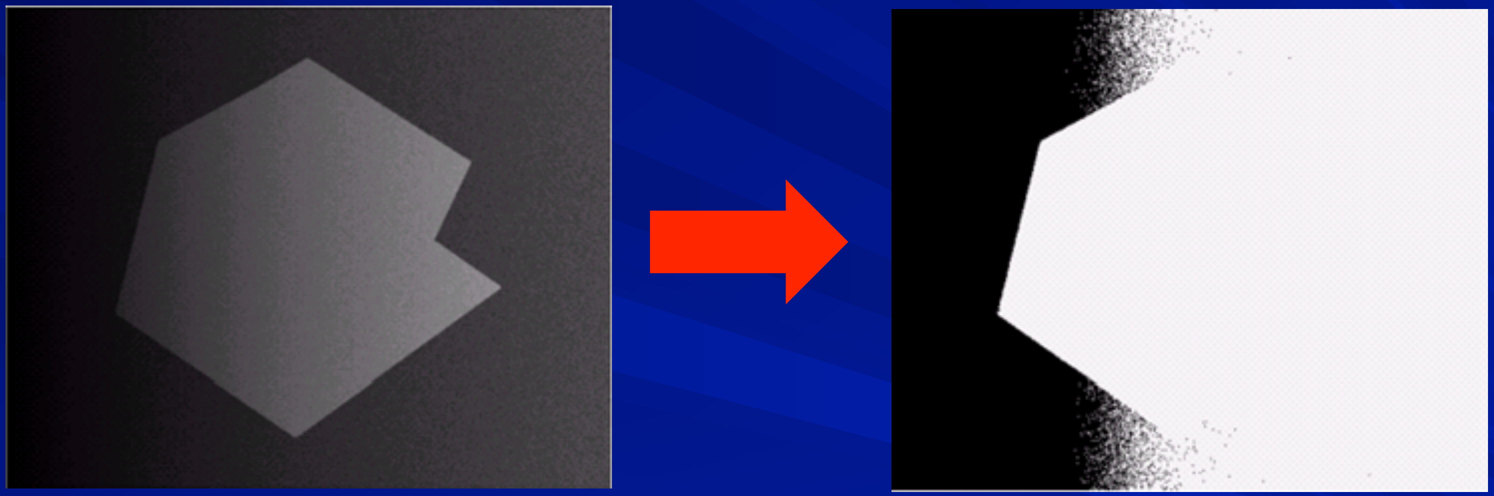


a
b
c
d
HANG SENG BANK
13 01 2000
$487.00
× 10^4
9
8
7
6
5
4
3
2
1
0
Tnew-1 = Totsu-1 = 89
Tnew-2 = Totsu-2 = 204
50
100
150
200
250

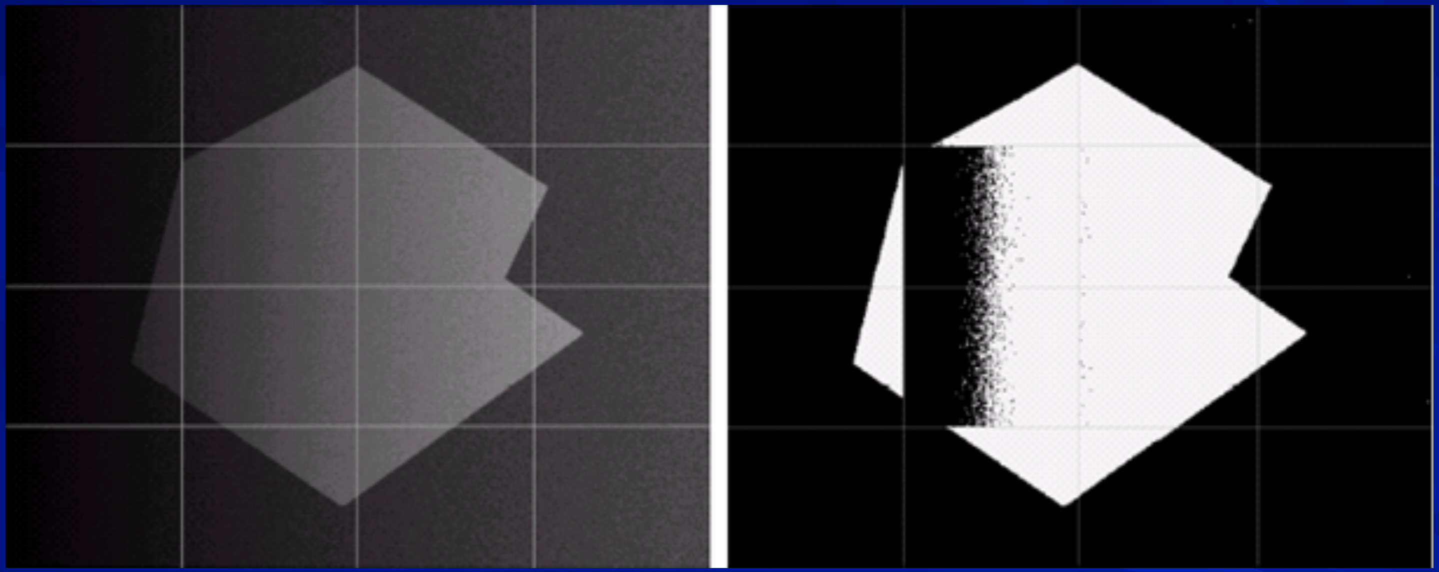
Examples…


a
b
c
d
3500
3000
2500
2000
1500
1000
500
0
50
100
150
200
250
Tnew-1=Totsu-1=110
Tnew-2=Totsu-2=181

Basic Adaptive Thresholding

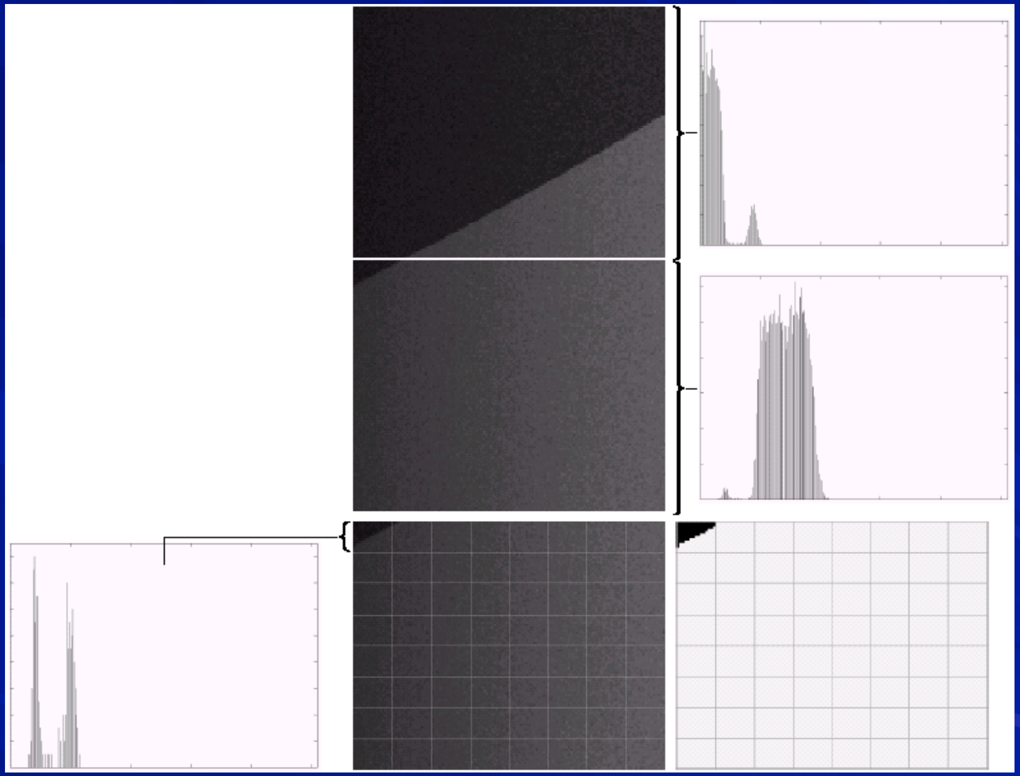


Basic Adaptive Thresholding



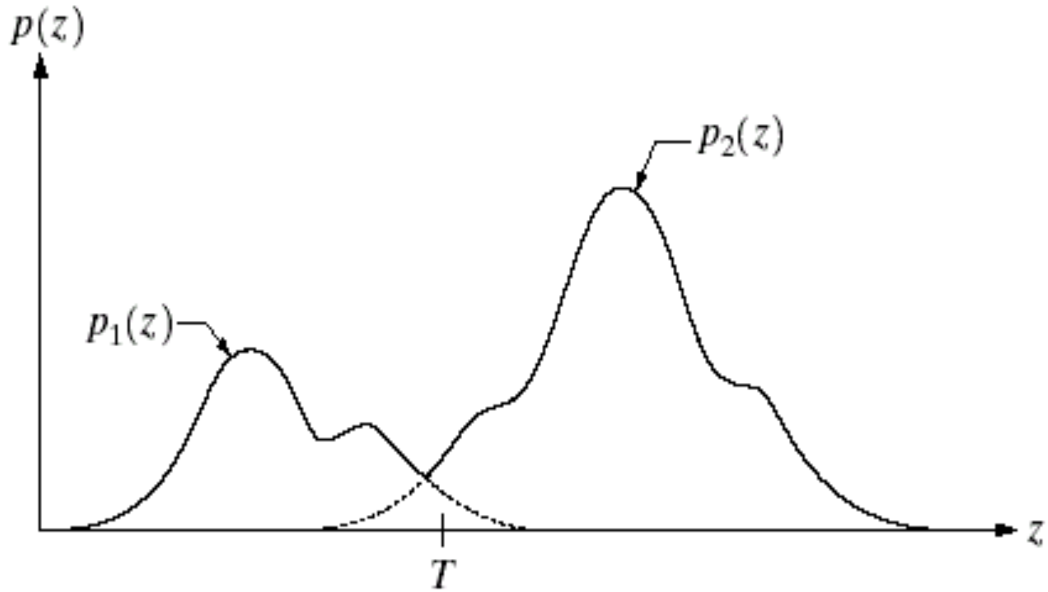
Further subdivision

a). Properly and improperly segmented subimages from previous example
b)-c). corresponding histograms
d). further subdivision of the improperly segmented subimage.
e). histogram of small subimage at top
f). result of adaptively segmenting d).



Optimal Global and Adaptive Thresholding

FIGURE 10.32 Gray-level probability density functions of two regions in an image.



$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

$$P_1 + P_2 = 1$$

Probability of erroneous classification

$$E_1(T) = \int_{-\infty}^{T} p_2(z)\,dz$$

$$E_2(T) = \int_{T}^{\infty} p_1(z)\,dz$$

$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$

Minimum error

Differentiating E(T) with respect to T (using Leibniz's rule) and equating the result to 0

$$\frac{dE(T)}{dT} = \frac{d(P_2 E_1(T) + P_1 E_2(T))}{dT} = 0$$

find T which makes

if $P_1 = P_2$ then the optimum threshold is where the curve $p_1(z)$ and $p_2(z)$ intersect

$$P_1 p_1(T) = P_2 p_2(T)$$

Gaussian density

Example: use PDF = Gaussian density : $p_1(z)$ and $p_2(z)$

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$
$$= \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

where

- μ_1 and σ_1^2 are the mean and variance of the Gaussian density of one object
- μ_2 and σ_2^2 are the mean and variance of the Gaussian density of the other object

Quadratic equation

$$AT^2 + BT + C = 0$$

$$where \quad A = \sigma_1^2 - \sigma_2^2$$

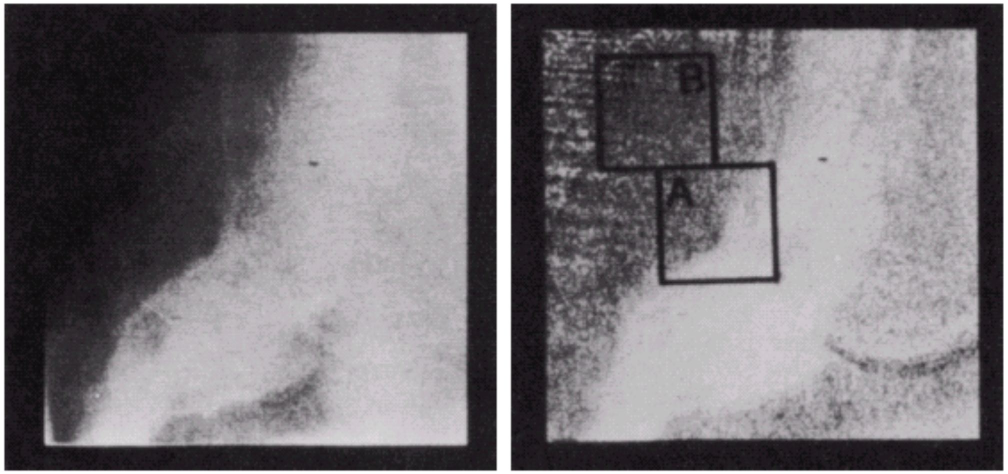
$$B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)$$

$$C = \sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2 2\sigma_2^2 \ln(\sigma_2 P_1 / \sigma_1 P_2)$$

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2}\ln\left(\frac{P_2}{P_1}\right)$$

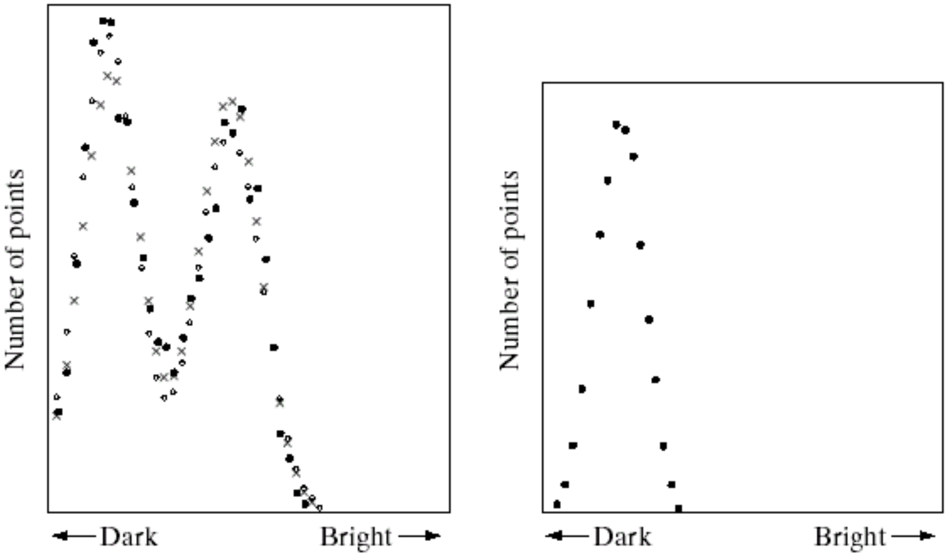
if $P_1 = P_2$ or $\sigma = 0$ then the optimal threshold is the average of the means

Example



a b

FIGURE 10.33 A cardioangiogram before and after preprocessing. (Chow and Kaneko.)

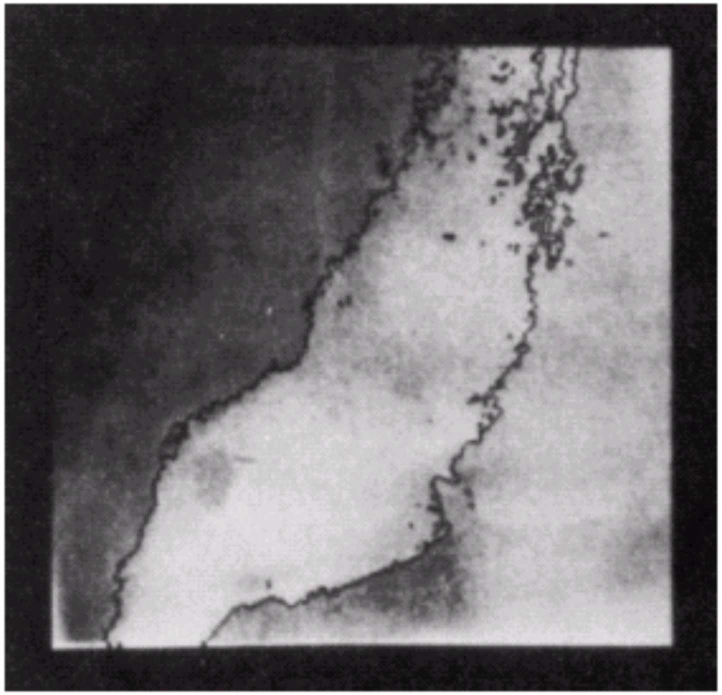


a b

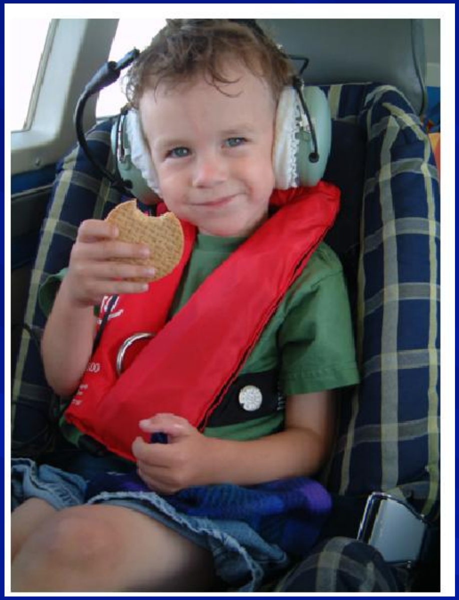
FIGURE 10.34 Histograms (black dots) of (a) region *A*, and (b) region *B* in Fig. 10.33(b). (Chow and Kaneko.)

Example: Boundary superimposed

FIGURE 10.35 Cardioangiogram showing superimposed boundaries. (Chow and Kaneko.)



Thresholding for Color Images?



K-means

- Notations
  - Samples (pixels)

$$\mathbf{x}_1 = (x_{11}, x_{12}, \cdots, x_{1d}), \mathbf{x}_2 = (\cdots), \ldots, \mathbf{x}_n = (x_{n1}, x_{n2}, \cdots x_{nd})$$

  - Prototypes

$$\mu_1, \mu_2, \ldots, \mu_K$$

  - L_2-distance

$$\| \mathbf{x}_i - \mu_j \|^2 = \sum_{k=1}^{d} (x_{ik} - \mu_{jk})^2$$

  - membership

$$r_{ik} \in \{0,1\},$$

K-means

- Objective function

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

- Two iterative phases
  - Expectation

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

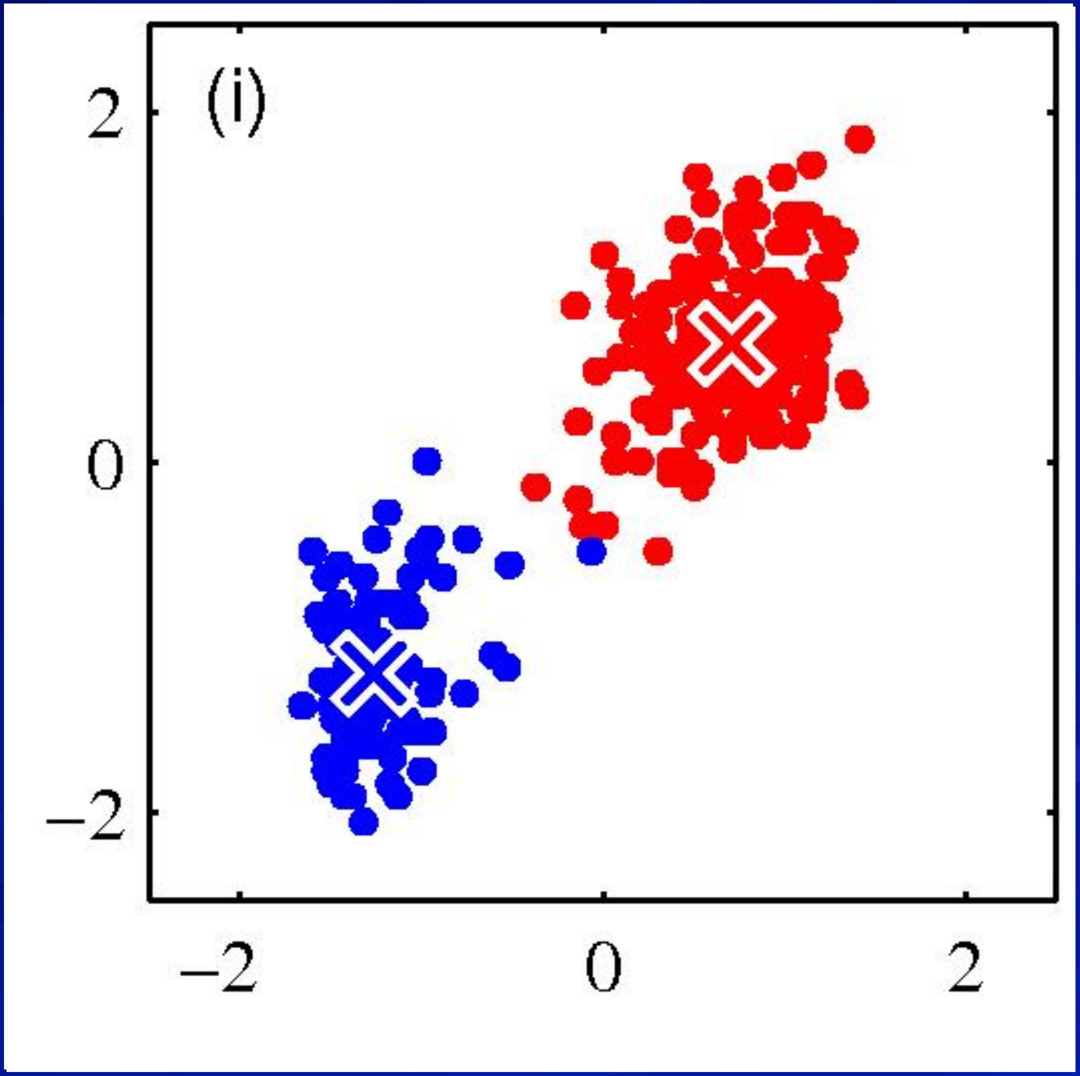
  - Maximization

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}.$$

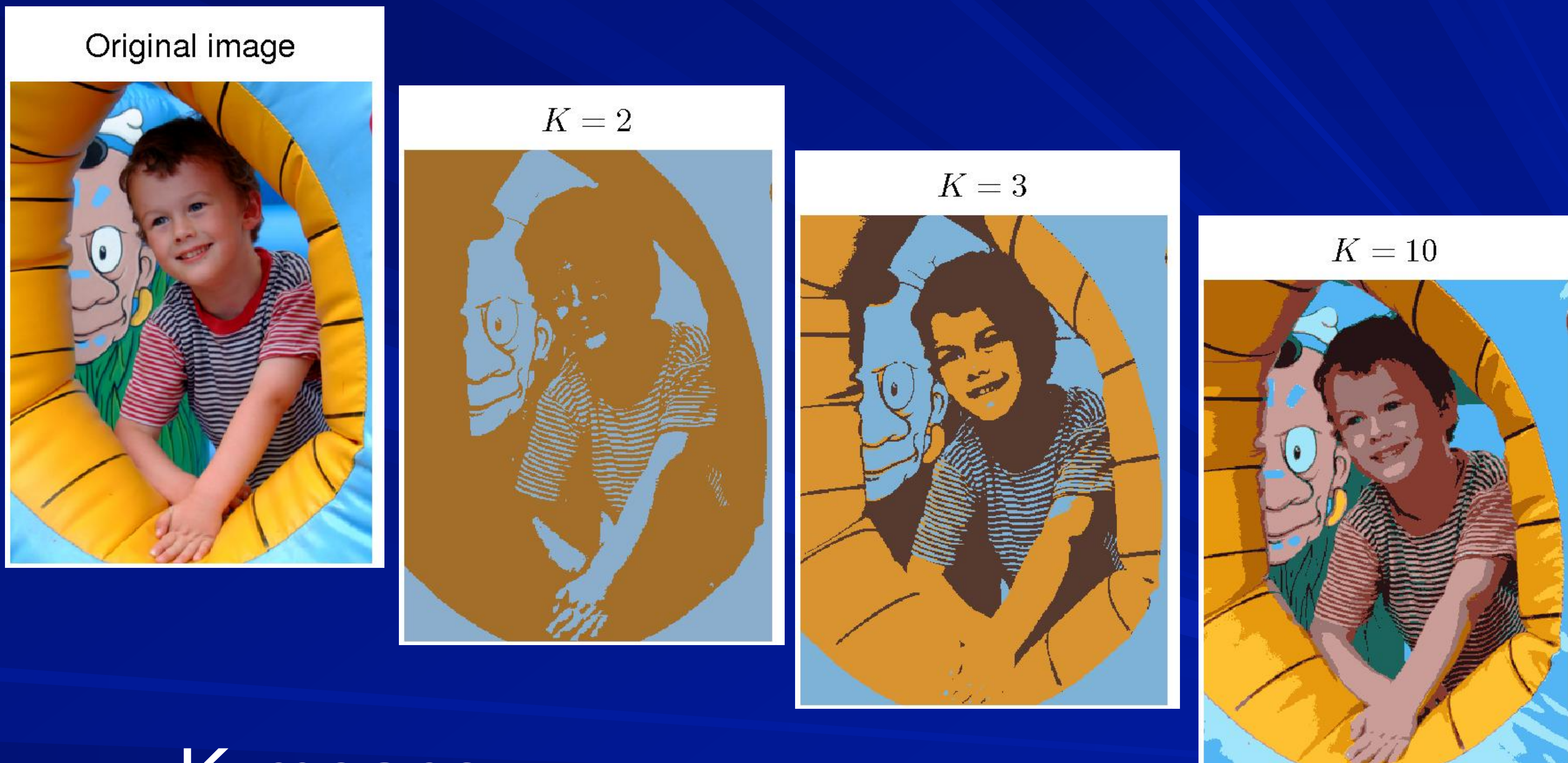
K-means algorithm

- Choose some initial values for the prototypes (means)
- (E-step) Minimize the objective function with respect to the sample memberships
  - Re-assigning samples to clusters
- (M-step) Minimize the objective function with respect to the K prototypes
  - Re-computing the cluster means
- Repeat the EM steps until
  - there is no further change in the assignment, or
  - some maximum number of iterations is exceeded

Illustration of K-means



Color Image Segmentation



K-means

Color Image Segmentation

Color Image Segmentation

Fuzzy C-Means (FCM)

- Membership matrix: $\mathbf{U}_{c \times n}$
  - U_{ij} is the grade of membership of sample *j* with respect to prototype *i*
- *Crisp* membership:

$$u_{ij} = 1, \quad \text{if } \| \mathbf{p}_i - \mathbf{x}_j \|^2 = \min_k \| \mathbf{p}_k - \mathbf{x}_j \|^2$$
$$u_{ij} = 0, \quad \text{otherwise}$$

- *Fuzzy* membership:

$$\sum_{i=1}^{c} u_{ij} = 1, \forall j = 1, \ldots, n$$

Fuzzy *c*-means (FCM)

- The objective function of FCM is

$$J = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m} d_{ij}^{2} = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m} \| \mathbf{p}_i - \mathbf{x}_j \|^2$$

FCM…

- Introducing the Lagrange multiplier λ with respect to the constraint

$$\sum_{i=1}^{c} u_{ij} = 1,$$

we rewrite the objective function as:

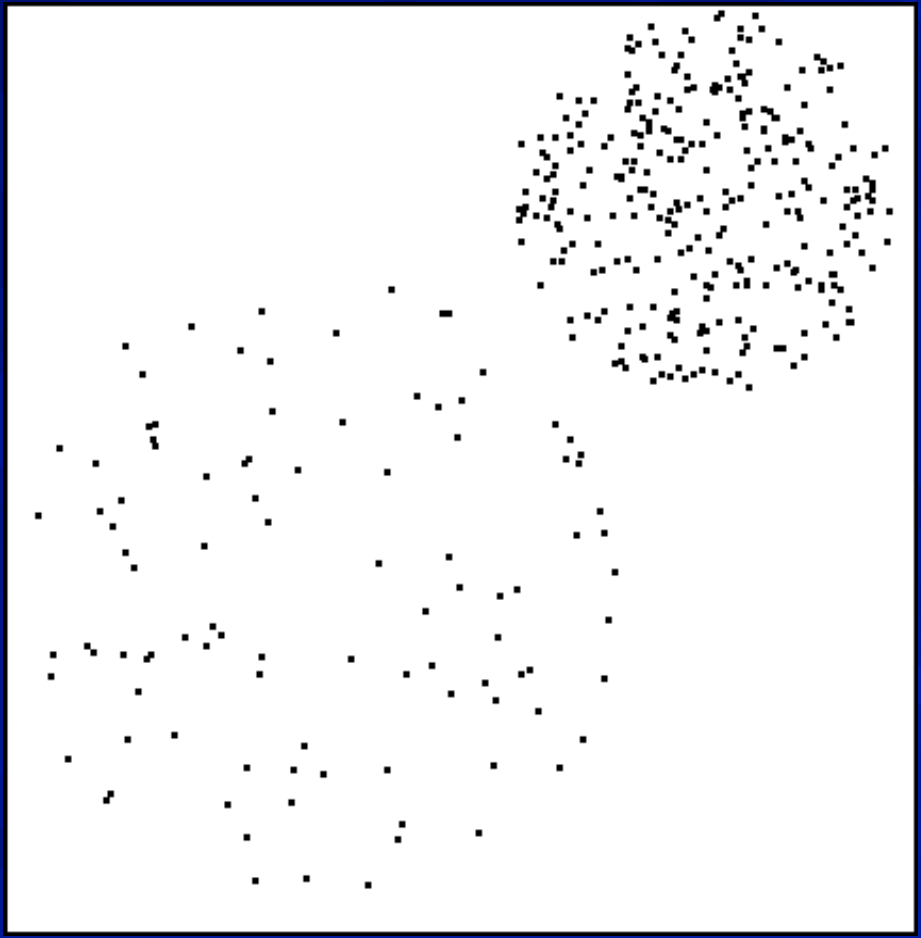
$$J = \sum_{i=1}^{c} u_{ij}^{m} d_{ij}^{2} - \lambda\left(\left(\sum_{i=1}^{c} u_{ij}\right) - 1\right)$$

FCM…

- EM updating rules

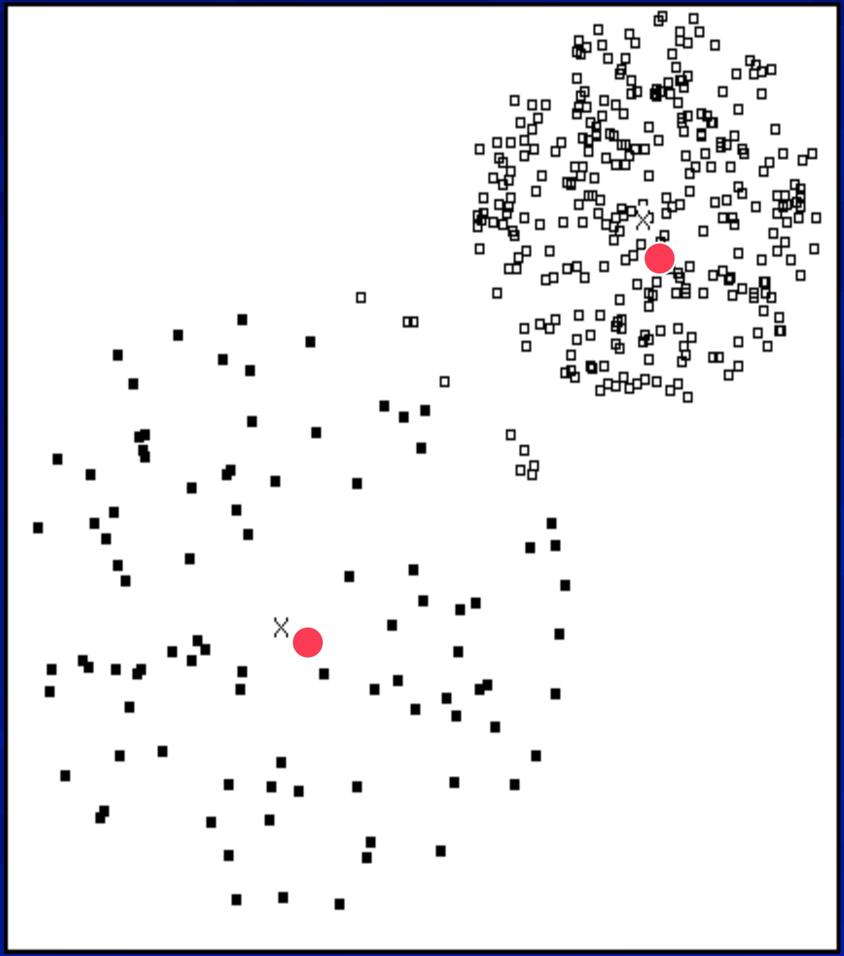
$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{kj}}\right)^{1/(m-1)}} \qquad \mathbf{p}_i = \frac{\sum_{j=1}^{n} u_{ij}^m \mathbf{x}_j}{\sum_{j=1}^{n} u_{ij}^m}$$

K-means vs. Fuzzy *c*-means

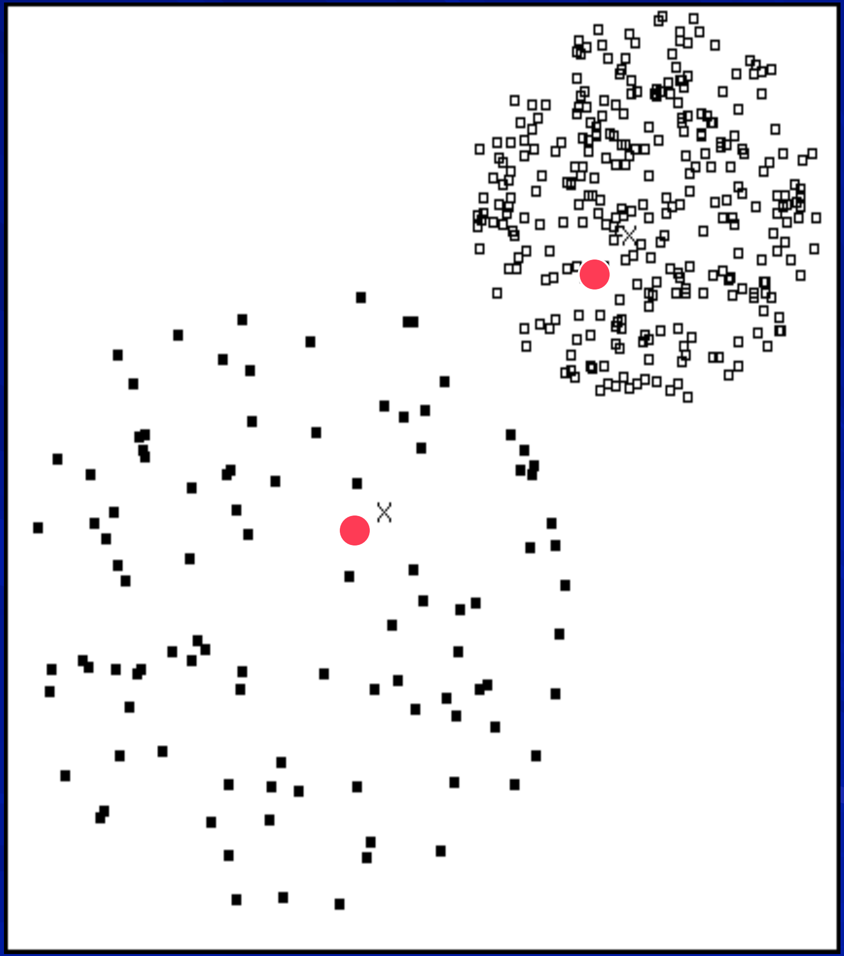


Sample Points

K-means vs. Fuzzy *c*-means



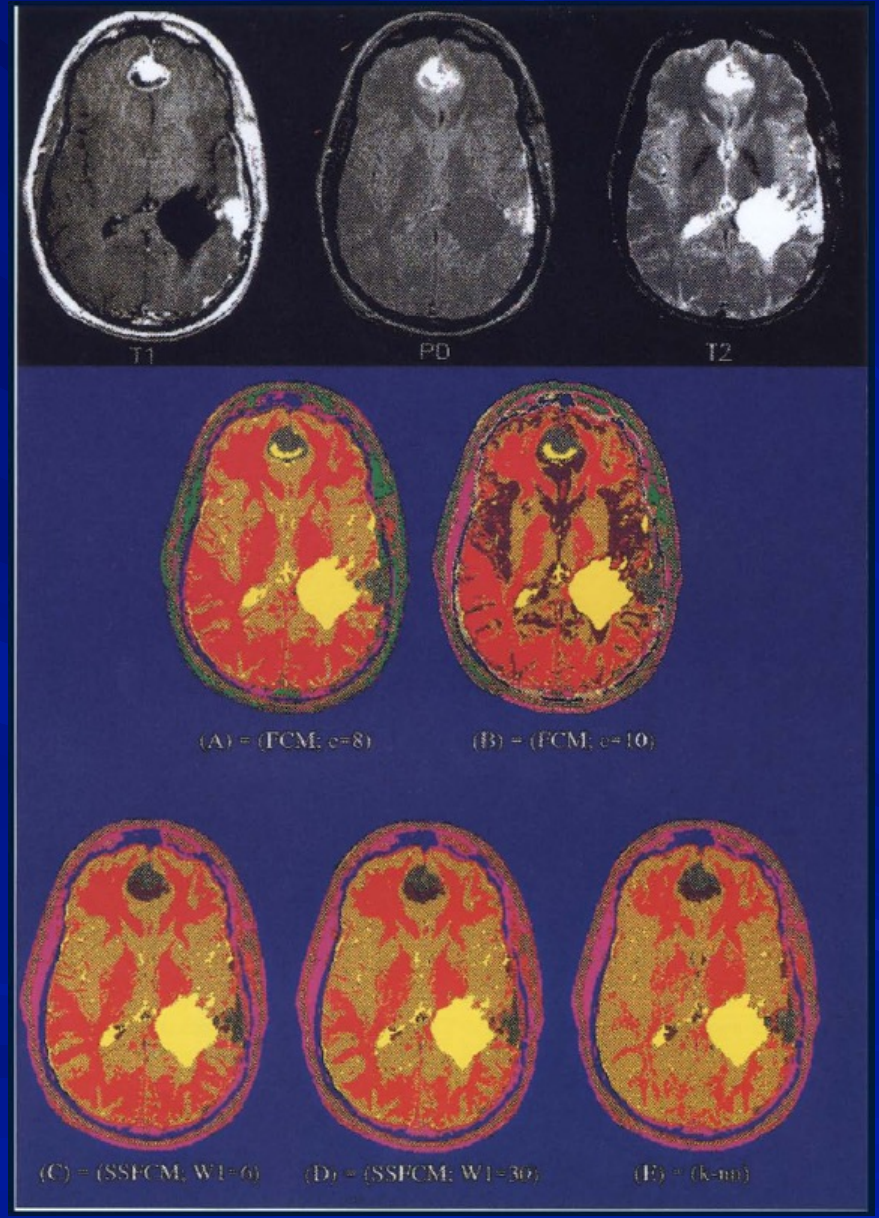
K-means



Fuzzy c-means

FCM segmentation examples

FCM



Post Processing

Post-Processing

- Binary Morphological Filters
- Majority voting

1	1	1	1	1	2	2
3	1	1	1	1	2	2
3	3	1	1	2	2	2
3	3	3	x	2	2	2
3	1	2	2	2	1	2
3	3	3	2	2	2	2
3	3	3	2	2	3	2

Region-Based Method

- Consider *simultaneously*
  - the similarity of visual properties (features), and
  - spatial connectivity

Region-Based Method: Region Growing

- **Region Growing**
  - Start from a seed, and let it grow (include similar neighborhood)

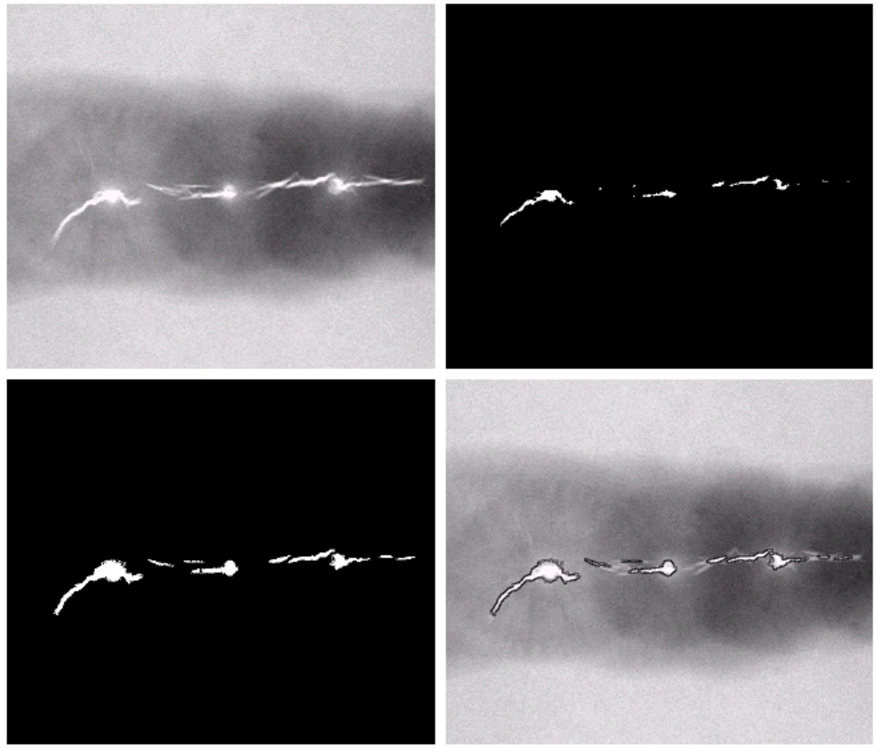


FIGURE 10.40 (a) Image showing defective welds. (b) Seed points. (c) Result of region growing. (d) Boundaries of segmented defective welds (in black). (Original image courtesy of X-TEK Systems, Ltd.).

Criteria:

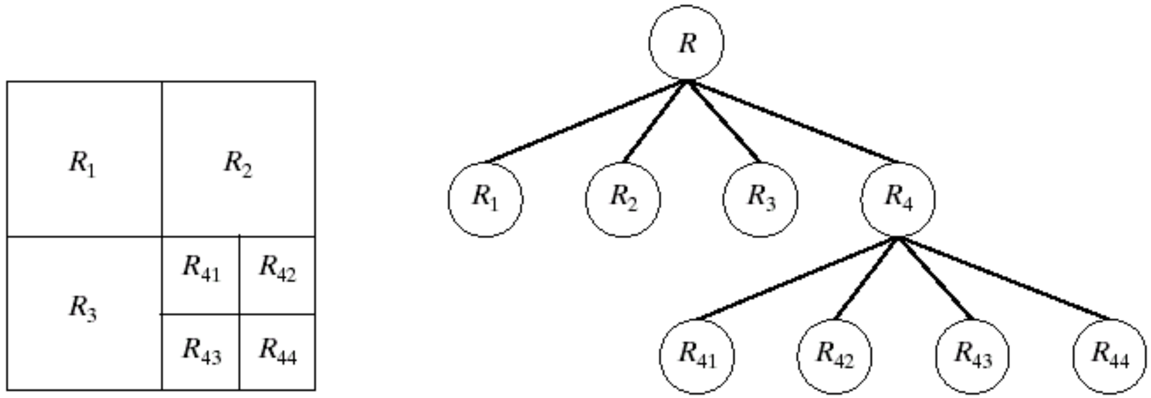
1. the absolute gray-level difference between any pixel and the seed has to be less than 65
2. the pixel has to be 8-connected to at least one pixel in that region (if more, the regions are merged)

Key: similarity measure

From [Gonzalez & Woods]

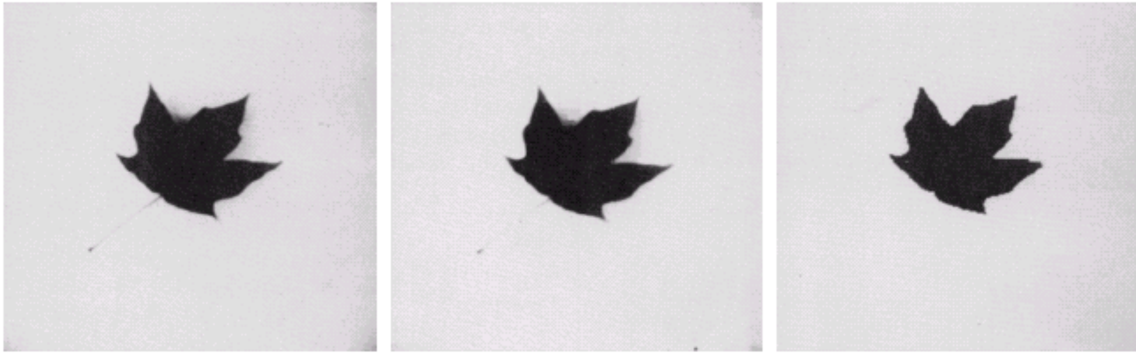
Region-Based Method: Split and Merge

- **Split and Merge**
  - Iteratively split (non-similar region) and merge (similar regions)
  - Example: Quadtree approach



a b

FIGURE 10.42 (a) Partitioned image. (b) Corresponding quadtree.



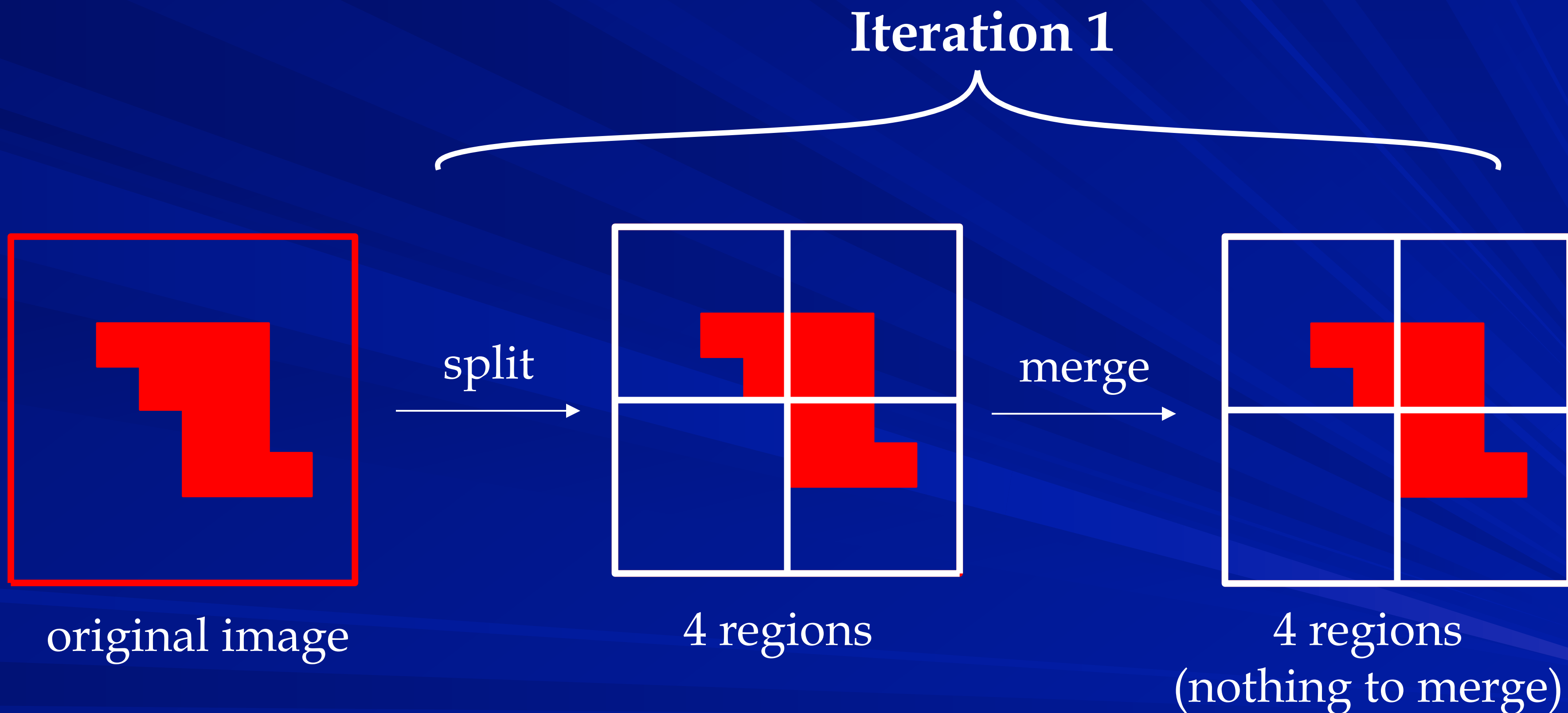
a b c

FIGURE 10.43 (a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).

From [Gonzalez & Woods]

Region-Based Method: Split and Merge

- **Example: Quadtree Split and Merge Procedure**

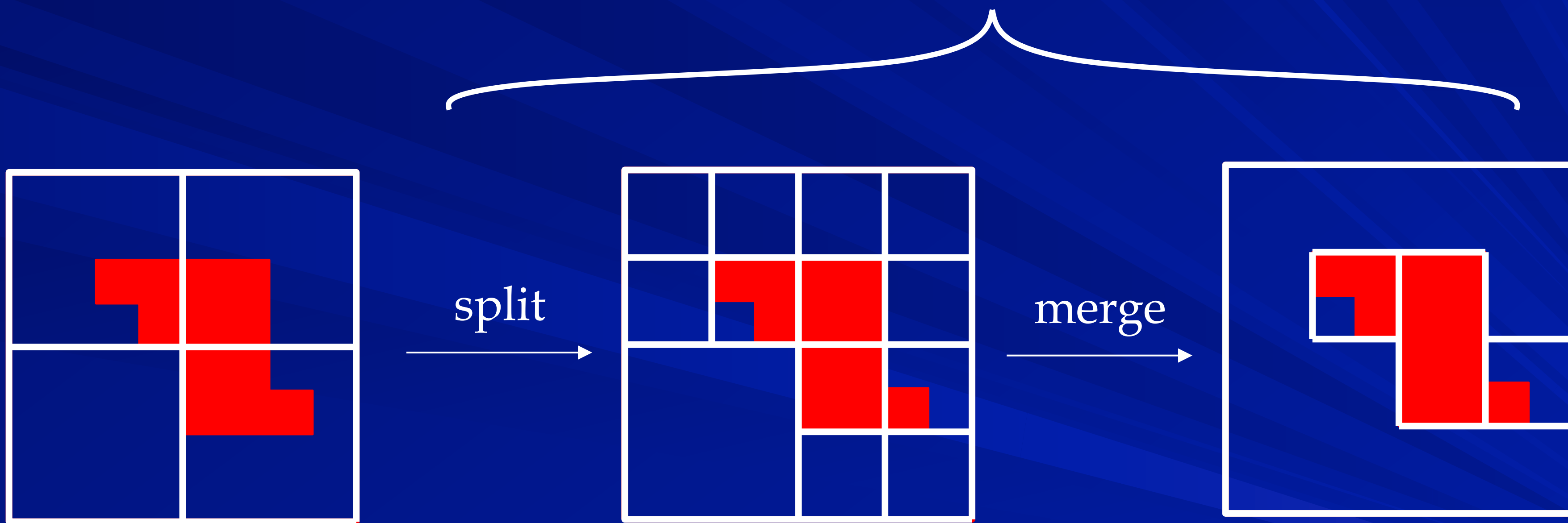


Split Step → split every non-uniform region to 4
Merge Step → merge all uniform adjacent regions

Region-Based Method: Split and Merge

- **Example: Quadtree Split and Merge Procedure**

Iteration 2



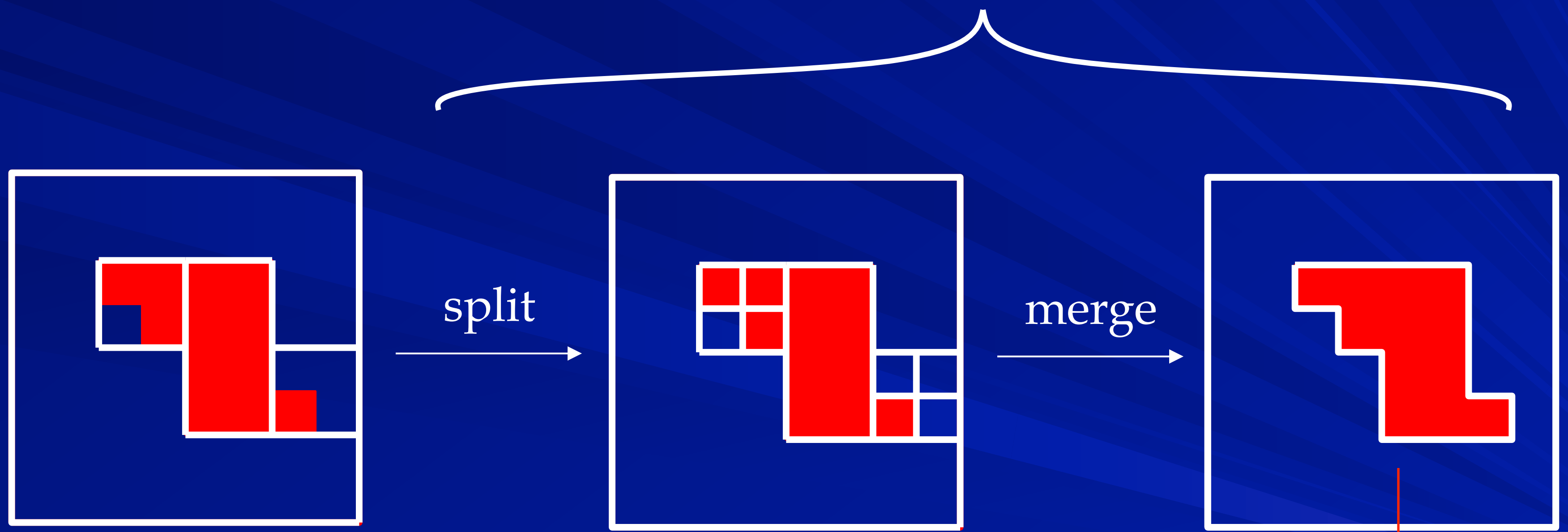
from Iteration 1 | 13 regions | 4 regions

Split Step → split every non-uniform region to 4
Merge Step → merge all uniform adjacent regions

Region-Based Method: Split and Merge

- **Example: Quadtree Split and Merge Procedure**

Iteration 3



from Iteration 2 | 10 regions | 2 regions

final segmentation result

Split Step → split every non-uniform region to 4
Merge Step → merge all uniform adjacent regions

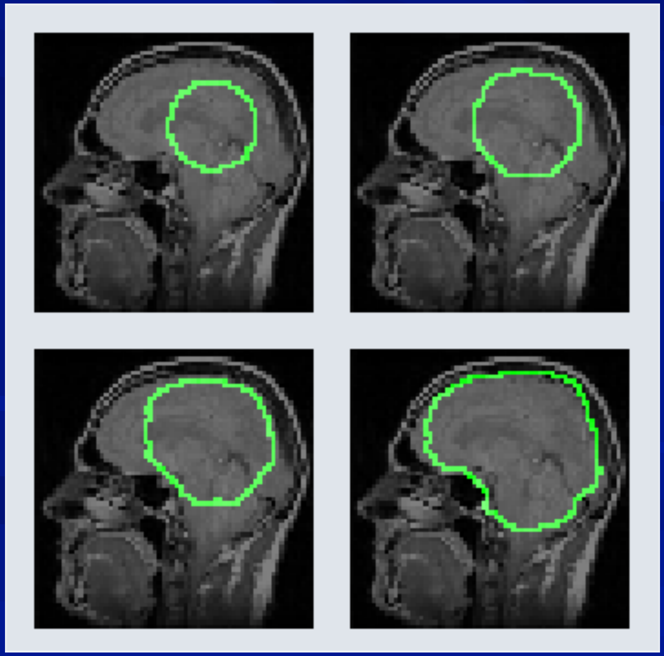
Contour-based
Image Segmentation

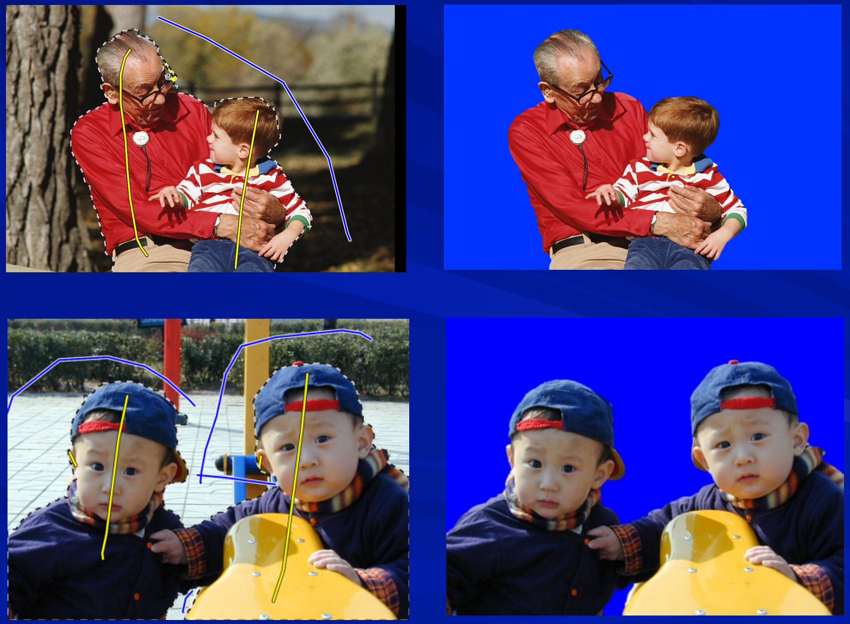
Contour-Based Method

image → edge detection → boundary detection → classification and labeling → segmentation

Boundary-Based Method

- **Advanced Method: Active Contour (Snake) Model**
  - Iteratively update contour (region boundary)
  - Partial differential equation (PDE) based optimization
  - Lazy Snapping (MSRA)





Applications of Image Segmentation

Segmentation

- Basic Formulation

$$(a)\quad \bigcup_{i=1}^{n} R_i = R$$

$$(b)\quad R_i \text{ is a connected region}, i = 1, 2, \ldots, n$$

$$(c)\quad R_i \cap R_j = \phi \text{ for all i and j}, i \neq j$$

$$(d)\quad P(R_i) = TRUE \text{ for } i = 1, 2, \ldots, n$$

$$(e)\quad P(R_i \cup R_j) = FALSE \ \ for\ i \neq j$$

$P(R_i)$ is a logical predicate property defined over the points in set R_i

ex. $P(R_i)$ = TRUE if all pixel in R_i have the same gray level

Applications of segmentation

- **Spark Plug Gap Measurement**

Applications of segmentation …

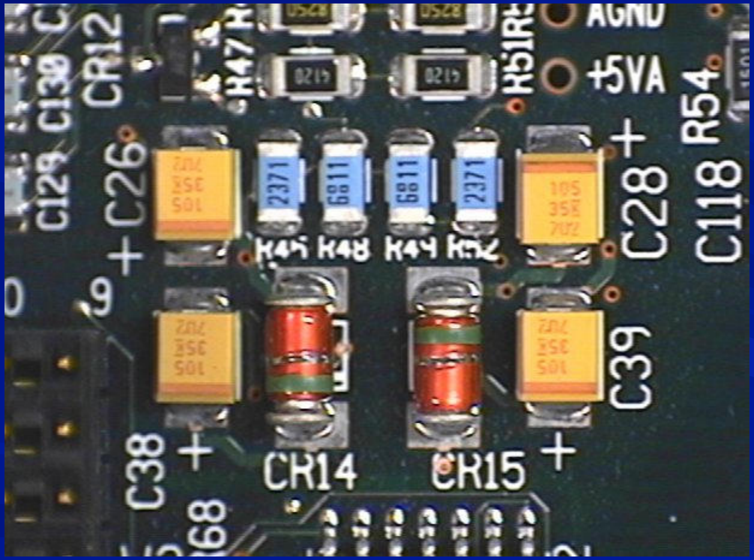
- **Blister Pack Inspection:** Ensuring that blister packs contain the correct number and type of pills before they reach pharmacies, ensuring the integrity of the product and increase the yield of production by automating the inspection of blister pack contents.




- Acquire colour images of the blister packs. Use colour location to count the number of green areas in the image. With colour location, you create a model or template that represents the colours that you are searching. Then the machine vision application searches for the model in each acquired image and calculates a score for each match. The surface area of each pill in the pack must be at least 50% green to pass inspection.

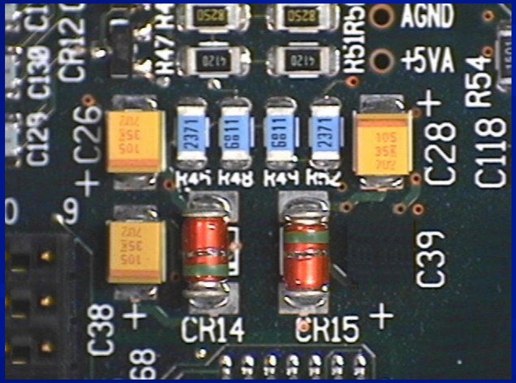
Applications of segmentation …

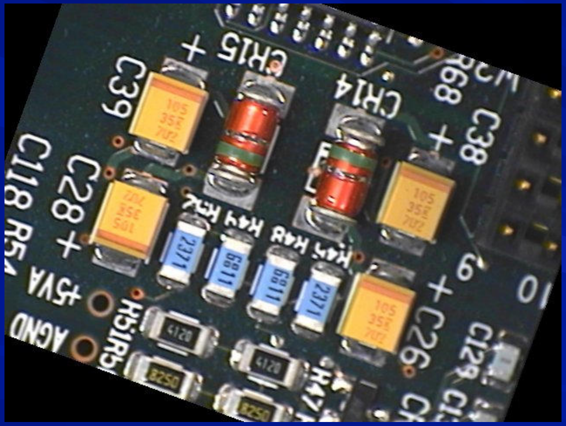
- **PCB Inspection**
- To ensure that components are present and at the correct orientation on a PCB.



Applications of segmentation …

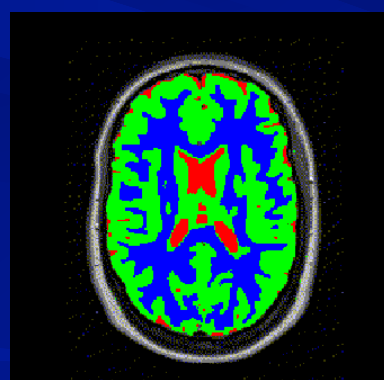
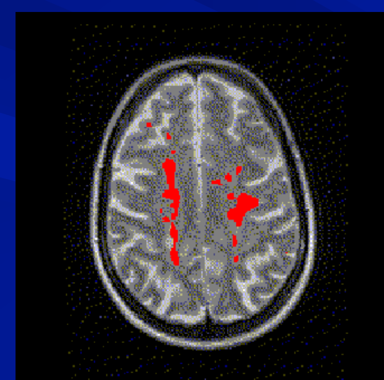
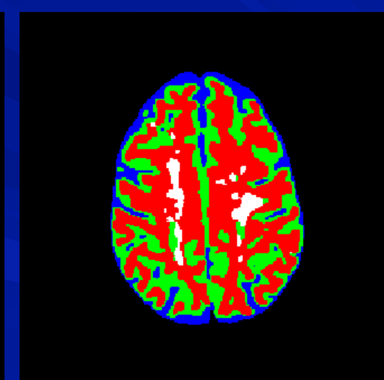
- **PCB Inspection (cont'd):** Colour information simplifies a monochrome problem by improving contrast or separation of the components from the background. Colour pattern matching can distinguish objects from the background more efficiently than grayscale pattern matching.
- This example uses rotation-invariant pattern matching because it can detect the components regardless of their orientations. You can use the orientation information to determine the correct placement of orientation-sensitive components, such as capacitors or diodes.





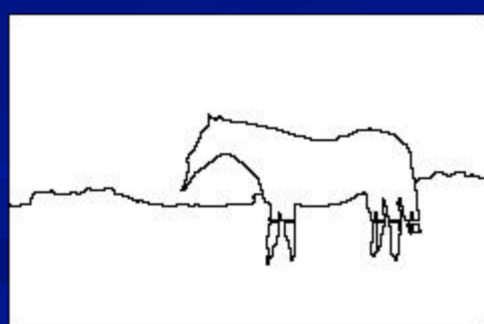
Applications of segmentation …

- **Computer assisted diagnosis**
  - Brain tissues
  - MS lesions

Applications of segmentation …

- **Photo annotation and retrieval**



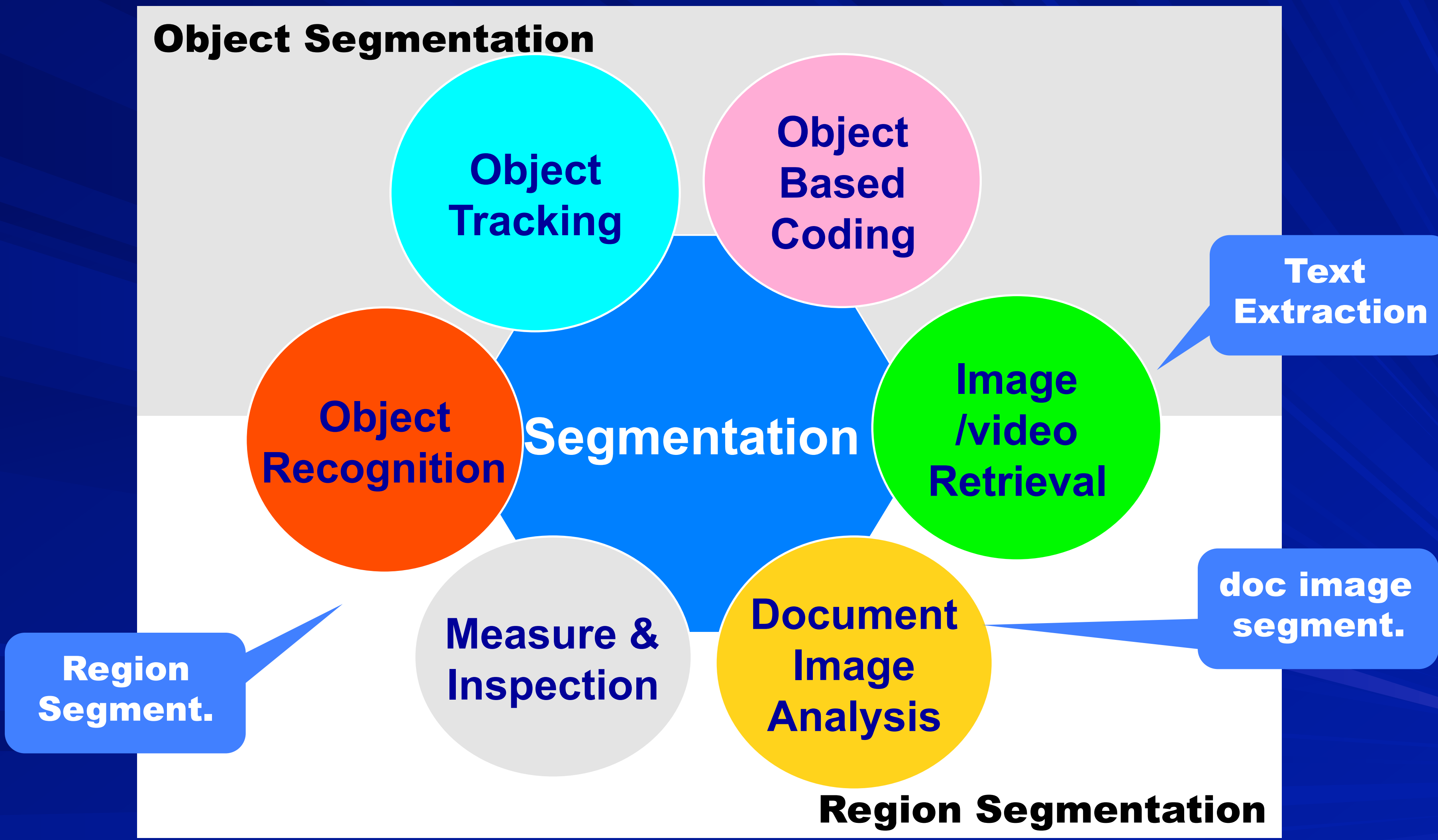
Applications of segmentation …

- **Video surveillance**
  - Moving objects

Applications of segmentation …

- **Recent applications**
  - Summarizing video
  - Finding machine parts
  - Finding people
  - Finding buildings or specific targets from satellite images
  - Searching a collection of images

Summary



Object Segmentation
Object Tracking
Object Based Coding
Text Extraction
Object Recognition
Segmentation
Image /video Retrieval
doc image segment.
Region Segment.
Measure & Inspection
Document Image Analysis
Region Segmentation

Key References

- D. A. Forsyth and J. Ponce, "Computer Vision: A Modern Approach", Prentice Hall, 2003 – Chapter 14
- R. C. Gonzalez and R. E. Woods, "Digital Image Processing", 2nd Ed. Prentice Hall, 2002
- N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", IEEE Trans. SMC, SMC-9(1), 1979, pp.62-66

CV & IP Tools

- Intel open source computer vision library OpenCV
  - C Libraries for CV
  - Available at Sourceforge
    http://sourceforge.net/projects/opencvlibrary/
- CVIPtools: is a UNIX/Win32-based software package
  - Collection of C-libraries
  - GUI-based interactive IP
  - Available from its home page
    http://www.ee.siue.edu/CVIPtools/index.php

FCM (Cnt'd)

- Setting the partial derivatives to zero, we obtain

$$\frac{\partial J}{\partial \lambda} = \left(\sum_{k=1}^{c} u_{ik} \right) - 1 = 0$$

$$\frac{\partial J}{\partial u_{ij}} = m \cdot u_{ij}^{m-1} \cdot d_{ij}^{2} - \lambda = 0$$

FCM (Cnt'd)

- From the 2nd equation, we obtain

$$u_{ij} = \left(\frac{\lambda}{m \cdot d_{ij}^2} \right)^{\frac{1}{m-1}}$$

- From this fact and the 1st equation, we obtain

$$1 = \sum_{k=1}^{C} u_{ik} = \sum_{k=1}^{C} \left(\frac{\lambda}{m \cdot d_{ik}^2} \right)^{\frac{1}{m-1}}$$

$$= \left(\frac{\lambda}{m} \right)^{\frac{1}{m-1}} \sum_{k=1}^{C} \left(\frac{1}{d_{ik}^2} \right)^{\frac{1}{m-1}}$$

FCM (Cnt'd)

- Therefore,

$$\left(\frac{\lambda}{m}\right)^{\frac{1}{m-1}} = \frac{1}{\sum_{k=1}^{c}\left(\frac{1}{d_{ik}^2}\right)^{\frac{1}{m-1}}}$$

and

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{1}{d_{ik}^2}\right)^{\frac{1}{m-1}}} \cdot \left(\frac{1}{d_{ij}^2}\right)^{\frac{1}{m-1}}$$

$$= \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}^2}{d_{ik}^2}\right)^{\frac{1}{m-1}}}$$

FCM (Cnt'd)

- Together with the 2nd equation, we obtain the updating rule for u_{ij}

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{1}{d_{ik}^{2}}\right)^{\frac{1}{m-1}}} \cdot \left(\frac{1}{d_{ij}^{2}}\right)^{\frac{1}{m-1}}$$

$$= \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}^{2}}{d_{ik}^{2}}\right)^{\frac{1}{m-1}}}$$

FCM (Cnt'd)

- On the other hand, setting the derivative of *J* with respect to $\mathbf{p}_i$ to zero, we obtain

$$
\begin{aligned}
0 = \frac{\partial J}{\partial \mathbf{p}_i} &= \frac{\partial}{\partial \mathbf{p}_i} \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} \| \mathbf{p}_i - \mathbf{x}_j \|^2 \\
&= \sum_{j=1}^{n} u_{ij}^{m} \frac{\partial}{\partial \mathbf{p}_i} \| \mathbf{p}_i - \mathbf{x}_j \|^2 \\
&= \sum_{j=1}^{n} u_{ij}^{m} \frac{\partial}{\partial \mathbf{p}_i} (\mathbf{p}_i - \mathbf{x}_j)^T (\mathbf{p}_i - \mathbf{x}_j) \\
&= -2 \sum_{j=1}^{n} u_{ij}^{m} (\mathbf{p}_i - \mathbf{x}_j)
\end{aligned}
$$

FCM (Cnt'd)

- It follows that

$$\frac{\partial J}{\partial \mathbf{p}_i} = \sum_{j=1}^{n} u_{ij}^m (\mathbf{p}_i - \mathbf{x}_j) = 0$$

- Finally, we can obtain the update rule of c_i:

$$\mathbf{p}_i = \frac{\sum_{j=1}^{n} u_{ij}^m \mathbf{x}_j}{\sum_{j=1}^{n} u_{ij}^m}$$

Semi-supervised FCM (Bensaid et al.)

- Samples

$$X = \Big\{\underbrace{\mathbf{x}_1^1,\ldots,\mathbf{x}_{n_1}^1}_{\text{labeled 1}}, \underbrace{\mathbf{x}_1^2,\ldots,\mathbf{x}_{n_2}^2}_{\text{labeled 2}},\ldots,\underbrace{\mathbf{x}_1^c,\ldots,\mathbf{x}_{n_c}^c}_{\text{labeled c}} \,\Big|\, \underbrace{\mathbf{x}_1^u,\ldots,\mathbf{x}_{n_u}^u}_{\text{unlabeled}}\Big\} = X^d \cup X^u, \tag{12}$$

- Membership matrix

$$U = \left[\left[\underbrace{\mathbf{U}_{(1)}^1 \cdots \mathbf{U}_{(n_1)}^1}_{\mathbf{e}_1} \underbrace{\mathbf{U}_{(1)}^2 \cdots \mathbf{U}_{(n_2)}^2}_{\mathbf{e}_2} \cdots \underbrace{\mathbf{U}_{(1)}^c \cdots \mathbf{U}_{(n_c)}^c}_{\mathbf{e}_c}\right] \left[\underbrace{\mathbf{U}_{(1)}^u \cdots \mathbf{U}_{(n_u)}^u}_{\text{unlabeled}}\right]\right], \tag{13a}$$

$$U = \left[\begin{bmatrix} 1 & \cdots & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & & 0 & 1 & \cdots & 1 & 0 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & & 0 & 0 & \cdots & 0 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{11}^u & \cdots & u_{1n_u}^u \\ u_{21}^u & \cdots & u_{2n_u}^u \\ \vdots & & \vdots \\ u_{c1}^u & \cdots & u_{cn_u}^u \end{bmatrix}\right].$$

A. M. Bensaid et al., Partially supervised clustering for image segmentation, Pattern Recognition, 29(5), 1996, pp.859-871

Semi-supervised FCM (Bensaid et al.)

- Initializing *w* and prototypes

$$\mathbf{v}_{i,0} = \frac{\sum_{k=1}^{n_d} (u_{ik,0}^d)^m \mathbf{x}_k^d}{\sum_{k=1}^{n_d} (u_{ik,0}^d)^m}, \quad 1 \le i \le c.$$

- (E-step) Updating membership

$$u_{ik,t}^u = \left[\sum_{j=1}^{c} \left(\frac{\|\mathbf{x}_k^u - \mathbf{v}_{i,t-1}\|_A}{\|\mathbf{x}_k^u - \mathbf{v}_{j,t-1}\|_A} \right)^{2/(m-1)} \right]^{-1},$$
$$1 \le i \le c; \quad 1 \le k \le n_u; \quad t = 1, 2, \ldots, T.$$

- (M-step) updating the prototypes

$$\mathbf{v}_{i,t} = \left(\frac{\sum_{k=1}^{n_d} w_k (u_{ik,t}^d)^m \mathbf{x}_k^d + \sum_{k=1}^{n_u} (u_{ik,t}^u)^m \mathbf{x}_k^u}{\sum_{k=1}^{n_d} w_k (u_{ik,t}^d)^m + \sum_{k=1}^{n_u} (u_{ik,t}^u)^m} \right),$$
$$1 \le i \le c; \quad t = 1, 2, \ldots, T.$$

Repeat until converge

Boundary Characteristic for Histogram Improvement and Local Thresholding

light object of dark background

$$s(x,y) = \begin{cases} 0 & \text{if } \nabla f < T \\ + & \text{if } \nabla f \geq T \text{ and } \nabla^2 f \geq 0 \\ - & \text{if } \nabla f \geq T \text{ and } \nabla^2 f < 0 \end{cases}$$

- local processing due to using gradient of each small area
- (…)(-,+)(0 or +)(+,-)(…)
- all pixels that are not on an edge are labeled 0
- all pixels that are on the dark side of an edge are labeled +
- all pixels that are on the light side an edge are labeled -

Example

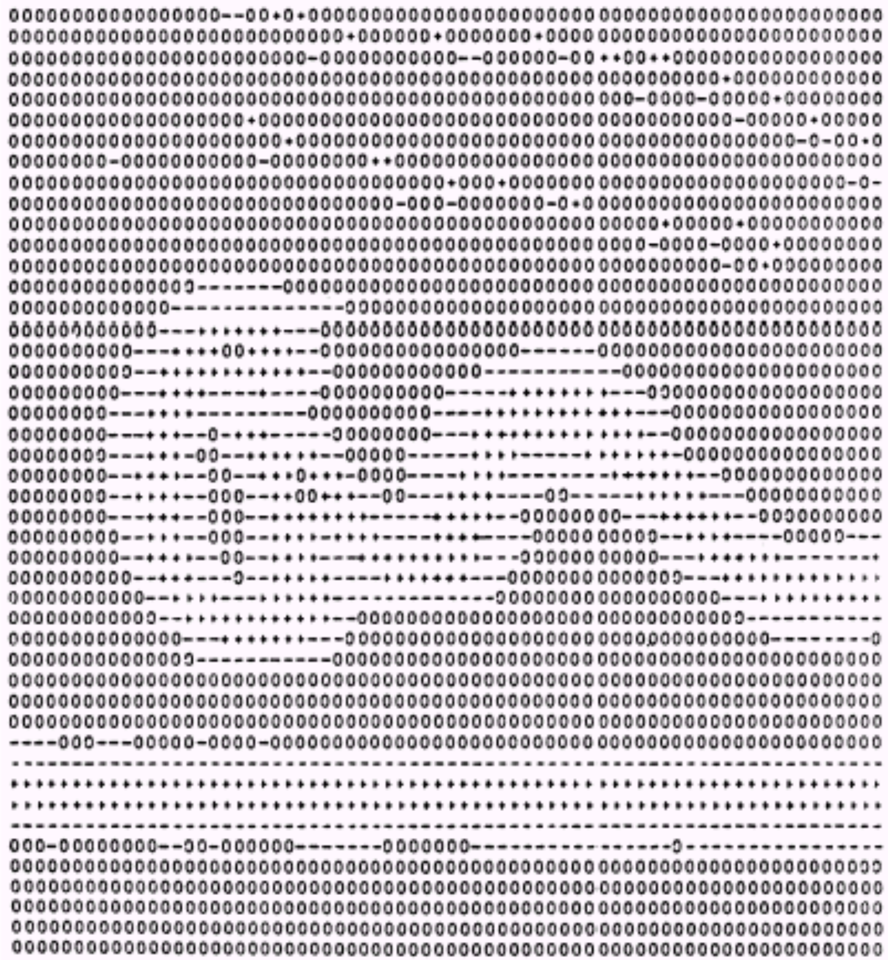


FIGURE 10.36 Image of a handwritten stroke coded by using Eq. (10.3-16). (Courtesy of IBM Corporation.)

Histogram of Gradient

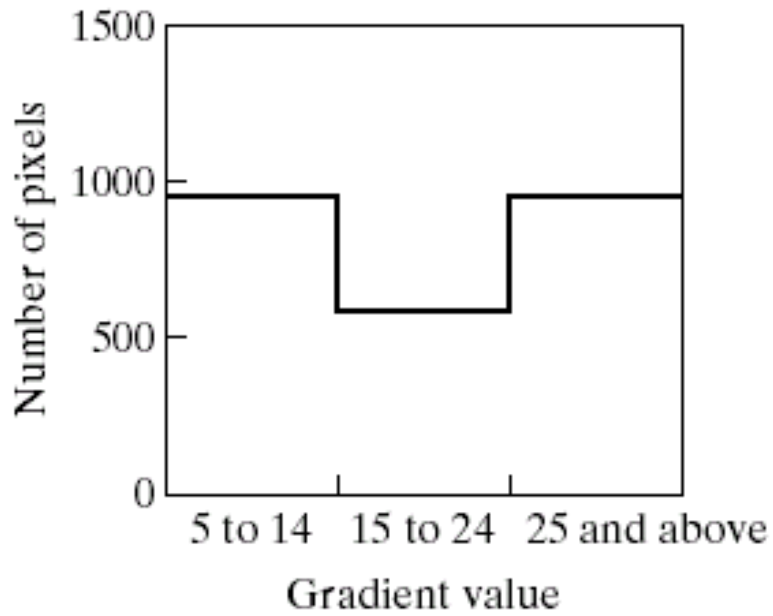


FIGURE 10.38 Histogram of pixels with gradients greater than 5. (Courtesy of IBM Corporation.)

select T at the midpoint of the valley

Result of applying T

a
b

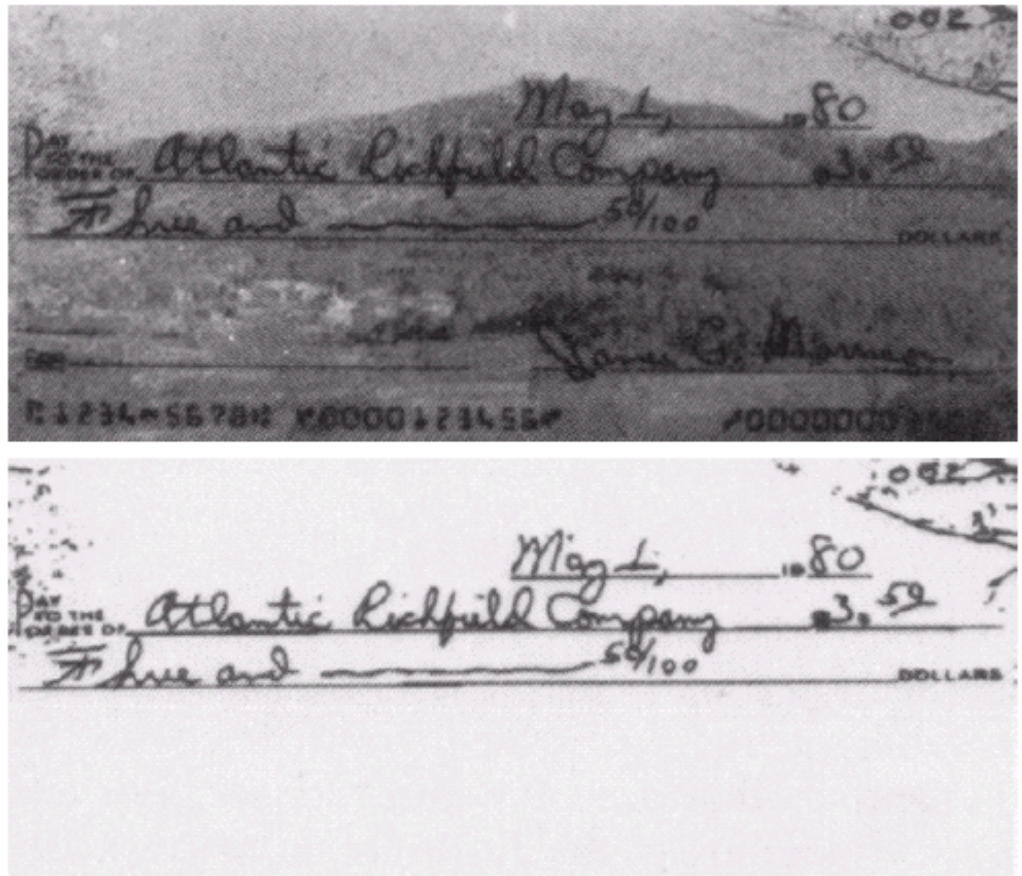
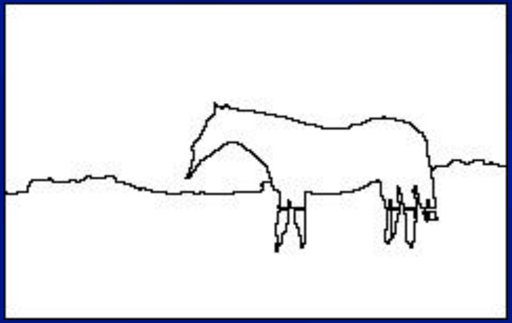


FIGURE 10.37
(a) Original image. (b) Image segmented by local thresholding. (Courtesy of IBM Corporation.)

General Approaches



- Contour-based
- Hybrid-approach
- Region-based
- Clustering-based

Clustering-based methods

Info from image:
- **luminance/color**
- **texture**
- **motion**

Feature Extraction → **Clustering pixels** → **Post Processing** → **Regions**

Contextual Info:
- **spatial/temporal coherence**

Sufficient information: most time, depending on applications

Clustering based methods

- In principle, all data clustering algorithms can be used in clustering based image segmentation
  - Thresholding
  - K-means
  - Fuzzy C-means

Basic Adaptive Thresholding

- Subdivide original image into small areas.
- Utilize a different threshold to segment each subimages.
- Since the threshold used for each pixel depends on the location of the pixel in terms of the subimages, this type of thresholding is ***adaptive***.

Histogram of the example

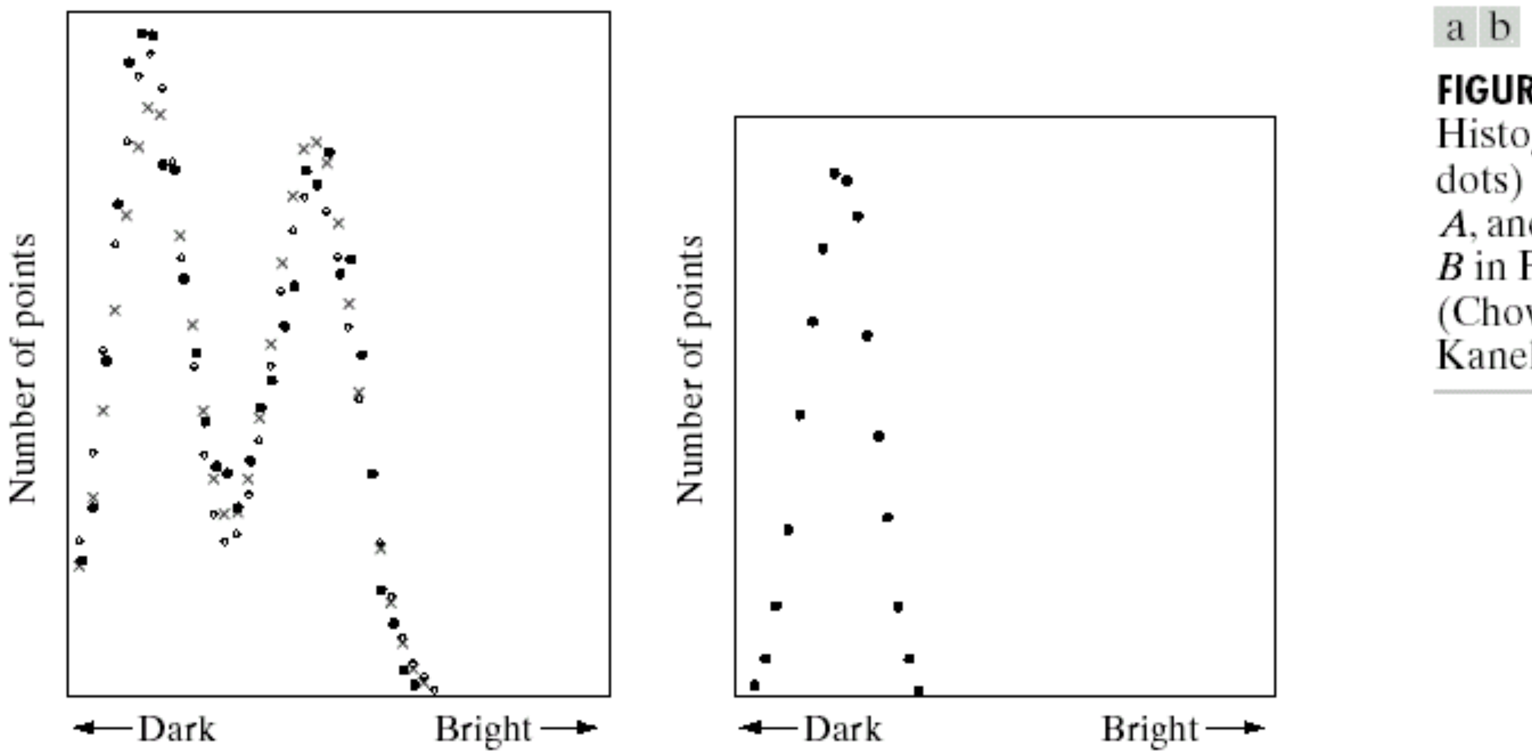


FIGURE 10.34 Histograms (black dots) of (a) region *A*, and (b) region *B* in Fig. 10.33(b). (Chow and Kaneko.)