Image segmentation and scene description

2 IMAGE SEGMENTATION

2 Image segmentation

1. Definitions, representation and evaluation results
2. Region based methods
3. Clustering based methods
4. Other methods
5. Actual methods

Today (1,2,3): we will see very easy and simple methods!

Next class (4,5): more complex!

2 Image segmentation
2.1 Definitions

Image segmentation [Haralick&Shapiro]

An image segmentation is the partition of an image into a set of nonoverlapping regions whose union is the entire image. The purpose of segmentation is to decompose the image into parts that are meaningful with respect to a particular application....

It is very difficult to tell a computer program what constitutes a "meaningful" segmentation. Instead, general segmentation procedures tend to obey the following rules.

2 Image segmentation
2.1 Definitions

Rules:

1. Regions of an image segmentation should be uniform and homogeneus with respect to some characteristics, such as grey level or texture.
2. Region interiors should be simple and without many holes
3. Adjacent regions of a segmentation should have different values with respect to the characteristic on which they are uniform.
4. Boundaries of each segment should be simple, not ragged, and must be spatially accurate.

2 Image segmentation
2.1 Definitions

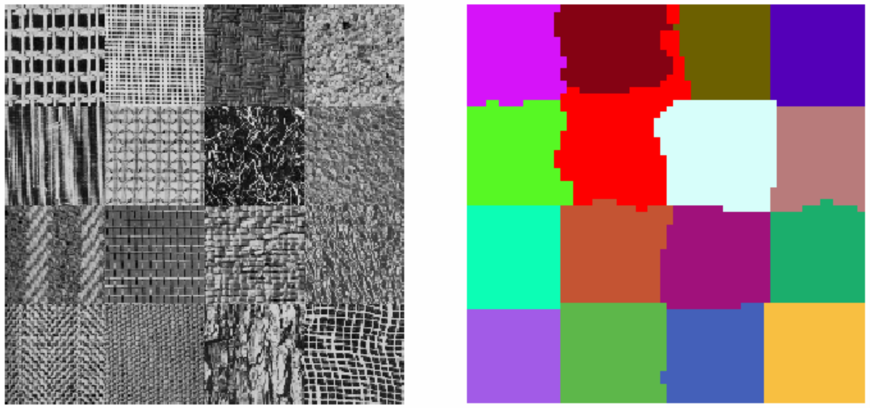
- The literature on Segmentation is large and generally inconclusive
- In some applications Segmentation is the crucial step

[*Hager*]

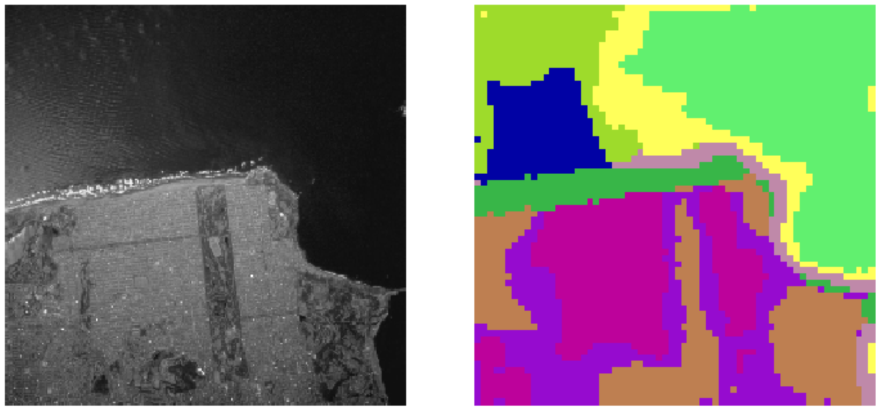
- “Old problem” in Computer Vision, but it is a very open field …
- “Segmentation is one of the oldest, and still unsolved, areas of Computer Vision”
- “In complex cases the segmentation problem can be very difficult and might require application of a great deal of domain knowledge”

2 Image segmentation
2.1 Definitions

Texture Test Image



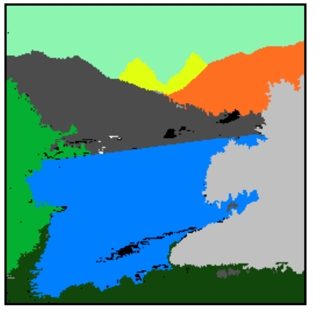
Aerial Image



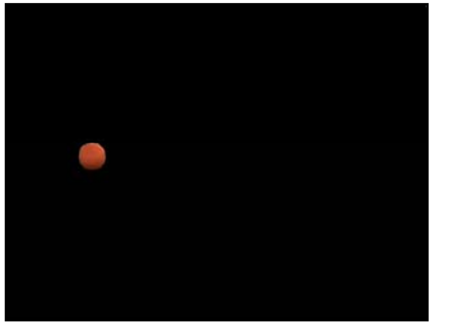
2 Image segmentation
2.1 Definitions

An initial classification:

- Unsupervised segmentation


- Purposive segmentation

2 Image segmentation
2.1 Evaluation results

How to evaluate? The common practice is the comparison with manual segmentation (called groundtruth):

- Using real images → hard work, non objective task
- Using synthetic images → other problems! That are not real

Two typical measures:

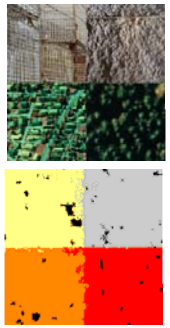
• Contour Criteria: To compare obtained region limits vs. original image contours

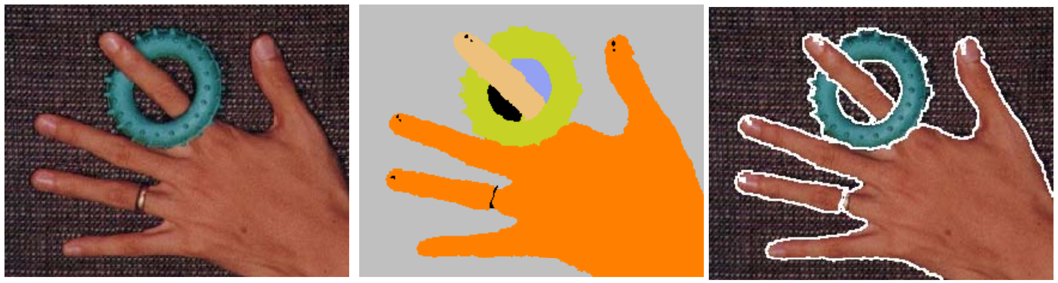
• Region Criteria: Area intersection (obtained region and original image region). Pixels belonging to a object and classified as a background (and vice versa)

2 Image segmentation
2.1 Evaluation results

Oversegmentation / Undersegmentation

How to show segmentation results:





2 Image segmentation
2.1 Evaluation results

ROC (Receiver Operated Curve): comparing the ground truth with the algorithm result

doctor / Algorithm	Mass	No Mass
Mass	TP	FP
No Mass	FN	TN

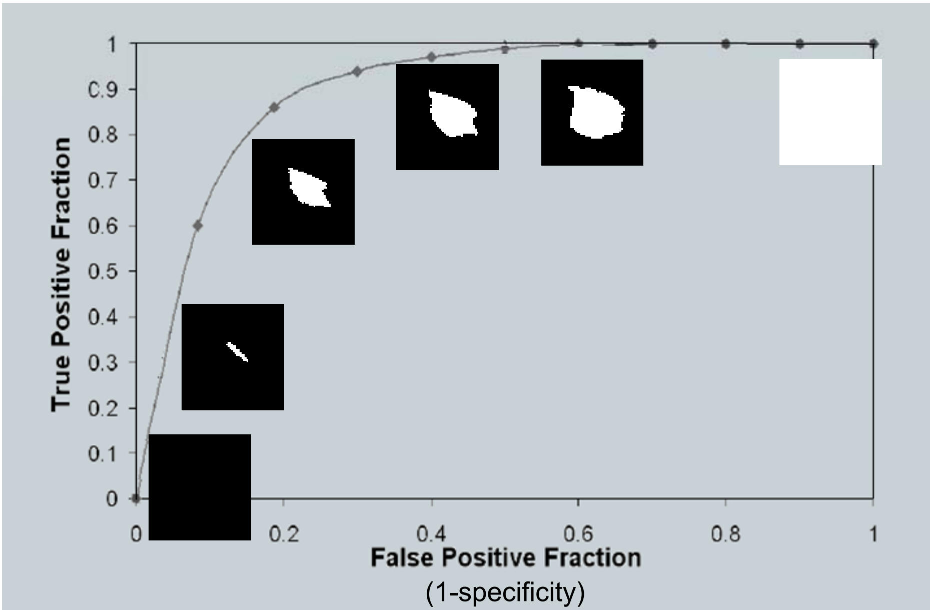
Sensitivity (Mass column) Specificity (No Mass column)

$$\text{sensitivity} = \frac{\text{number of True Positives}}{\text{number of True Positives} + \text{number of False Negatives}}.$$

$$\text{specificity} = \frac{\text{number of True Negatives}}{\text{number of True Negatives} + \text{number of False Positives}}$$

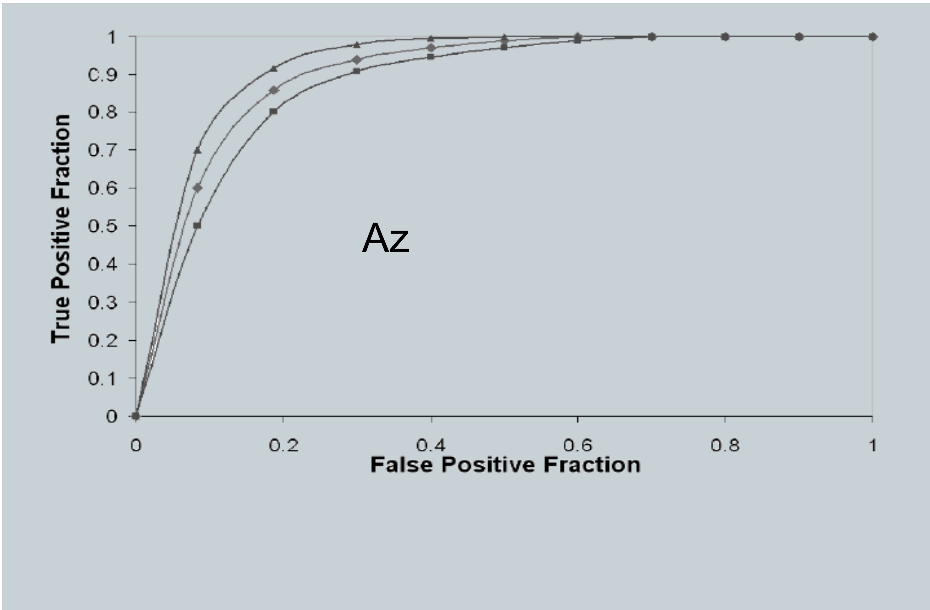
2 Image segmentation
2.1 Evaluation results

ROC (Receiver Operated Curve): comparing the hand segmented with the algorithm result



2 Image segmentation
2.1 Evaluation results

ROC (Receiver Operated Curve): comparing the hand segmented with the algorithm result

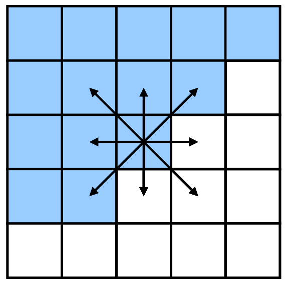


2 Image segmentation

2. Region based methods
   - Region Growing
   - Split & Merge
3. Clustering based methods
   - Thresholding methods
   - K-means
   - EM algorithms

2 Image segmentation
2.2 Region based methods

Region Growing



Considerations:

- Implementation: recursive, sequential, concurrent,...
- Seed distribution: how many and placement
- Aggregation criteria: intensity, color, texture. A typical one is:

$$\text{If } |f(x,y) - \mu_{Ri}| <= \Delta \qquad \text{then add } f(x,y) \text{ to } Ri, \text{ update } \mu_{Ri}$$

2 Image segmentation
2.2 Region based methods

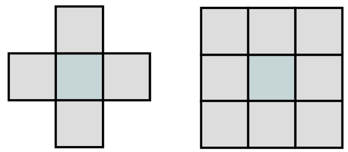
```
PROGRAM Region_Growing

Mark all the pixels as not considered

FOR all the pixels (x,y) of the image  DO
        IF pixel(x,y) no considered THEN
                Begin statistics new region Ri
                Mark pixel(x,y) as a considered
                Explore(x,y)
                Increase the number of seeds
        ENDIF

ACTION Explore(x,y)
WHILE   ((x',y') is an adjacent pixel respect to (x,y) not considered)
and ((x',y') belongs to actual region) DO
        Mark pixel(x',y') as a considered
        Recompute statistics of region Ri
        Explore(x',y')
```

A4 o A8

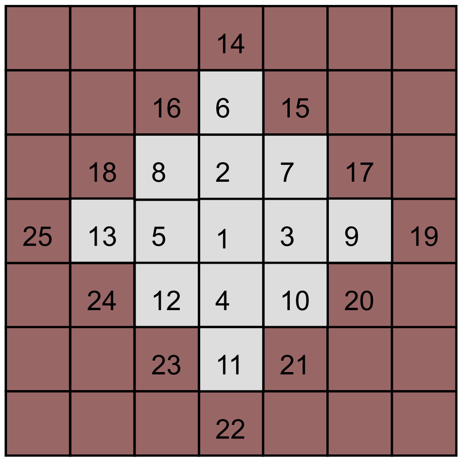


Aggregation Criteria: intensity

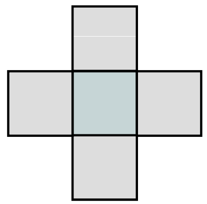
If $|f(x,y) - \mu Ri| <= \Delta$

2 Image segmentation
2.2 Region based methods

Implementation example: Visiting the neighbouring pixels using a queu

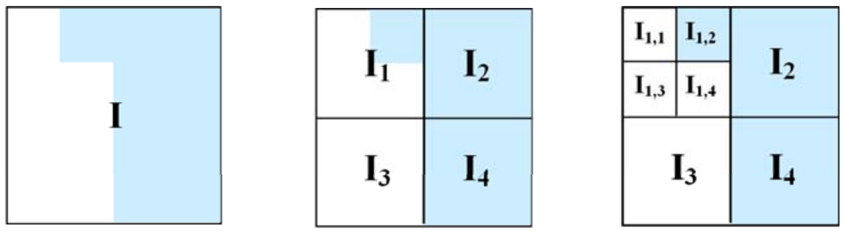


connectivity A4

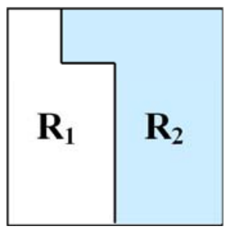


2 Image segmentation
2.2 Region based methods

Split & Merge (Pavlidis method)



Split



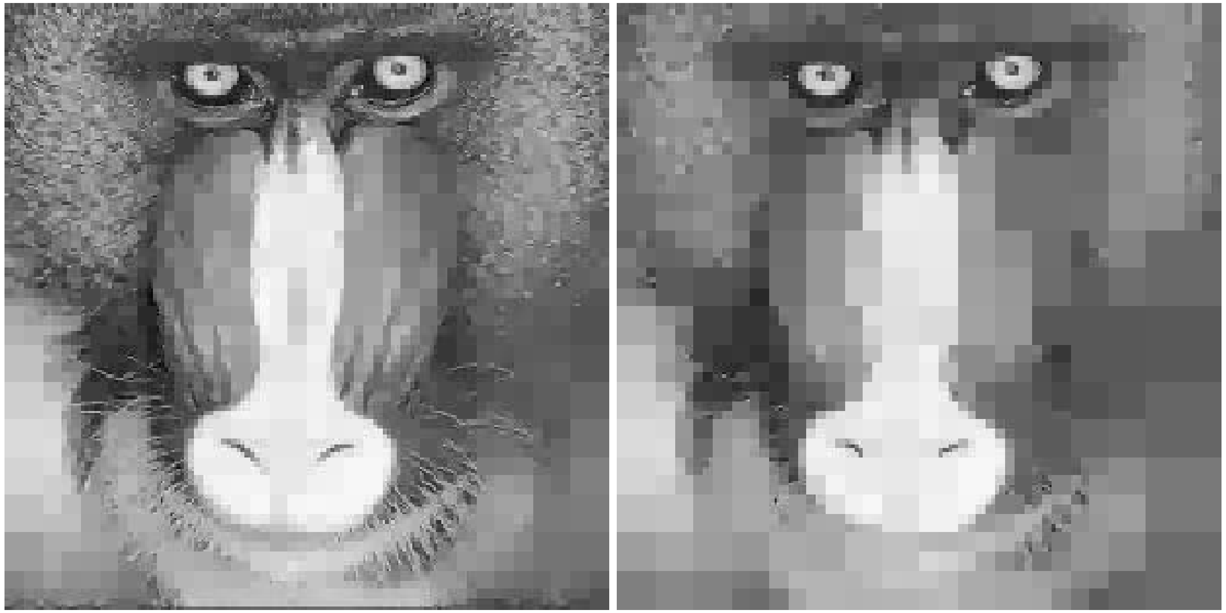
Merging

Considerations :

- Criteria and features for splitting and merging: intensity,color,texture
- Quad-tree representation
- Usually has high cost on merging, and pixeled aspect of result

2 Image segmentation
2.2 Region based methods

Split & Merge results



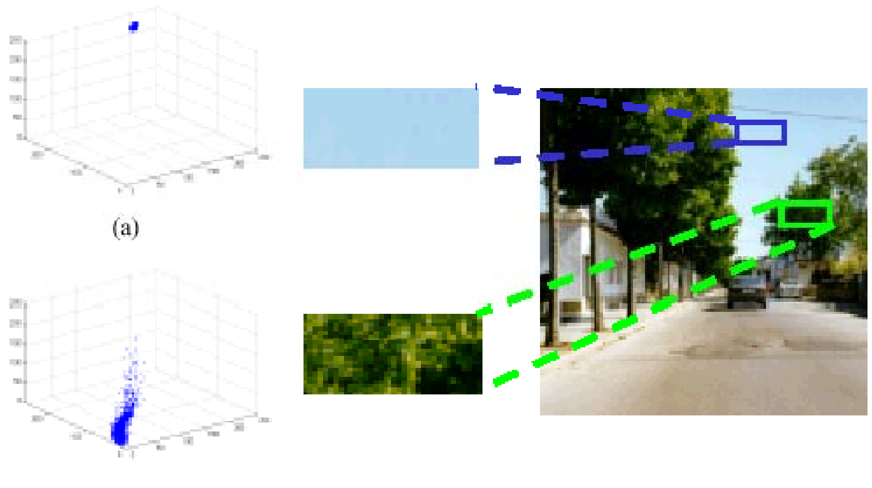
2 Image segmentation

2. Region based methods
   - Region Growing
   - Split & Merge
3. Clustering based methods
   - Thresholding methods
   - K-means
   - EM algorithms

2 Image segmentation
2.3 Clustering based methods

Image segmentation [Haralick&Shapiro]

The difference between image segmentation and clustering is that in clustering, the grouping is done in measurement space: In image segmentation, the grouping is done in spatial domain of the image.


(a)
(b)

2 Image segmentation
2.3 Clustering based methods

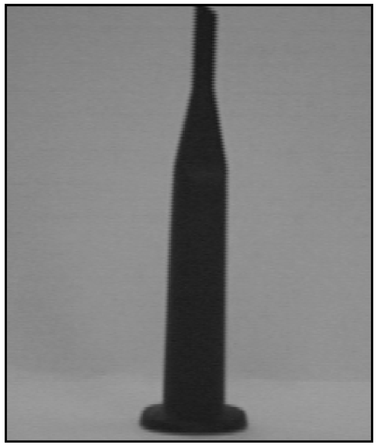
Cluster: grouping of pixels considering 1 o more features

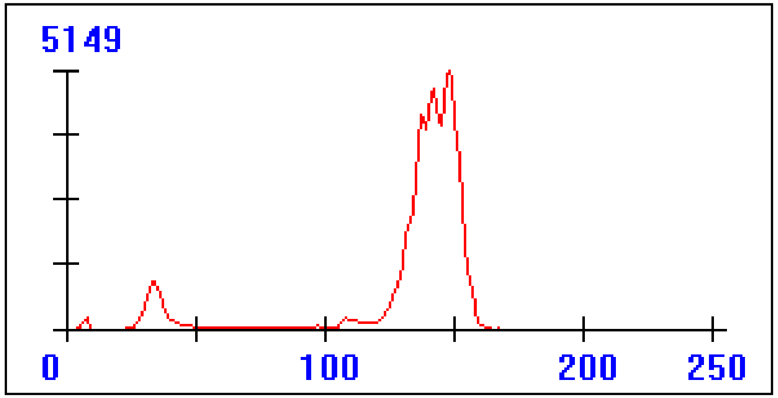
- Feature selection: Feature Space (R,G and B, in the previous Figure)
- Similar features… in a cluster
- Features significantly different between different clusters
- The spatial distribution of pixels in the image is not considered (feature space is considered)
- Shape of clusters: compact but, spherical, ellipsoidal, elongated, …
- Ownership of a pixel to a cluster depends on a proximity measure (distance). Grouping Methodology !!!
- Results may be subjective…

Key point: number of clusters of an image ??

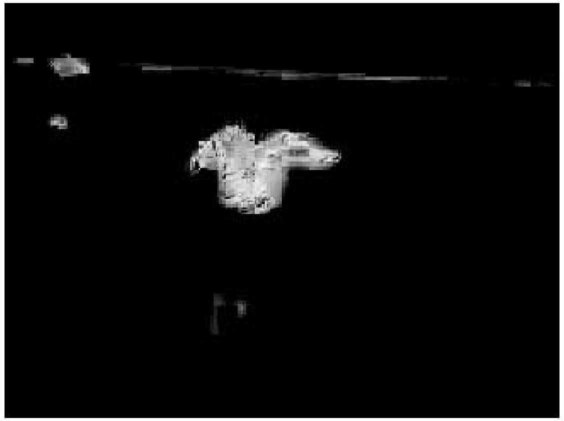
2 Image segmentation
2.3 Clustering based methods

Clustering en 1D = Histogram analysis → thresholding





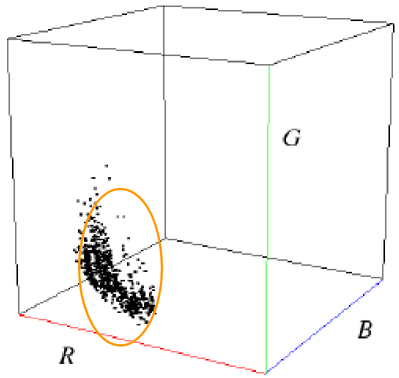
2 Image segmentation
2.3 Clustering based methods



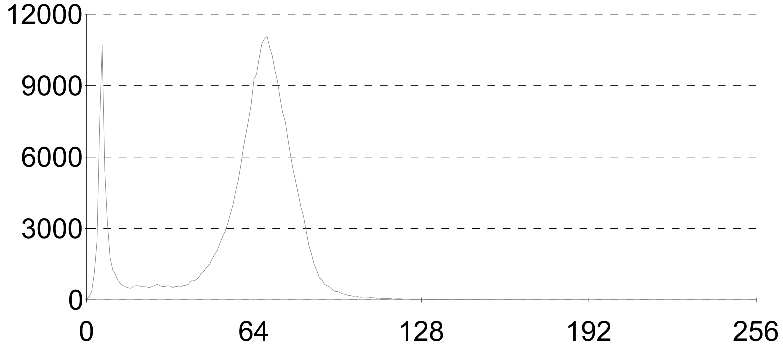
Region characterization

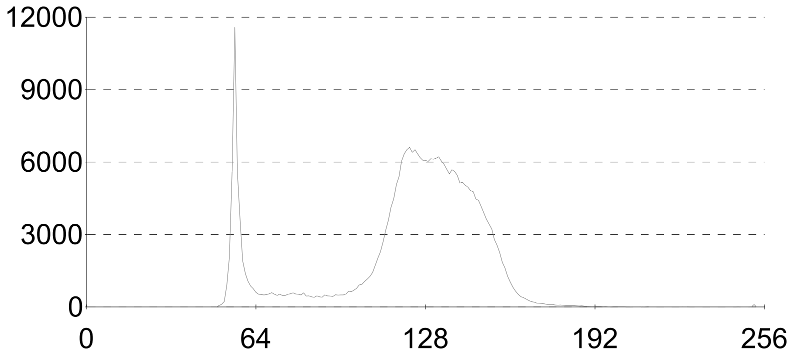
based on colour R,G,B

cluster

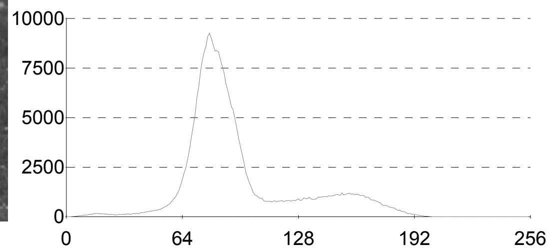


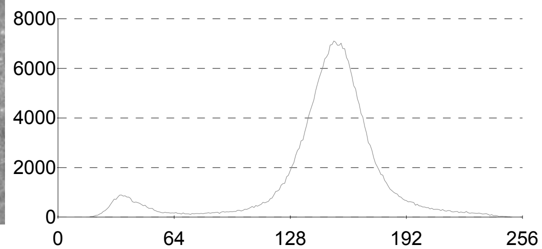
2 Image segmentation
2.3 Clustering based methods





2 Image segmentation
2.3 Clustering based methods





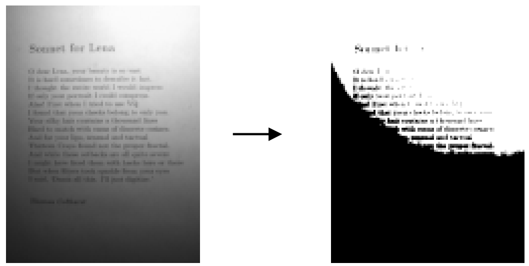
2 Image segmentation
2.3 Thresholding

Image thresholding

2 Image segmentation
2.3 Thresholding: fixed or adaptive

- In fixed or global thresholding, the threshold value is held constant throughout the image
- A variation could uses two or more thresholds

Global Thresholding



Local/Adaptive Thresholding

7x7; T = mean 7x7; T = mean-7 75x75; T = mean-10

2 Image segmentation
2.3 Optimal thresholding

Methods able to find the optimal threshold:

- Isodata, Peak and valley, Otsu, p-tile,...

- What happens when appearing more than two modes: Ohlander,...

Let's see a couple of examples

2 Image segmentation
2.3 Optimal thresholding

ISODATA (Iterative Self-Organising Data Analysis Technique Algorithm)

Initially the histogram is segmented in 2 parts. Then, we compute the mean value associated to each part of the histogram m1, m2. With these values we compute a new threshold doing T= (m1+m2) / 2. We repeat the process until convergence

ISODATA step by step:

1)Chose the initial threshold Ti=T0 (median histogram, maximum gray level…)
2)Divide the image in two groups R1 and R2 using Ti
3)Compute the mean of the gray level for both parts m1 and m2
4)Select the new threshold Ti+1 = (m1 + m2)/2
5)Repeat steps 2-4 until Ti = Ti-1

2 Image segmentation
2.3 Optimal thresholding

Example:

H(z)=

2	5	7	16	34	22	17	4	5	5	3	23	45	56	12	3

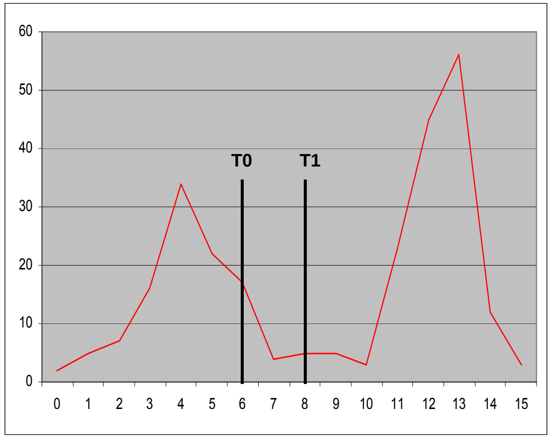
z has values from 0 a 15 (4 bits)

1) T_0=position(maximum)/2=13/2=6
   (we could also do T_0=mean of the histrogram $T_0=\Sigma\ z*H(z)/\ \Sigma H(z) \rightarrow T_0=8.85$)

2) $m_1=\Sigma\ z*H(z)/\ \Sigma H(z)$ for z = 0 to 6= 4.02
   $m_2= \Sigma\ z*H(z)/\ \Sigma H(z)$ for z = 7 to 15= 12.03
   T_1 = (m1 + m2)/2 = 8.03

3) $m_1=\Sigma\ z*H(z)/\ \Sigma H(z)$ for z = 0 to 8= 4.31
   $m_2= \Sigma\ z*H(z)/\ \Sigma H(z)$ per z = 9 to 15= 12.31
   $T_2 = (m_1 + m_2)/2 = 8.\ 31$

FINAL floor(T2)=floor(T3)



2.3 Optimal thresholding

Peak and valley method

1. Assign to **Z1** the gray level z which is the maximum of H(z)

2. Assign to **Z2** the gray level that **maximises g(z)=|Z1-z| * H(z)**

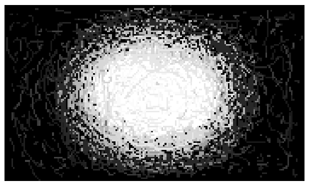
This is called to maximise the peakiness, the difference between peaks and valley

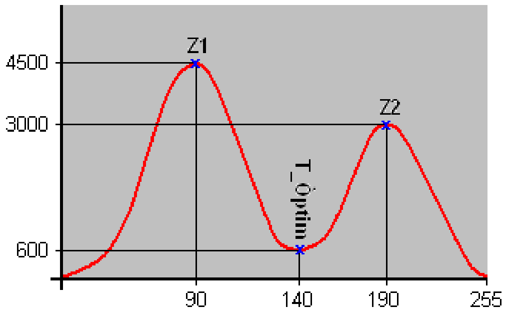
3. Assign the threshold **T** to the gray level that is the **minimum of H(z)** going:

from z=Z1+1 to Z2-1, if Z2>Z1
from z=Z2+1 to Z1-1, if Z2<Z1

2 Image segmentation
2.3 Optimal thresholding

- ✓ It works when the histogram has 2 well defined pics

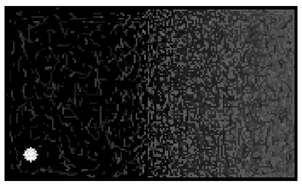


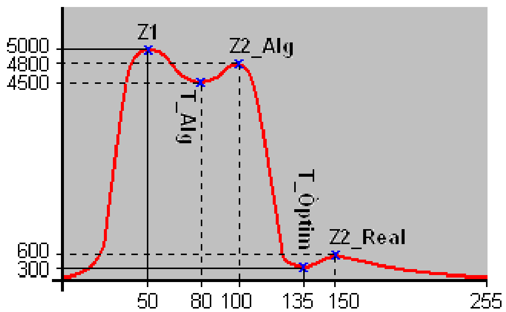


Computing Z2:

g(z)=|Z1-z|*H(z)
Z2→ g(z)max =
= |90-190|*3.000 = 300.000

- ✗ If **1 peak is very small** we can have problems computing Z2





Computing Z2:

g(z)=|Z1-z|*H(z)
Alg→ |50-100|*4.800 = 240.000
Real→ |50-150|*600 = 60.000

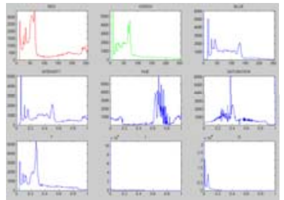
2 Image segmentation
2.3 Optimal thresholding

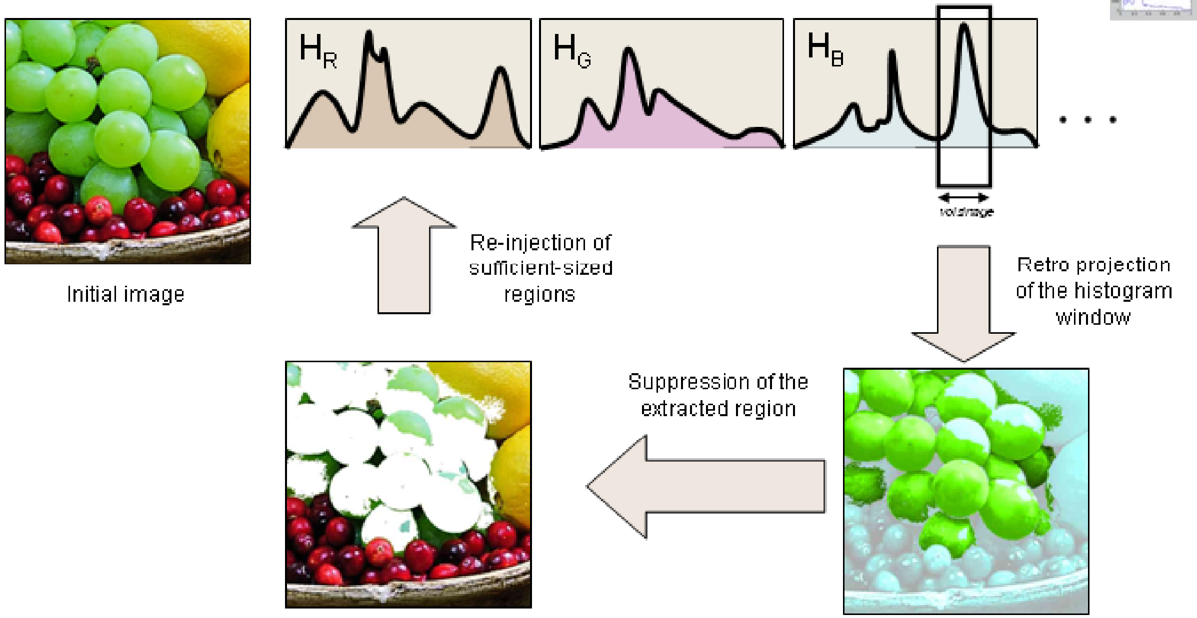
Ohlander method

- Region based algorithm using thresholding technique
- Based on exploiting the chromatic information by constructing colour and hue histograms
- Regions are split recursively based upon histogram analysis
- Picture is thresholded at its most clearly separated peak

2 Image segmentation
2.3 Optimal thresholding

Ohlander method





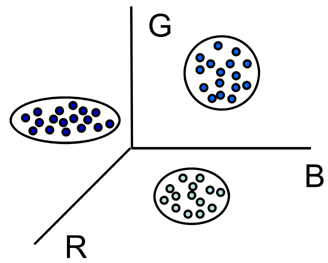
Initial image

2 Image segmentation

2. Region based methods
   - Region Growing
   - Split & Merge
3. Clustering based methods
   - Thresholding methods
   - K-means
   - EM algorithms

2 Image segmentation
2.3 Clustering based methods

K-means algorithm



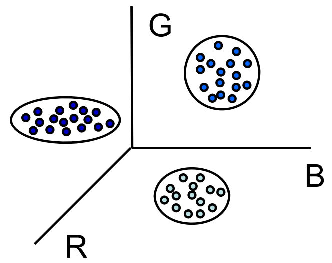
-Successive generation of clusters, minimizing a cost function J, obtaining the “best” clusters

- “a priori” knowledge of: # of clusters, ownership of every pixel to only one cluster Cj (minimum distance to the centroid Φj)

$$\phi j = \frac{1}{N} \sum_{Xi \in Cj} Xi \qquad d(Xi, \phi j) = \left\| Xi - \phi j \right\|$$

2 Image segmentation
2.3 Clustering based methods

K-means algorithm



Cost Function J:

$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{ij} \times d(x_i, \phi_j)$$

$$where\, \phi_j \; is\, the\, centroid\, of\; cluster\, C_j$$

- *Uij* is a coefficient related to the ownership of a pixel *i* to a cluster *j*:

    Uij = 1, if *d* is the minimum distance

    Uij = 0, otherwise

2 Image segmentation
2.3 K-means

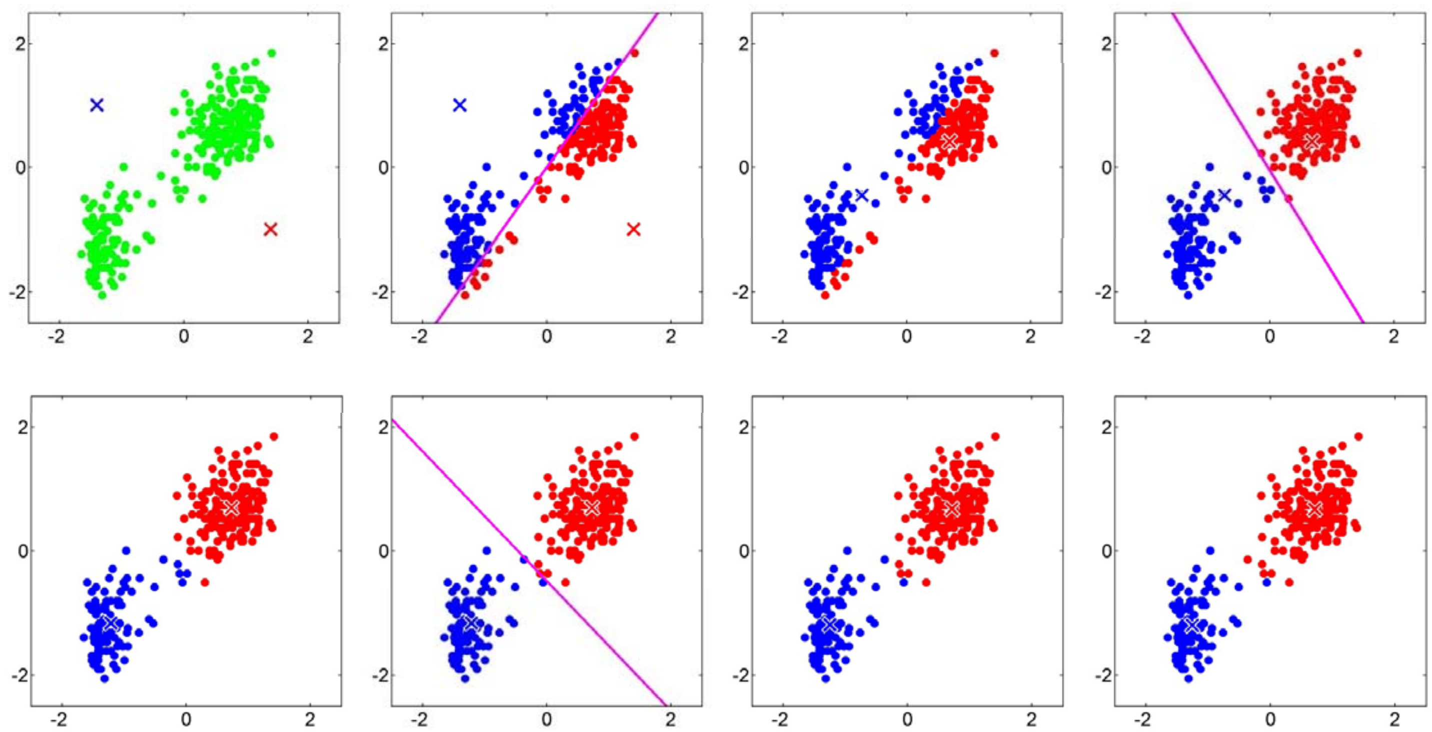
4 steps, iterative algorithm:

- **Phase 1**
  K is known
  Determination of the initial centers of each cluster ϕ_j, j = 1…K
- **Phase 2**
  Each pixel x_i is assigned to its nearest cluster C_j:
- **Phase 3**
  Compute the "new" centers of the clusters ϕ_j
- **Phase 4**
  If any pixel has changed the cluster (phase 2) then go to phase 2, else the algorithm finish.

Final goal, minimize: $$J = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{ij} \times d(x_i, \phi_j)$$

$$on \quad \phi_j \quad és \quad el \quad centroid \quad del \quad cluster \quad C_j$$

2 Image segmentation
2.3 K-means



Example taken from Christopher M. Bishop BCS Summer school, Exeter 2003

http://research.microsoft.com/~cmbishop/talks.htm

2 Image segmentation
2.3 Clustering based methods

```
PROGRAM K-Means(number_clusters,inicial_centers)
        centers=Compute_inicial_centers(number_clusters,inicial_centers)
        REPEAT
        FOR all the pixels (x,y) of the image  DO
            changes = Compute_cluster_pixel(x,y,centers) ENDFOR
        centers = Compute_centers(number_clusters)
        UNTIL NO changes
ENDPROGRAM

FUNCTION Compute_cluster_pixel(x,y,centers)
    cluster_previous = cluster_actual
    cluster_actual = nearest_cluster(x,y,centers)
    IF cluster_previous != cluster_actual THEN  RETURN true
    ELSE  RETURN false
ENDFUNCTION

FUNCTION Compute_centers(number_clusters)
    FOR all number_clusters i DO
        centers(i) = Mean_pixels_cluster(i)
    ENDFOR
    RETURN centers
ENDFUNCTION
```

2 Image segmentation
2.3 EM algorithm

- The Expectation-Maximization (EM) algorithm is an iterative technique designed for probabilistic models.
- It is often used for finding the unknown parameters of a mixture model.

INITIALIZE randomly the model parameters.

REPEAT

Expectation step:

Compute labels for all the dataset given the current cluster parameters.

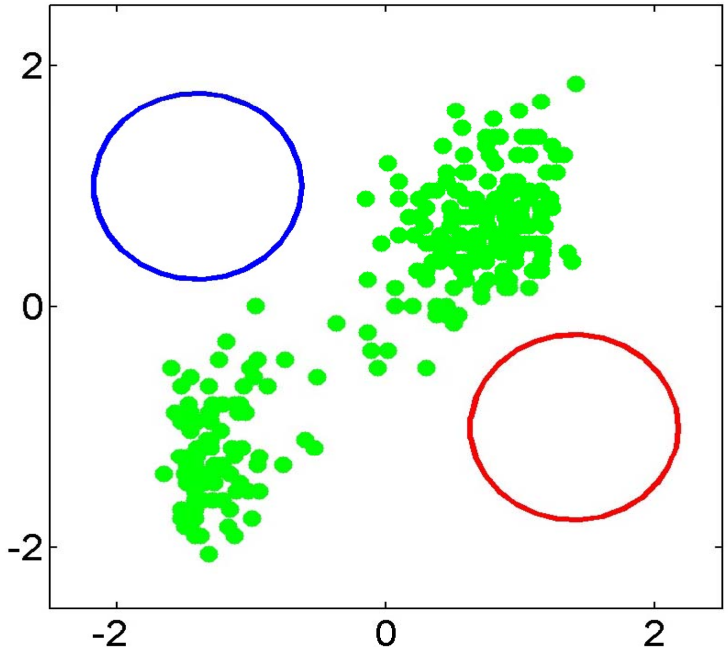
Maximization step:

Use that classification to reestimate the parameters

UNTIL convergence is achieved → not significant changes on the parameters

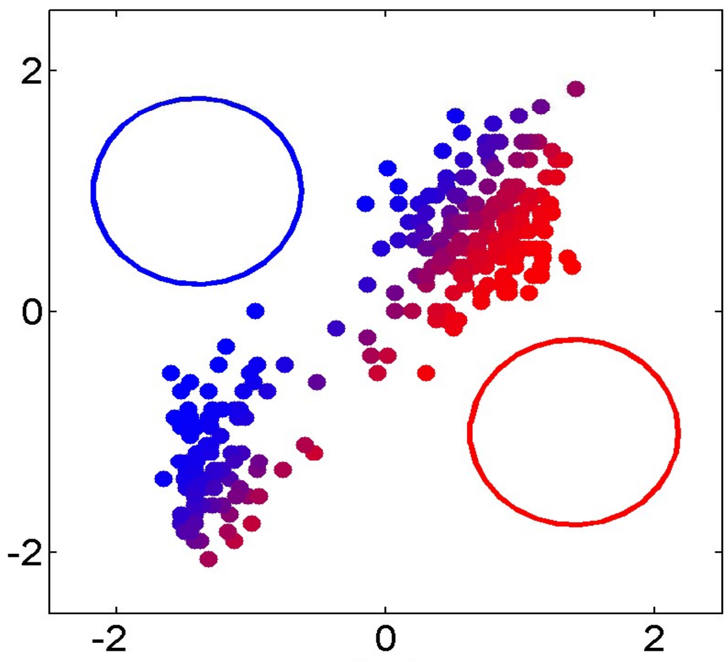
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians



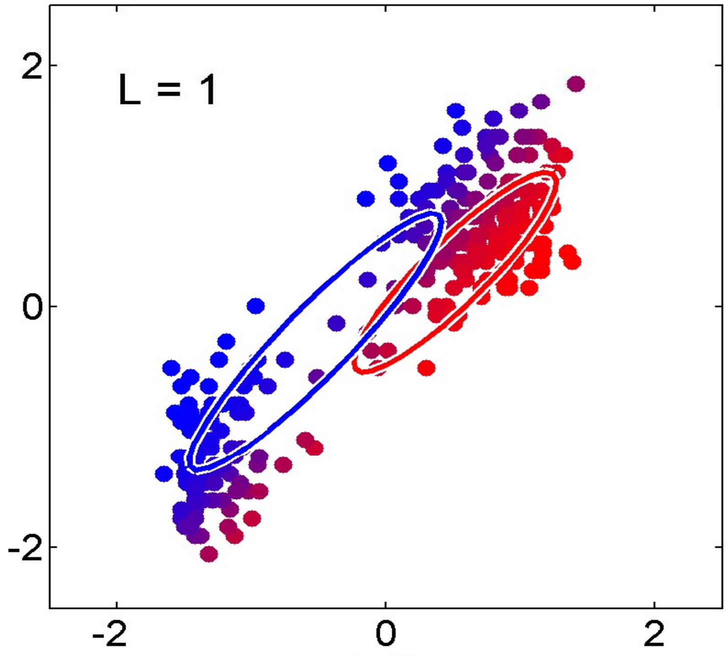
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians



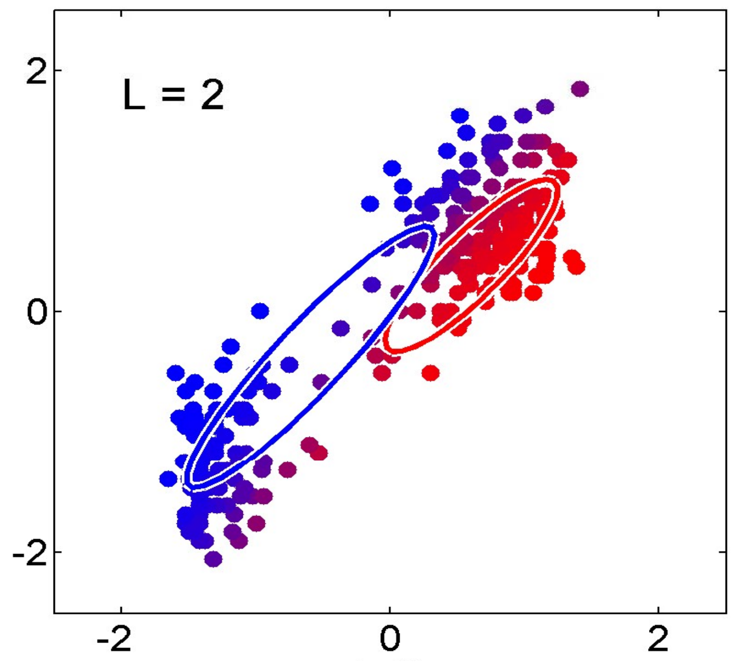
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians



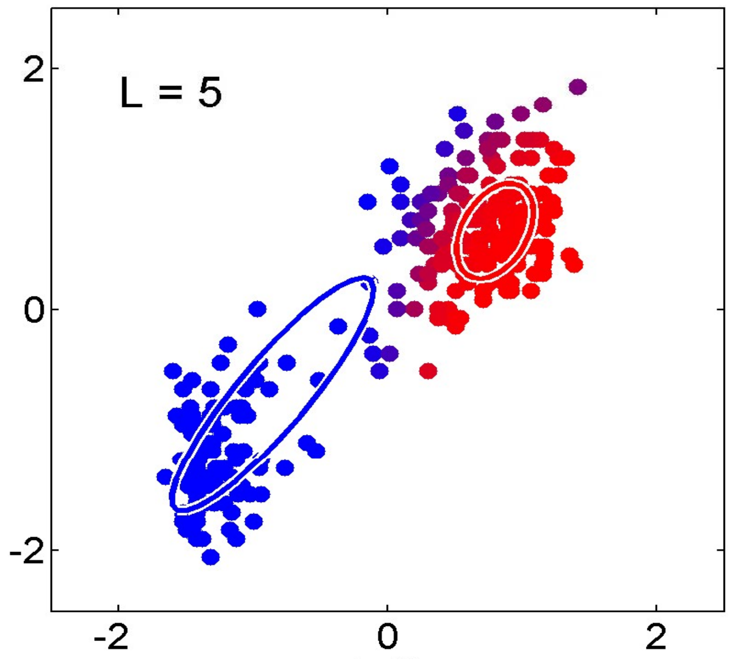
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians



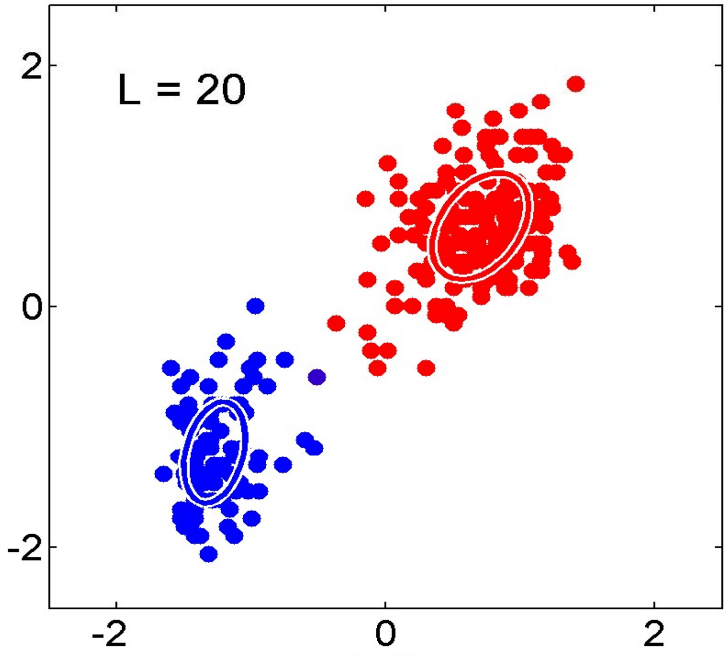
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians



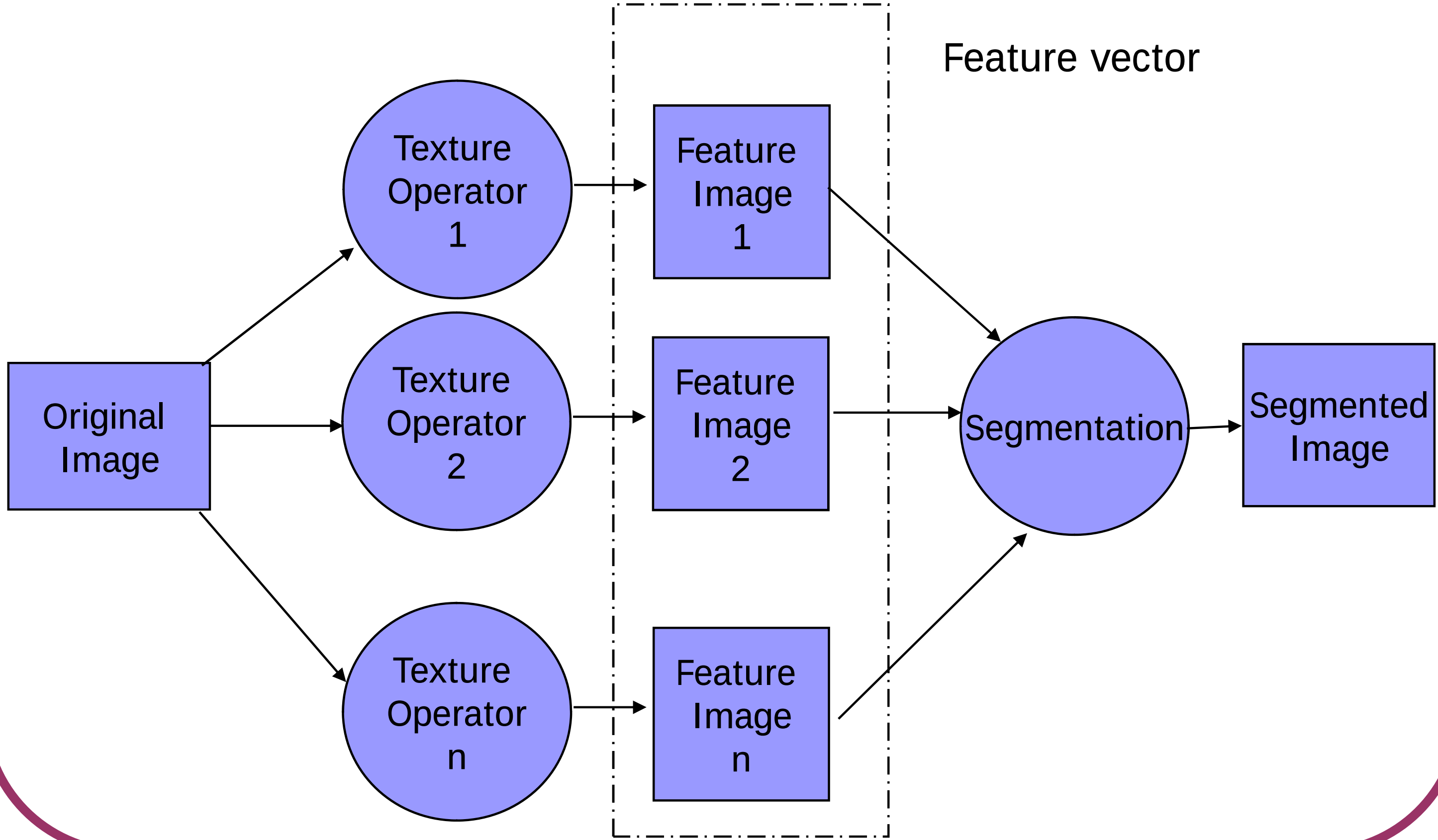
2 Image segmentation
2.3 EM algorithm

EM example: two class problem using mixture of Gaussians

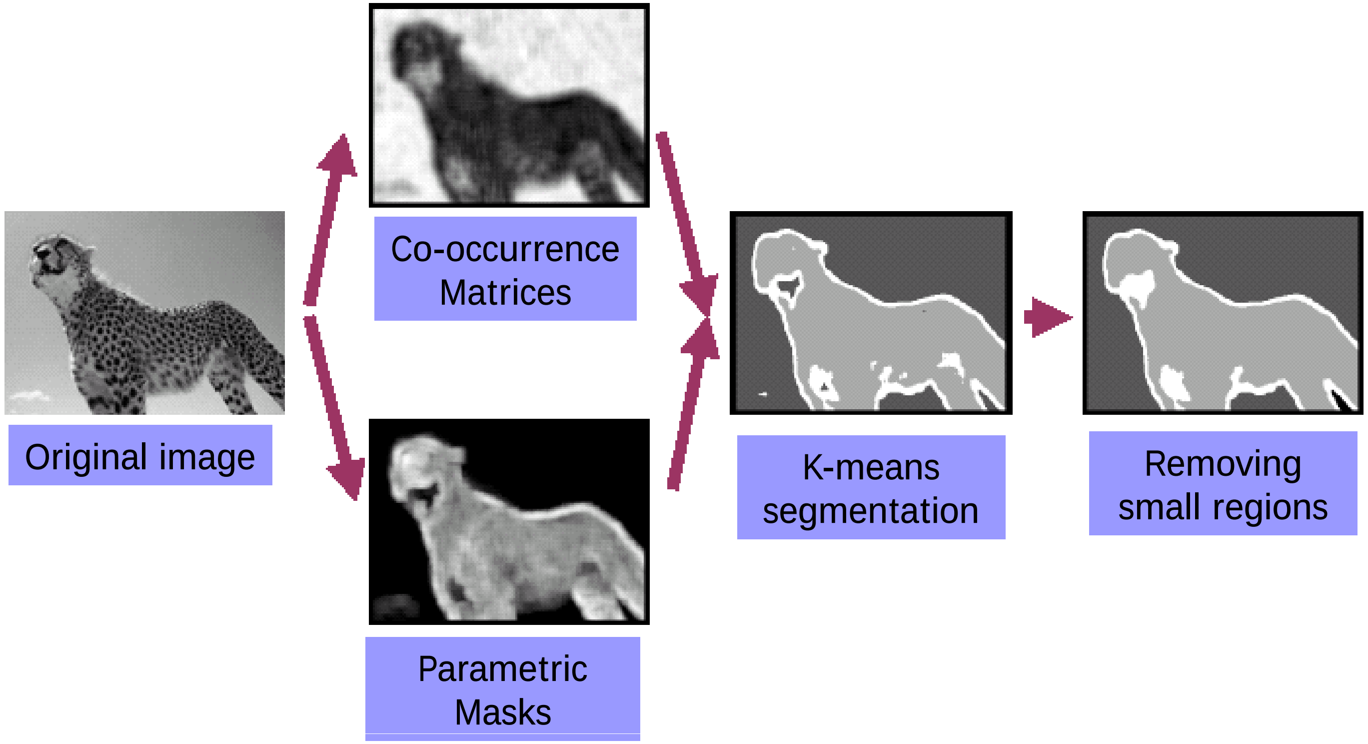


Segmentation and Texture

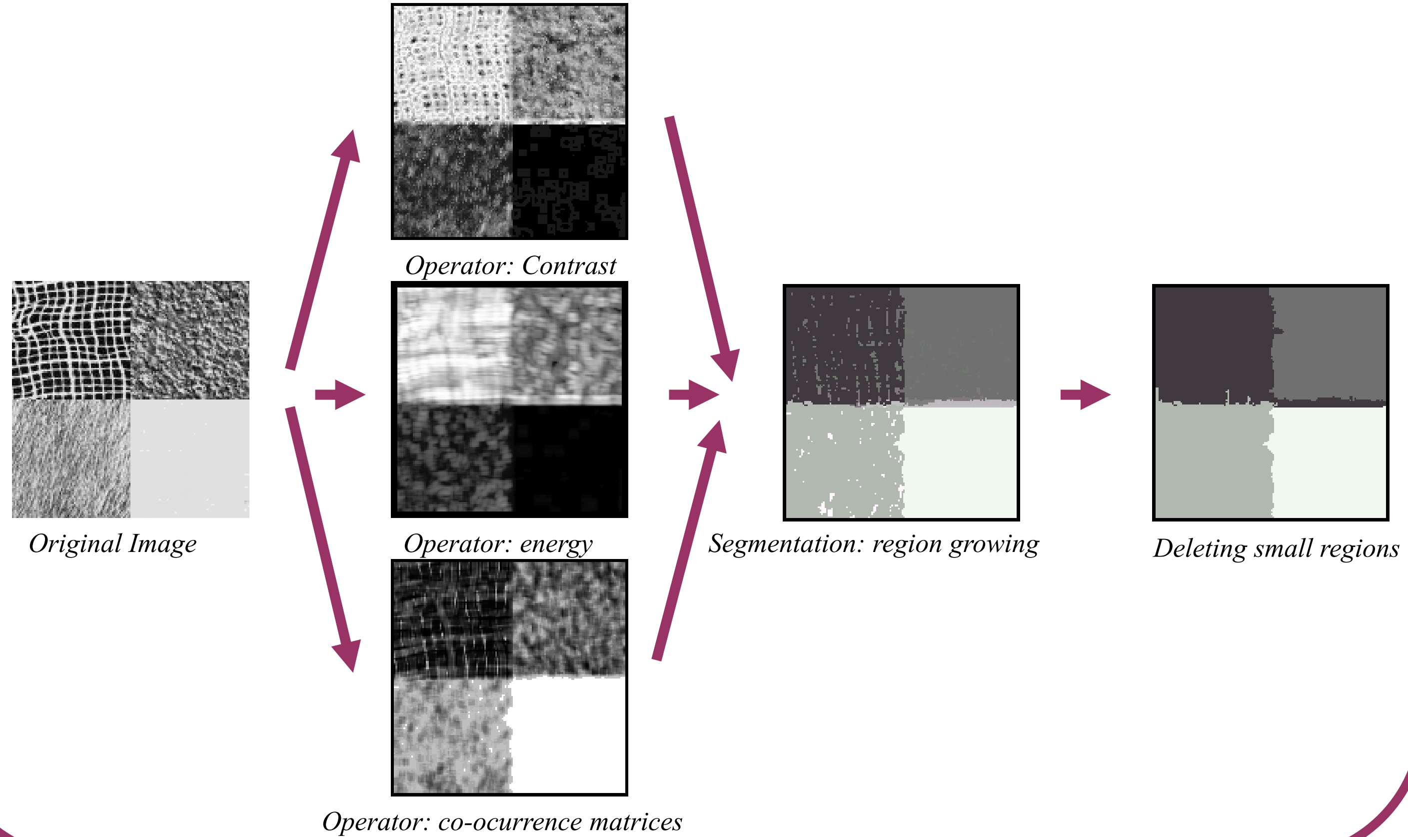
Segmentation and Texture



Segmentation and Texture



Segmentation and Texture



2 Image segmentation

1. Definitions, representation and evaluation results
2. Region based methods
3. Clustering based methods
4. Other methods
5. Actual methods

Last week (1,2,3): introduction and simple methods

Today (4,5): we will see more complex methods!