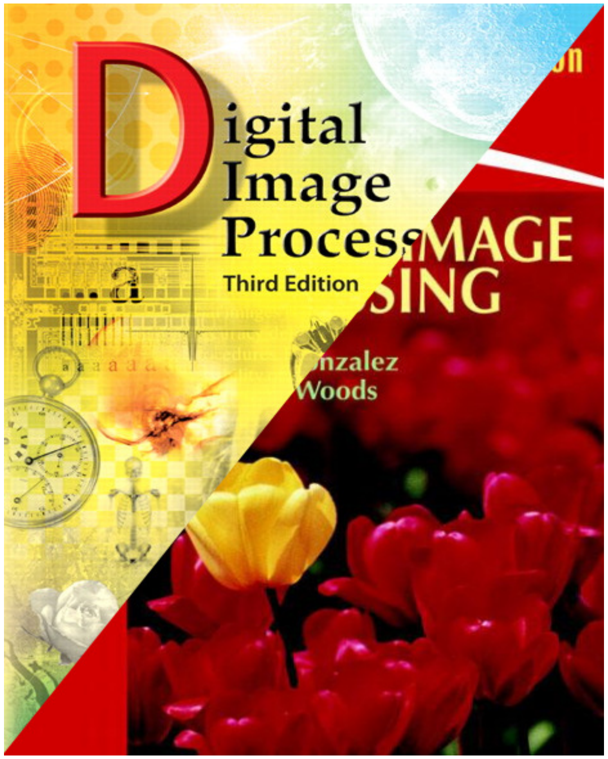
Lecture 8
Object Descriptors

Azadeh Fakhrzadeh
Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University

Reading instructions



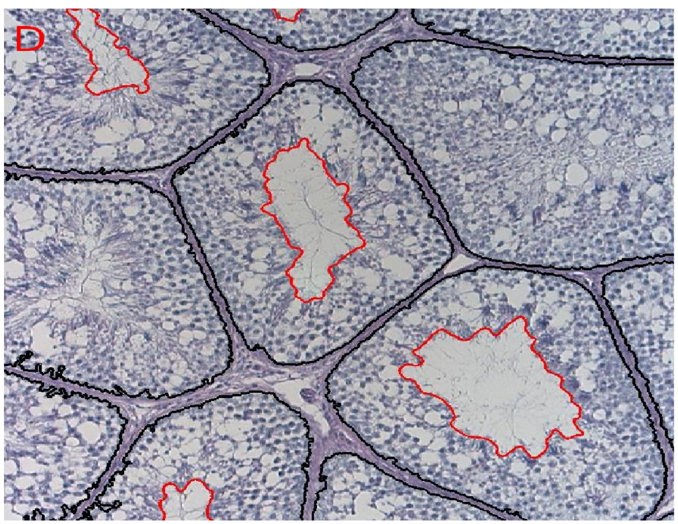
Chapter 11.1 – 11.4 in G-W

Azadeh Fakhrzadeh

- PhD student
- Research topic:

  study of morphological changes in the testicular tissue
- azadeh@cb.uu.se

www.udel.edu/biology



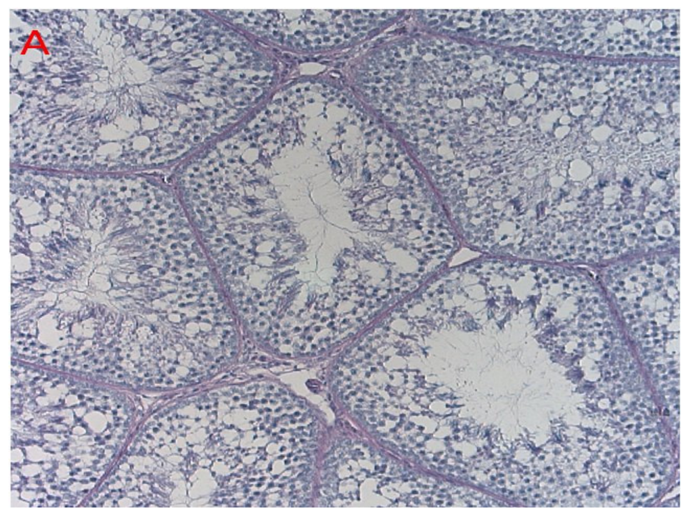
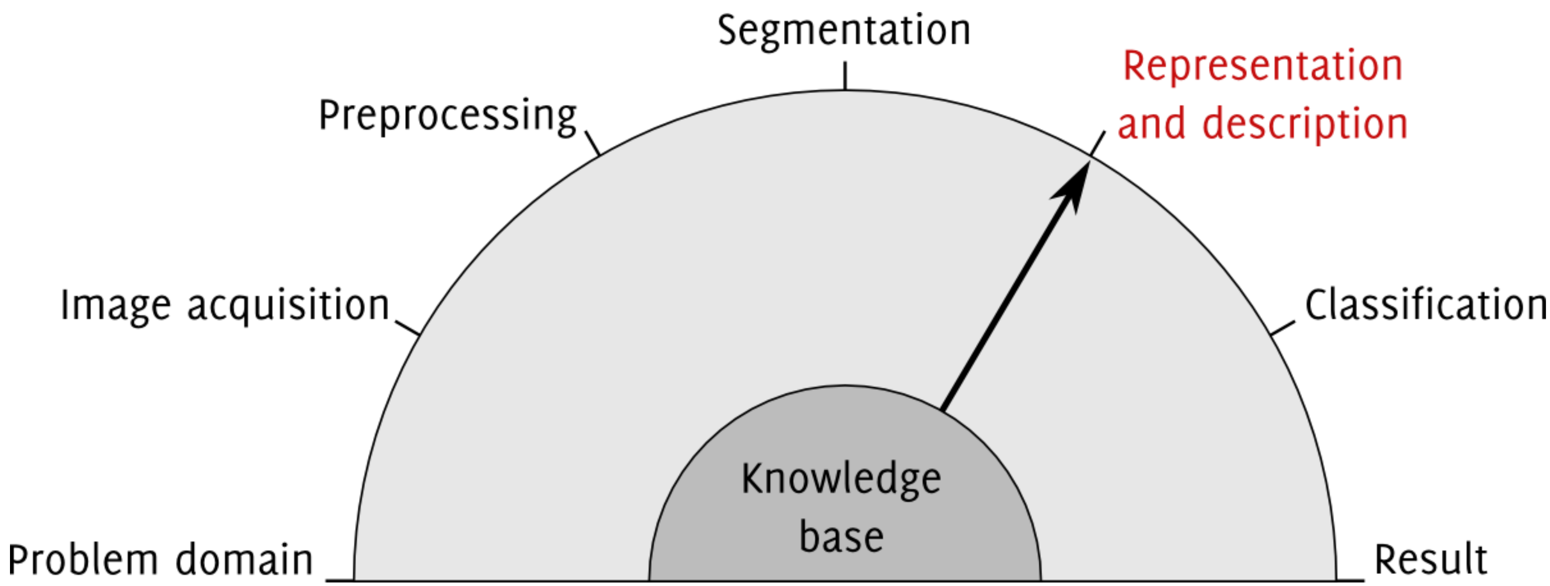
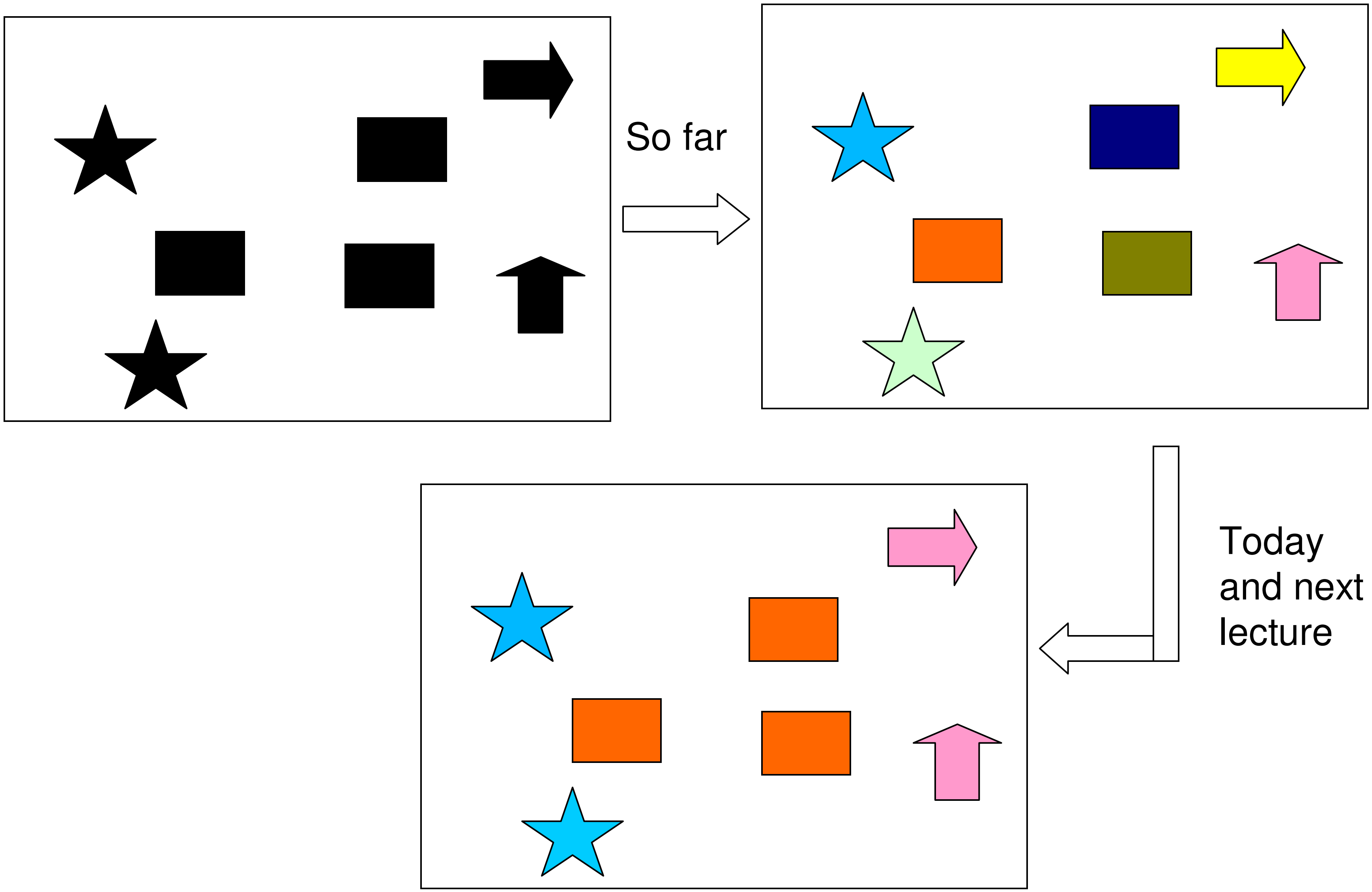


Image analysis

- Our progress in the analysis process




So far
Today
and next
lecture

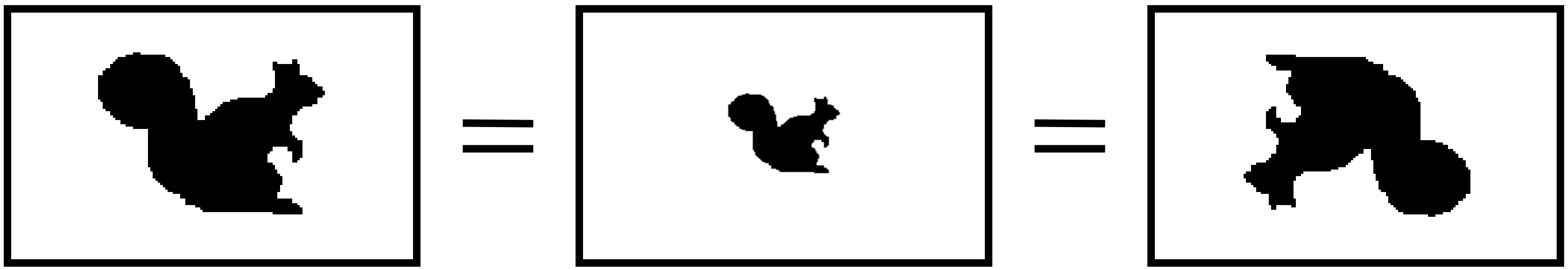
Representations and descriptors

Next step after segmentation is to represent the object in a good way that makes it Possible to describe it.

- Two ways to represent regions:
  - **Boundary** (external characteristics)
    - Shape, orientation
  - **Whole region** (internal characteristics)
    - Color, texture, histogram

Scale, rotation and translation

- Most of the time we are interested to choose descriptors that are invariant of variations of scale, rotation and translation whenever possible

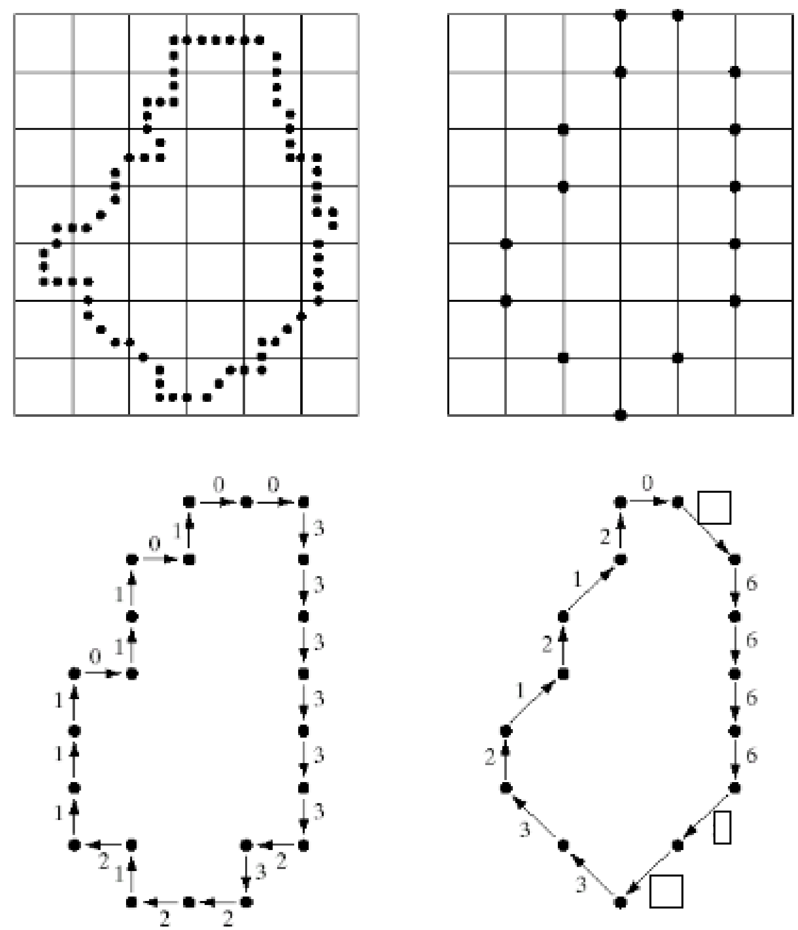


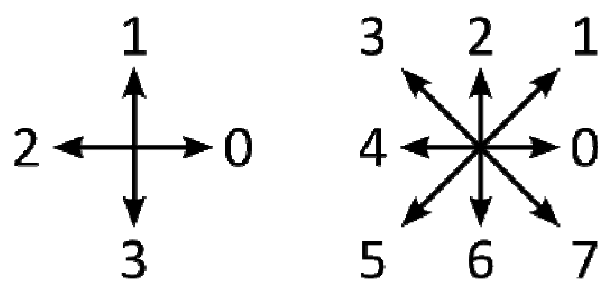
But not always in Optical character recognition (OCR) Rotation and scale is important like P and d

Chain Coding

Boundary representation

- Walk around the object boundary and describe directional change in each step by a number





Chain Coding

Considerations

- Code become very long and noise sensitive
  - Use larger grid spacing, 0710 = 00
- Scale dependent
  - Choose appropriate grid spacing
- Start point determines result
  - Treat code as circular (minimum magnitude integer)
    754310 → 075431
- Depends on rotation
  - Calculate difference code (counterclockwise)
    075431 → 767767

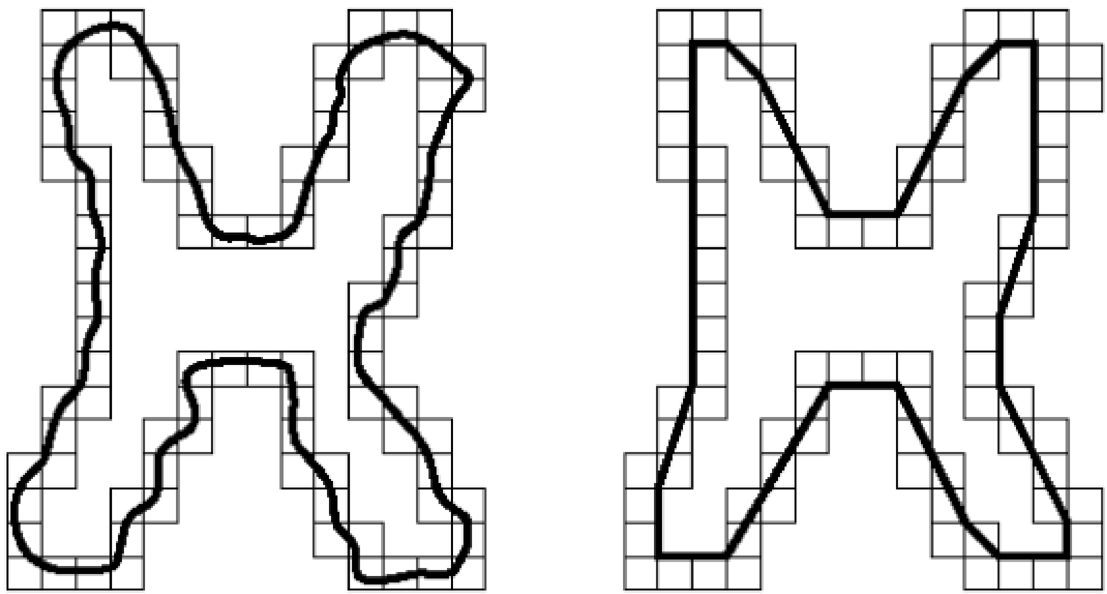
Polygonal Approximations

Boundary representation

- A digital boundary can be approximated (simplified)
- For closed boundaries:
  - Approximation becomes exact when no. of segments of the polygons is equal to the no. of points in the boundary
- Goal is to capture the essence of the object shape
- Approximation can become a time consuming iterative process

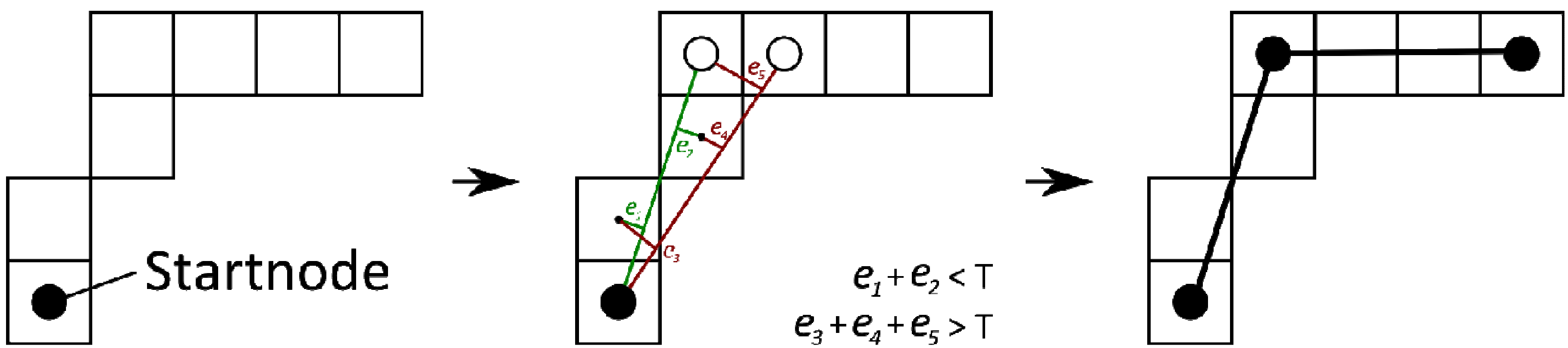
Polygonal Approximations

- Minimum Perimeter Polygons (MPPs)
  - Cover the boundary with cells of a chosen size and force a rubber band like structure to fit inside the cells



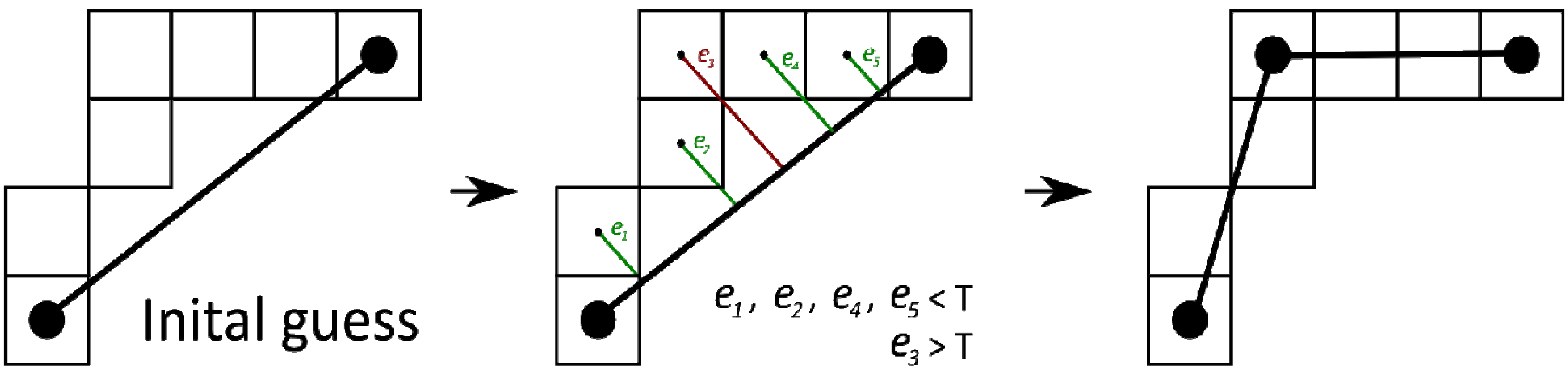
Polygonal Approximations

- Merging techniques
  1. Walk around the boundary and fit a least-square-error line to the points until an error threshold is exceeded
  2. Start a new line, go to 1
  3. When start point is reached the intersections of adjacent lines are the vertices of the polygon



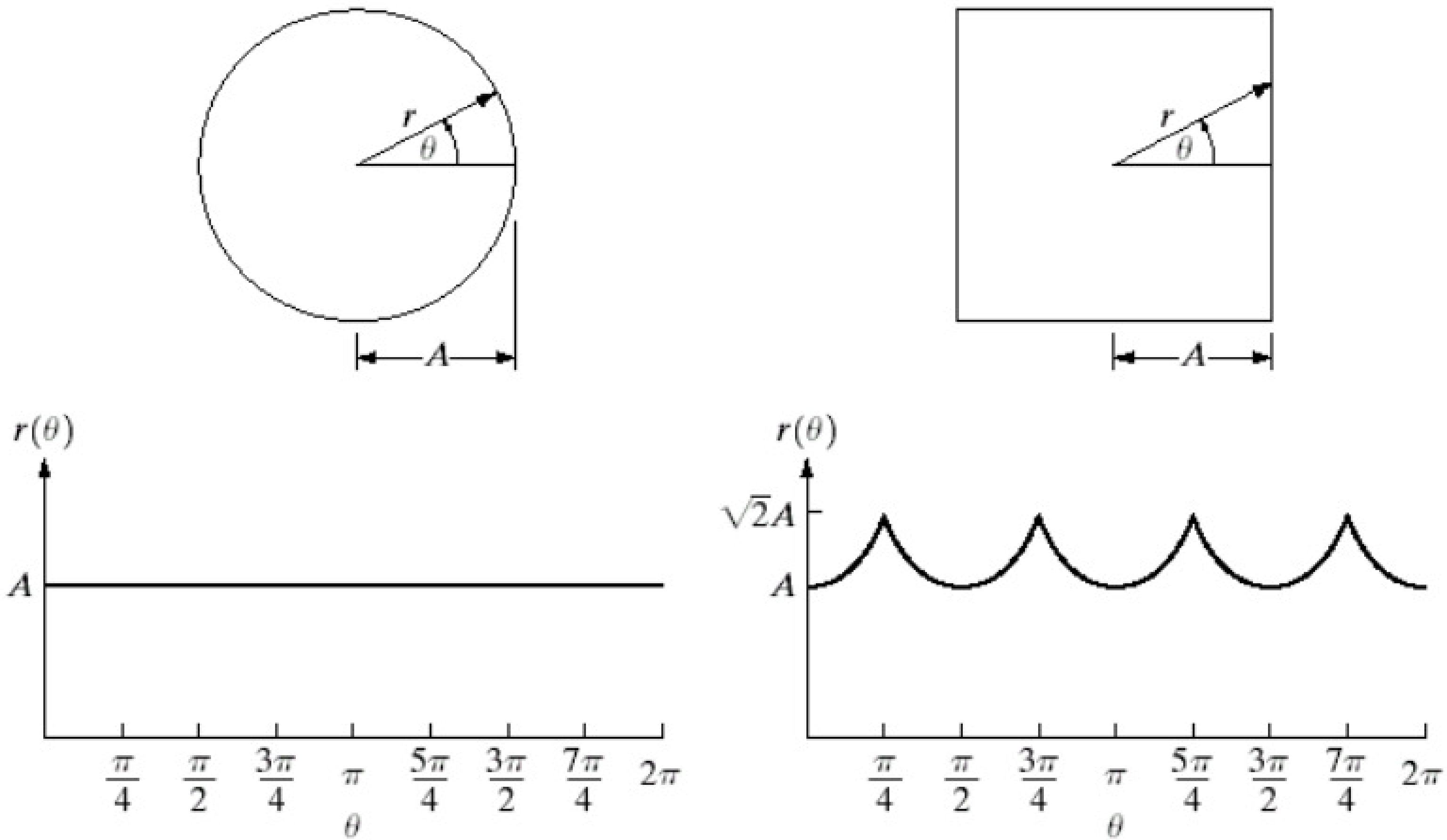
Polygonal Approximations

- Splitting techniques
  1. Start with an initial guess, e.g., based on majority axes
  2. Calculate the orthogonal distance from lines to all points
  3. If maximum distance > threshold, create new vertex there
  4. Repeat until no points exceed criterion



Signatures

Boundary representation



Signatures

Boundary representation

- A 1D representation of a boundary
- Could be implemented in different ways
  - Distance from centre point to border as a function of angle
  - Angle between the tangent in each point and a reference line (slope density function)
- Independent of translation, but not rotation & scaling. Possible solutions:
  - Select unique starting point (e.g. based on major axis)
  - Normalize amplitude of signature (divide by variance)

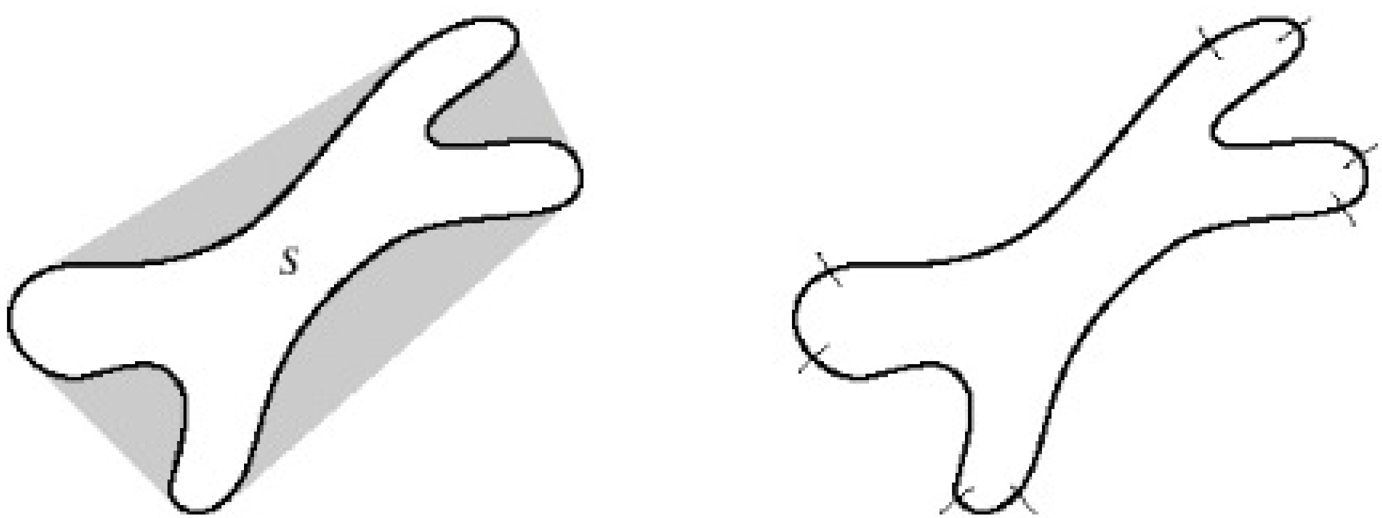
Boundary segments

Boundary representation

- When a boundary contains major concavities that carry shape information it can be worthwhile to decompose it into segments
- A good way to achieve this is to calculate the convex Hull of the region enclosed by the boundary
- Can be a bit noise sensitive
  - Smooth prior to Convex hull calculation
  - Calculate Convex Hull on polygon approximation

Boundary segments

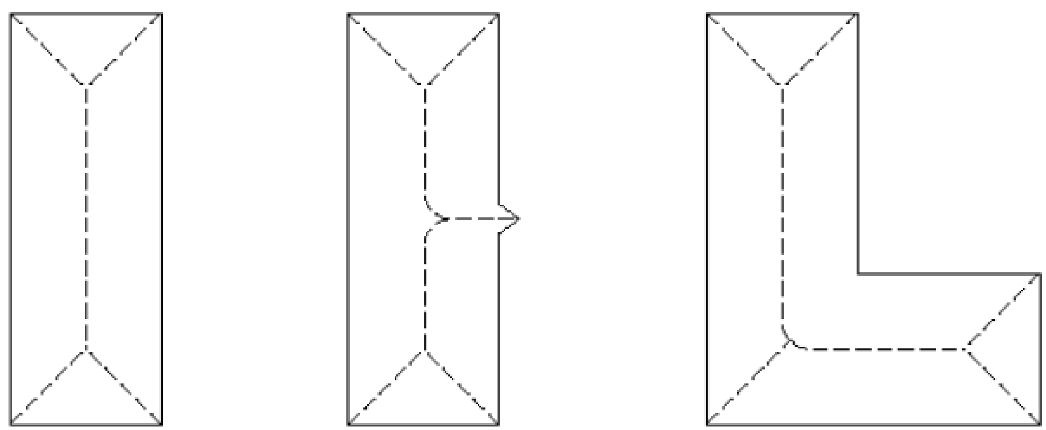
- Can be a bit noise sensitive
  - Smooth prior to Convex Hull calculation
  - Calculate Convex Hull on polygonal approximation



Skeletons

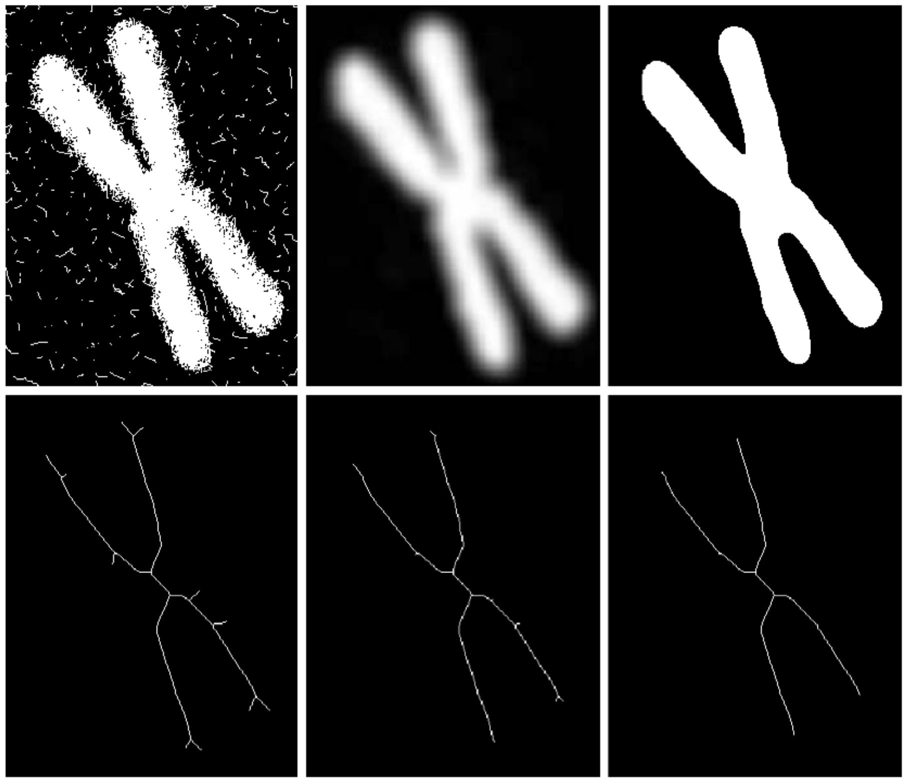
Shape representation

- Skeletons could be used as curve representations of an object
- Should in general be thin, centered, topologically equivalent to original object and reversible



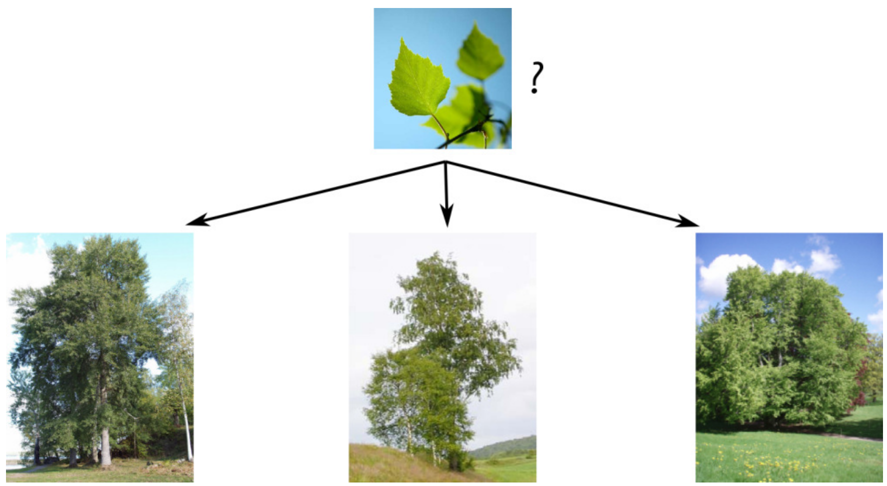
Skeletons

- Example:



Descriptors

- We have now represented our objects in different ways (using boundary representation and skeletons)
- The next step is to **describe** our regions so that we later can **classify** them (next lecture)



Simple boundary descriptors

Boundary descriptors

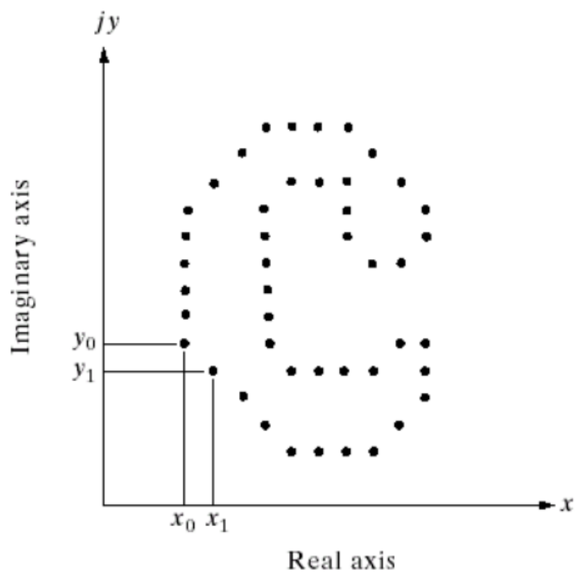
- Length (perimeter)
- Diameter = $\max_{i,j}\left[D(p_i,p_j)\right]$ = major axis
- Minor axis (perpendicular to major axis)
- Basic rectangle = major × minor
- Eccentricity = major / minor
- Curvature= rate of change of slope

Fourier descriptors

Boundary descriptors

- Redefine the x- & y-coordinates of the boundary as the real and imaginary parts respectively of a complex number
- Fourier transform of the new coordinates generates the Fourier descriptors
- Inverse transformation will regenerate the original image
- Doing an inverse transform on a part of the descriptors will result in an approximation of the shape

Fourier descriptors



- Represent the boundary as a sequence of coordinates
- Treat each coordinate pair as a complex number (2D → 1D)

$$s(k) = [x(k), y(k)]\,, k = 0, 1, 2, \ldots, K-1$$

$$s(k) = x(k) + iy(k)$$

Fourier descriptors

- From the DFT of the complex number we get the Fourier descriptors (the complex coefficients, a(u))

$$a(u) = \sum_{k=0}^{K-1} s(k) e^{-j2\pi uk/K}, u = 0, 1, 2, \ldots, K-1$$

- The IDFT from these coefficients restores s(k)

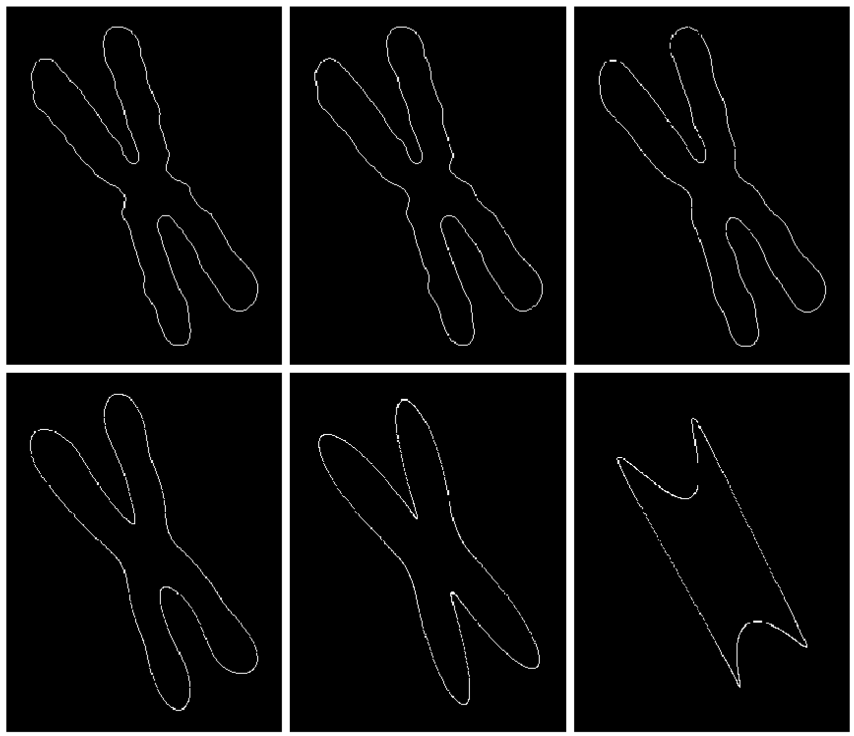
$$s(k) = \frac{1}{K} \sum_{u=0}^{K-1} a(u) e^{j2\pi uk/K}, k = 0, 1, 2, \ldots, K-1$$

- We can create an approximate reconstruction of s(k) if we use only the first P Fourier coefficients

$$\hat{s}(k) = \frac{1}{P} \sum_{u=0}^{P-1} a(u) e^{j2\pi uk/K}, k = 0, 1, 2, \ldots, K-1$$

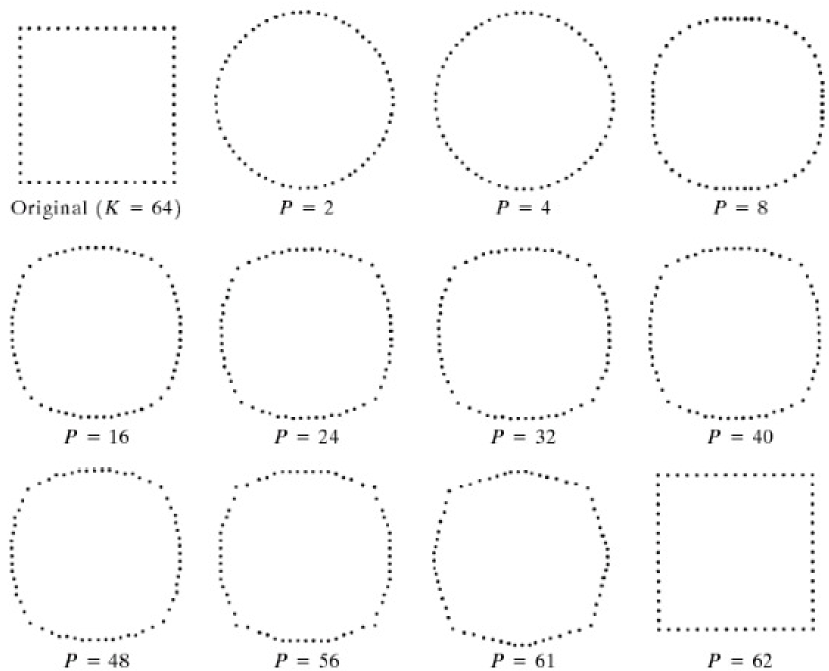
Fourier descriptors

- Boundary reconstruction using 546, 110, 56, 28, 14 and 8 Fourier descriptors out of a possible 1090.



Fourier descriptors

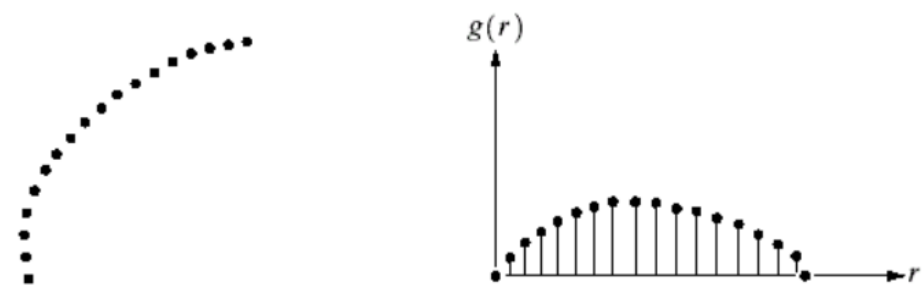
- This boundary consists of 64 point, P is the number of descriptors used in the reconstruction



Statistical moments

Boundary descriptors

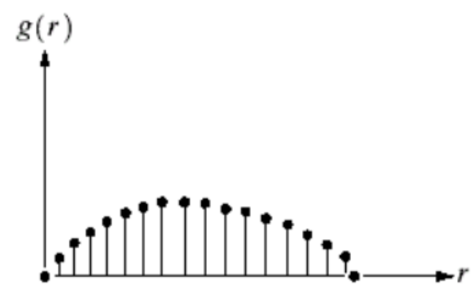
- Useful for describing the shape of boundary segments (or other curves)
- Suitable for describing the shape of convex deficiencies
- The histogram of the function (segment curve) can also be used for calculating moments
  - 2^{nd} moment gives spread around mean (variance)
  - 3^{rd} moment gives symmetry around mean (skewness)



Statistical Moments

- If v is a discrete random variable representing discrete amplitude in the range [0,A-1] then the nth statistical moment of v (about its mean) is calculated as:

$$\mu(v) = \sum_{i=0}^{A-1} (v_i - m)^n p(v_i), m = \sum_{i=0}^{A-1} v_i p(v_i)$$



Simple Regional Descriptors
Regional descriptors

- Area = number of pixels in a region
- Compactness (P2A) = perimeter^2 / area
- Circularity ratio = 4×π×area / perimeter^2
- Graylevel measures
  - Mean
  - Median
  - Max
  - Etc.

Topological descriptors

Regional descriptors

- Topology = The study of the properties of a figure that are unaffected by any deformation
- Topological descriptors
  - Number of holes in a region, H
  - Number of connected components, C
  - Euler number, E = C – H

A B

Topological descriptors

Regional descriptors

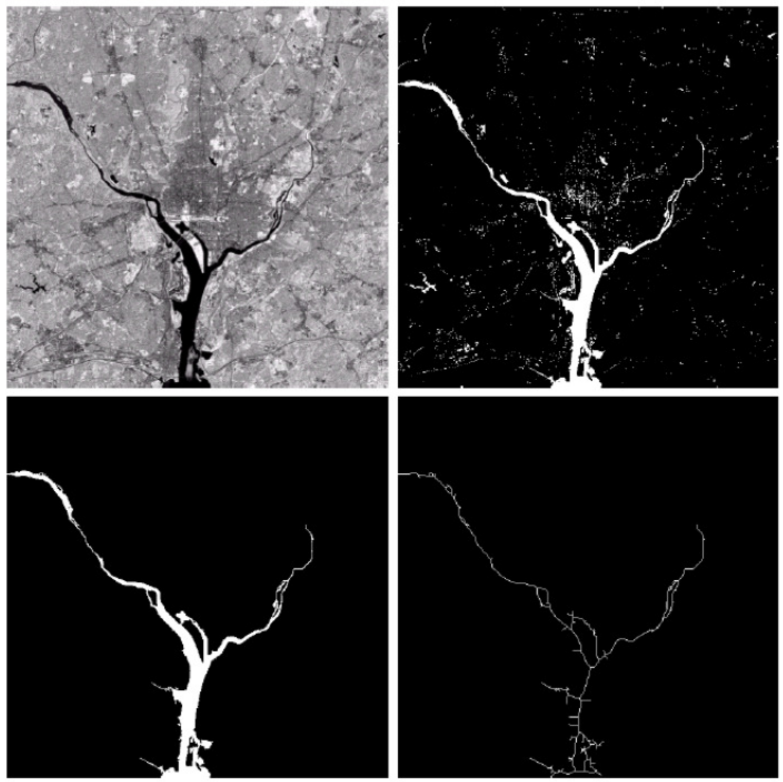
- Using connected components

IR image

Thresholded image

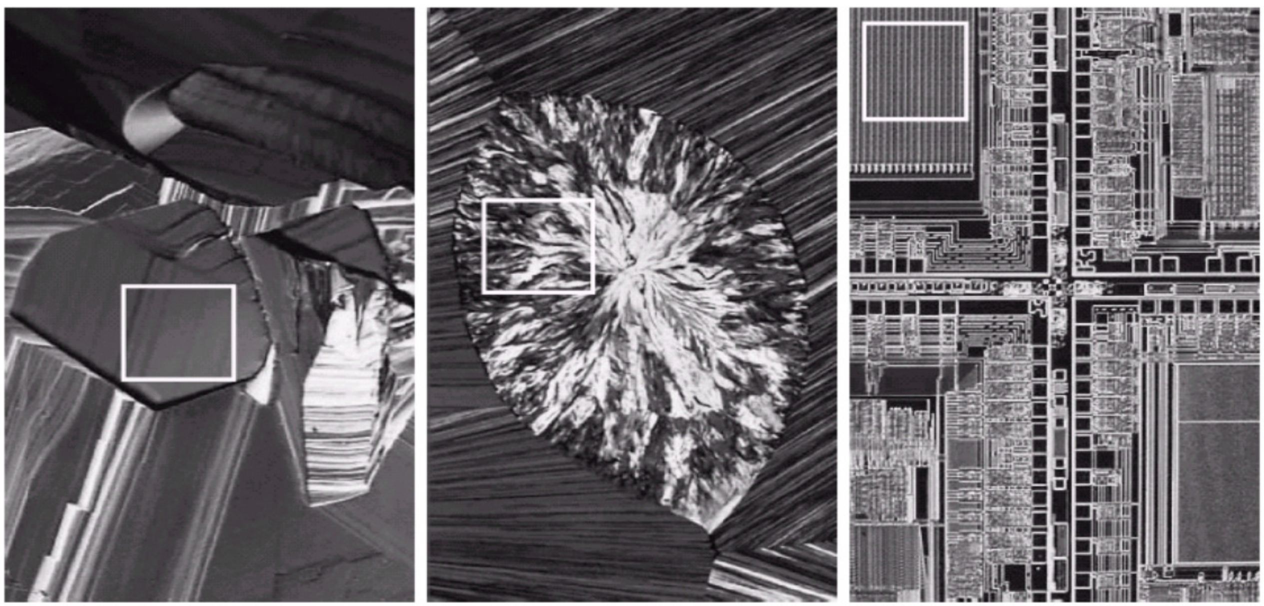
Largest connected component

Skeleton



Texture
Regional descriptors

- Textures can be very valuable when describing objects
- Example below: Smooth, coarse and regular textures



Texture

- Statistical texture descriptors:
  - Histogram based
  - Co-occurence based

(Statstical moments, Uniformity, entropy,...)

- Spectral texture descriptor
  - Use fourier transform

Histogram based descriptors
Regional descriptors

- Properties of the graylevel histogram, of an image or region, used when calculating statistical moments
  - z : discrete random variable representing discrete graylevels in the range [0 , L-1]
  - $P(z_i)$: normalized histogram component corresponding to the i^{th} value of z

$$\mu_n(z) = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i), \quad m = \sum_{i=0}^{L-1} z_i p(z_i)$$

2nd moment : Variance of z

3rd moment : Skewness

4th moment : Relative flatness

Histogram based descriptors

Uniformity and average entropy also uses z & $P(z_i)$

- Uniformity (maximum for image with just one grayvalue):

$$U = \sum_{i=0}^{L-1} p^2(z_i)$$

- Average entropy (measure of variability. Is 0 for constant images)

$$e = -\sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i)$$

Co-occurrence matrix
Regional descriptors

- For an image with N graylevels, and P, a positional operator, generate A, a N × N matrix, where $a_{i,j}$ is the number of times a pixel with graylevel value z_i is in relative position P to graylevel value z_j

- Divide all elements in A with the sum of all elements in A. This gives a new matrix C where $c_{i,j}$ is the probability that a pair of pixels fulfilling P has graylevel values z_i and z_j which is called the co-occurrence matrix

Co-occurrence Matrix

$P =$ "one pixel down"

2	0	1	0
1	1	2	1
2	0	1	2
1	1	0	1

$$\mathbf{A}_1 = \begin{pmatrix} 0 & 2 & 0 \\ 3 & 0 & 4 \\ 0 & 3 & 0 \end{pmatrix}, \quad \mathbf{C}_1 = \begin{pmatrix} 0 & \frac{1}{6} & 0 \\ \frac{1}{4} & 0 & \frac{1}{3} \\ 0 & \frac{1}{4} & 0 \end{pmatrix}$$

1	0	1	2
1	0	2	1
0	1	2	1
2	1	0	1

$$\mathbf{A}_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \mathbf{C}_2 = \begin{pmatrix} \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{3} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \end{pmatrix}$$

Co-occurrence Matrix Descriptors

- Maximum probability (strongest response to P)

$$max_{i,j}(c_{ij})$$

- Uniformity

$$\sum_i \sum_j c_{ij}^2$$

- Entropy (randomness)

$$-\sum_i \sum_j c_{ij} \log_2 c_{ij}$$

Co-occurrence Matrix

- Example 1
  - Maximum probability = 1/3
  - Uniformity ≈ 0.264
  - Entropy ≈ undefined

$$\mathbf{C}_1 = \begin{pmatrix} 0 & \frac{1}{6} & 0 \\ \frac{1}{4} & 0 & \frac{1}{3} \\ 0 & \frac{1}{4} & 0 \end{pmatrix}$$

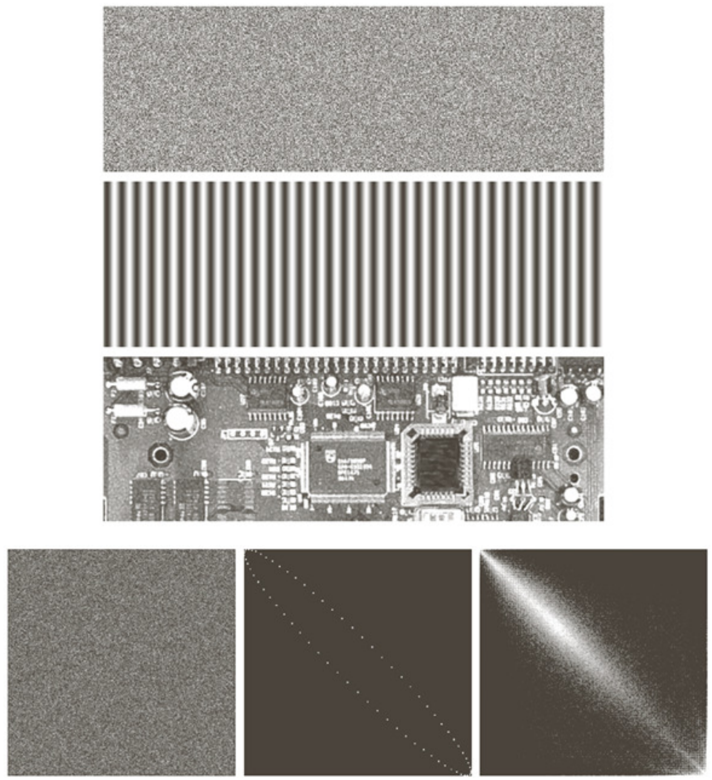
Co-occurrence Matrix

- Example 2
  - Maximum probability = 1/3
  - Uniformity ≈ 0.167
  - Entropy ≈ 2.918

$$\mathbf{C}_2 = \begin{pmatrix} \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{3} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \end{pmatrix}$$

Co-occurrence Matrix

- Examples: match image to co-occurrence matrix



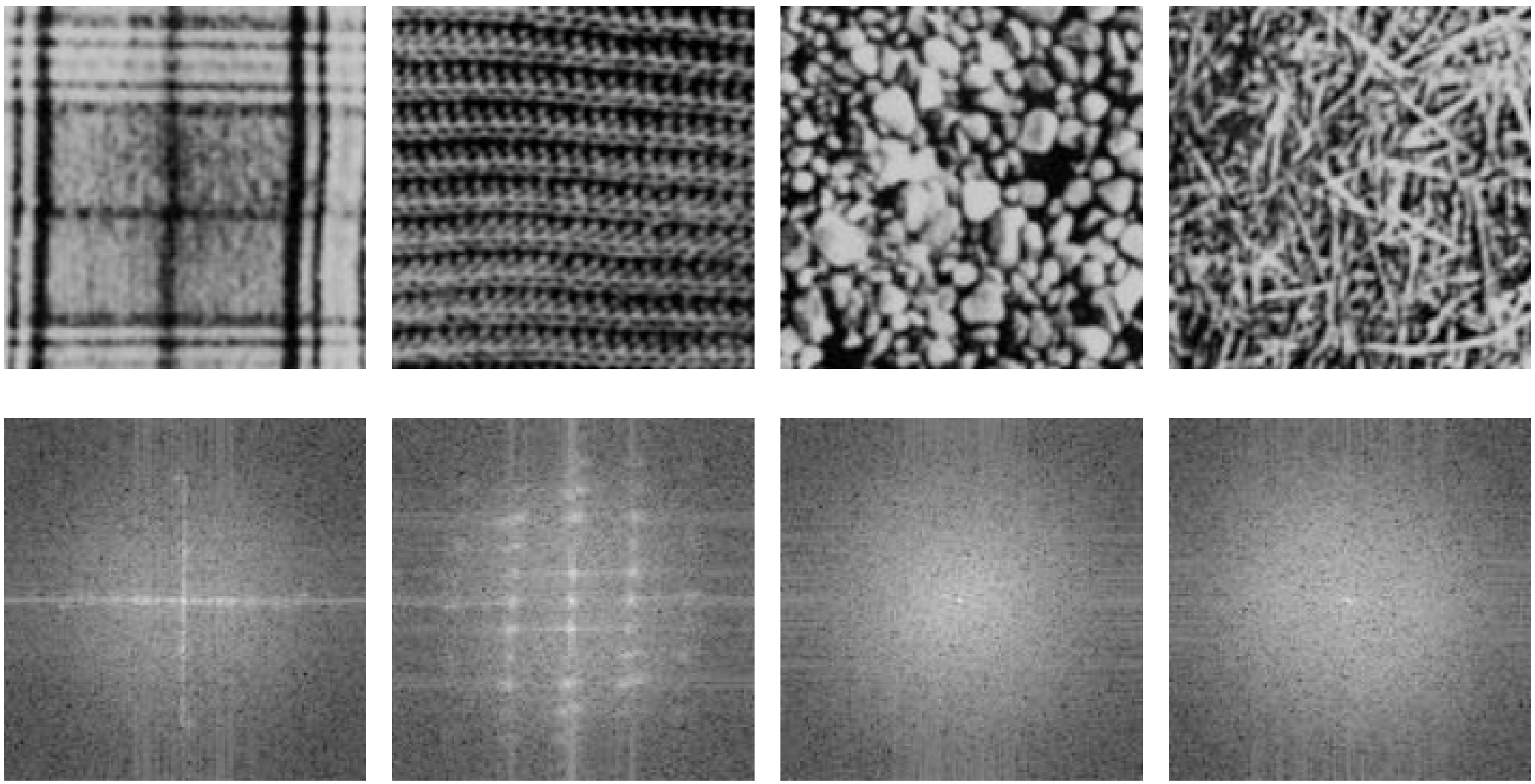
Spectral Texture

Regional descriptors

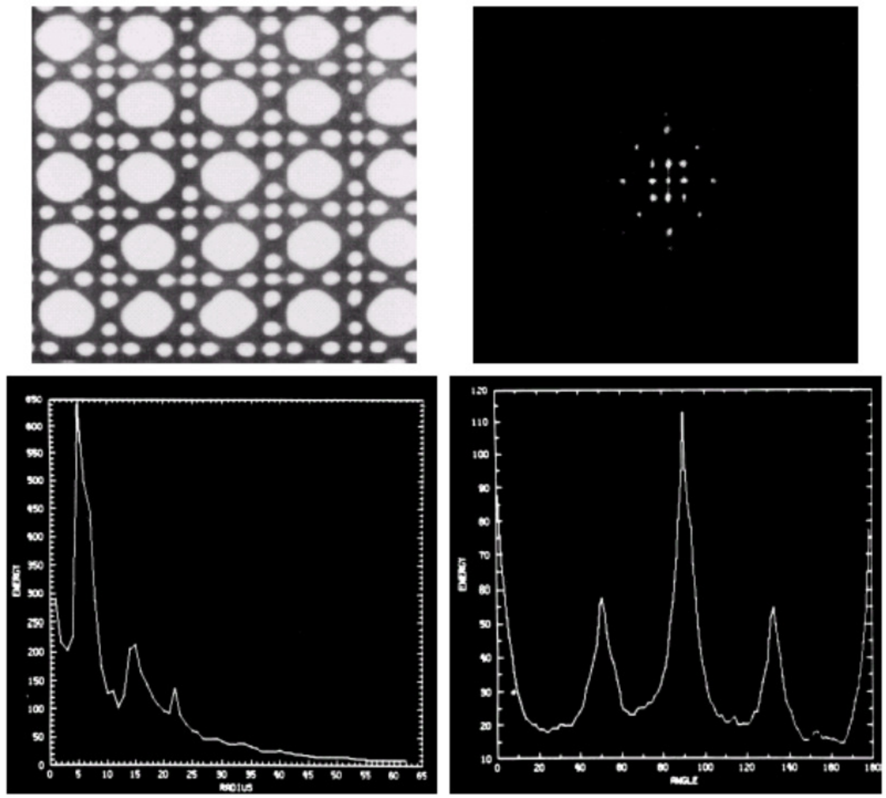
- Peaks in the Fourier spectrum give information about direction and spatial period patterns
- The spectrum can be described using polar coordinates $S(r,\theta)$
- For each angle θ, $S(r,\theta)$ is a 1D function $S_\theta(r)$
- Similarly, for each frequency r, $S_r(\theta)$ is a 1D function
- A global description can be obtained by summing $S_\theta(r)$ and $S_r(\theta)$

$$S(r) = \sum_{\theta=0}^{\pi} S_\theta(r), \quad S(\theta) = \sum_{r=1}^{R} S_r(\theta)$$

Spectral Analysis
Regional descriptors



Spectral Analysis



S(r) S(θ)

Moment Invariants
Regional descriptors

- For a 2D continuous function f(x,y), the moment of order (p + q) is defined as

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x,y)\, dx\, dy$$

for p, q = 0,1,2,…

- The central moments are defined as

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y)\, dx\, dy$$

where

$$\bar{x} = \frac{m_{10}}{m_{00}} \quad \text{and} \quad \bar{y} = \frac{m_{01}}{m_{00}}$$

Moment Invariants

- If f(x,y) is a digital image, the central moments become

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y)$$

- The normalized central moments, denoted η_{pq}, are defined as

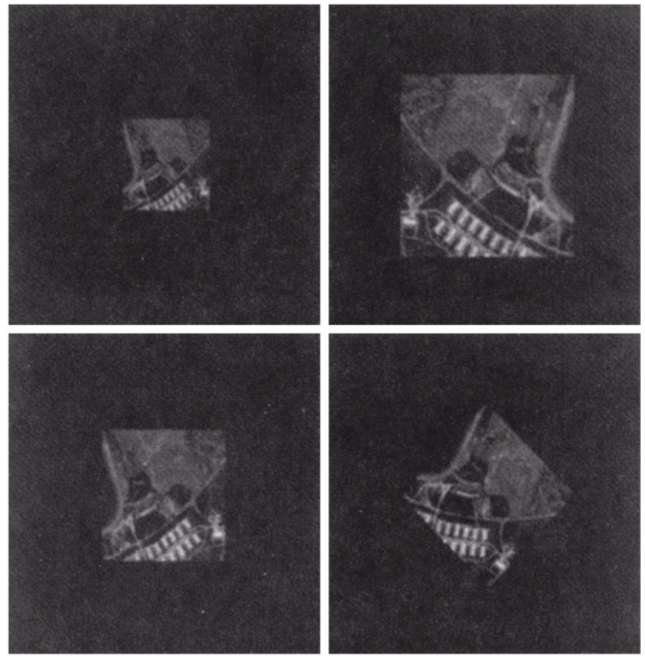
$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}$$

where $\gamma = \frac{p+q}{2} + 1$ for p+q = 2,3,…

Moment Invariants

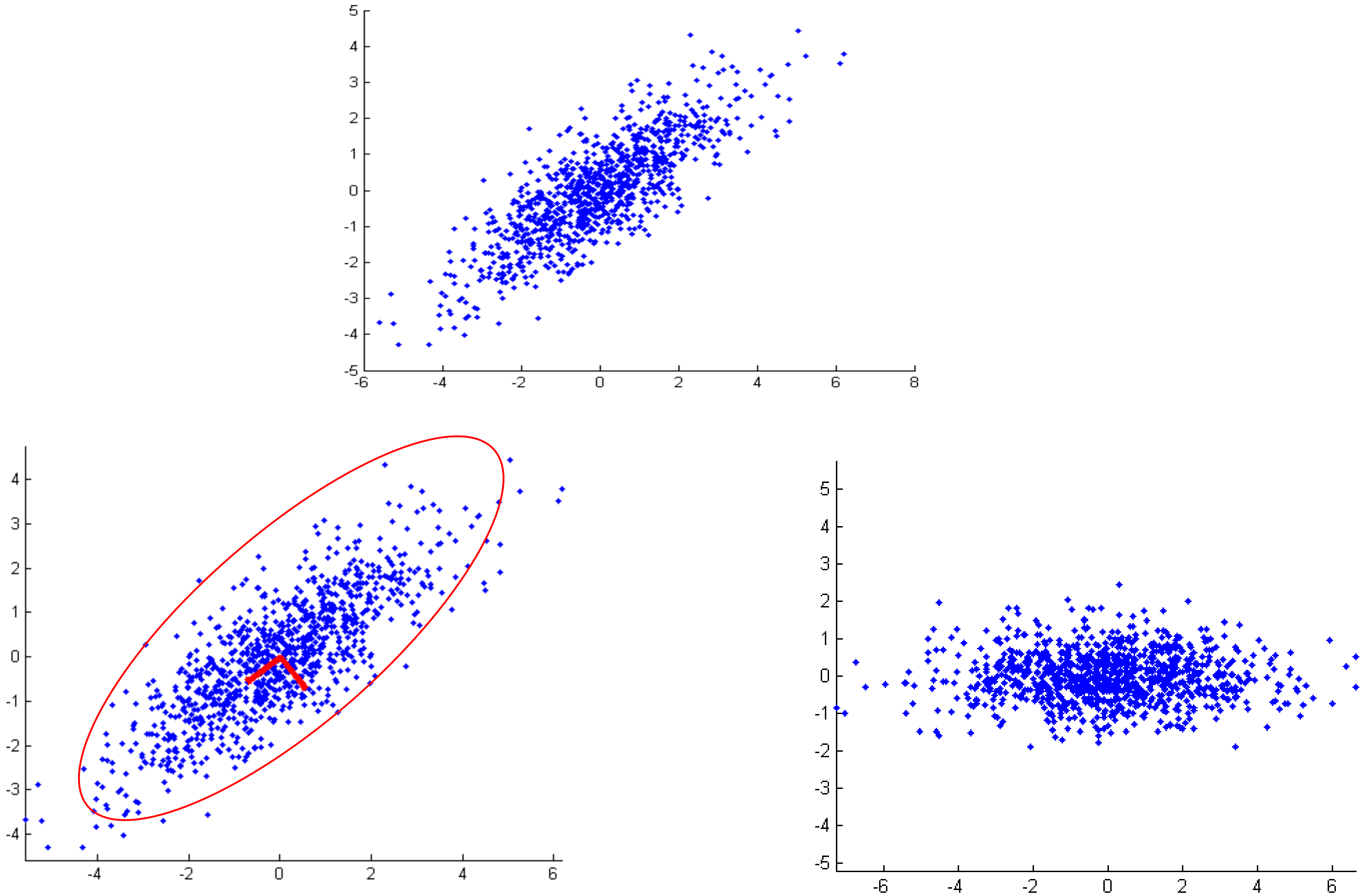
- A set of seven invariant moments can be derived from the 2nd and 3rd moments
- These moments are invariant to changes in translation, rotation and scale

$$\phi_1 = \eta_{20} + \eta_{02}$$
$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$
$$\phi_3 = (\eta_{30} - \eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$
$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$
$$\ldots \text{(see textbook)}$$



Invariant (Log)	Original	Half Size	Mirrored	Rotated 2°	Rotated 45°
ϕ_1	6.249	6.226	6.919	6.253	6.318
ϕ_2	17.180	16.954	19.955	17.270	16.803
ϕ_3	22.655	23.531	26.689	22.836	19.724
ϕ_4	22.919	24.236	26.901	23.130	20.437
ϕ_5	45.749	48.349	53.724	46.136	40.525
ϕ_6	31.830	32.916	37.134	32.068	29.315
ϕ_7	45.589	48.343	53.590	46.017	40.470

Principal components Analysis



Matlab code

```
m = [0 0];
Sigma = [3 2;2 2]; R = chol(Sigma);
z = repmat(m,1000,1) + randn(1000,2)*R;
c=cov(z);
figure,scatter(z(:,1),z(:,2),'.');
c=cov(z);
[E,D]=eig(c)
figure,scatter(z(:,1),z(:,2),'.');
hold on
quiver(0,0,E(1,1),E(2,1),'r','LineWidth',4);
axis('equal');
hold on
quiver(0,0,E(1,2),E(2,2),'r','LineWidth',4);
axis('equal');

[E,D]=eig(c)
[pc score latent]=princomp(z);
figure,scatter(score(:,1),score(:,2),'.');
axis('equal');
```

Principal components Analysis

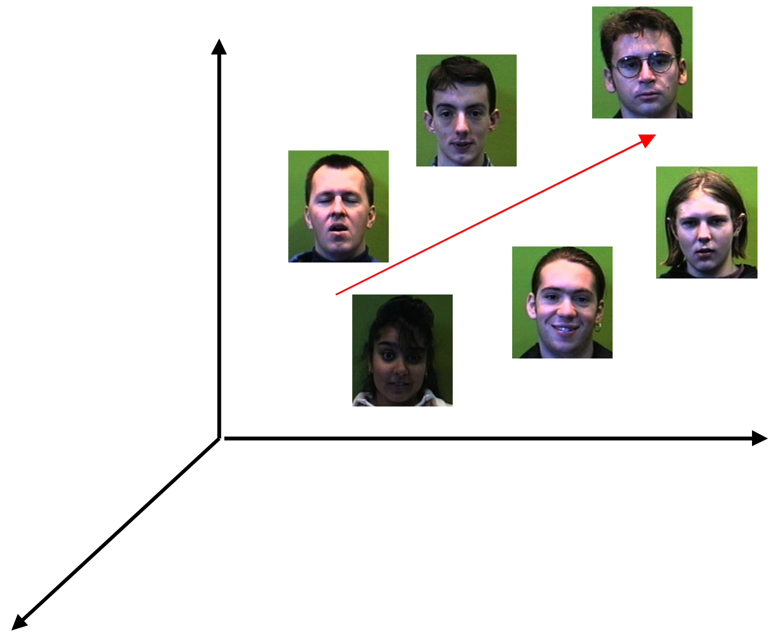
- Calculate $\mathbf{C}_x$, covariance matrix of data $\mathbf{X}$
- Find eigenvectors and corresponding eigen values of covariance matrix ($\mathbf{C}_x\mathbf{e}_i = \lambda_i\mathbf{e}_i$)
- Find A which is a matrix with the eigenvectors as rows, ordered corresponding to decreasing eigenvalue
- Use A to transform $\mathbf{x}$ to $\mathbf{y}$: $\mathbf{y} = A(\mathbf{x} - \mathbf{m}_x)$.
- Any vector $\mathbf{x}$ can be recovered from $\mathbf{y}$ by: $\mathbf{x} = A^T\mathbf{y} + \mathbf{m}_x$ and approximated by only using some (say k) of the eigenvalues and an A_k matrix constructed from the k eigenvectors

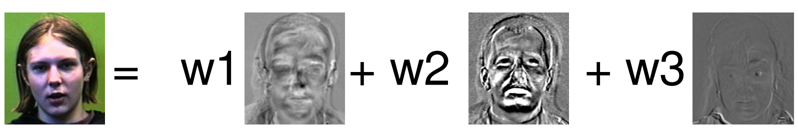
Face recognition Using PCA

We need a training data set: bunch of sample images of people we want to recognize

Using PCA analysis we find eigenfaces (eigenvectors)

Every new image that we have, we project the Image on eigenvectors and based on the weights we obtained we can classify it.





http://www.mathworks.co.uk/matlabcentral/fileexchange/17032-pca-based-face-recognition-system

Conclusion

- Boundary representation (chain code, polygonal, signature, convex hull)
- Shape represent (skeleton)
- Boundary descriptor (Fourier descriptor , statistical descriptor)
- Regional descriptor (Histogram, Texture)
- PCA

Exercise 11.19 and 11.25