Digital Image Processing

8 February 2007

Dr. ir. Aleksandra Pizurica
Prof. Dr. Ir. Wilfried Philips

Aleksandra.Pizurica @telin.UGent.be
Tel: 09/264.3415

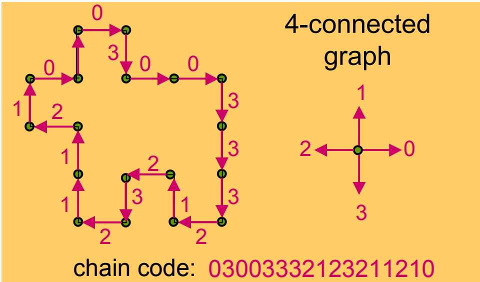
UNIVERSITEIT GENT
Telecommunicatie en Informatieverwerking

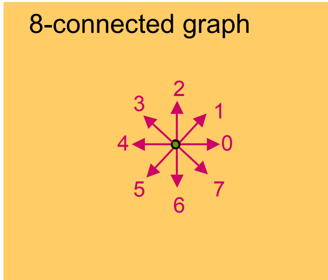
INSTITUUT VOOR PERMANENTE VORMING

Image representation

Chain codes…

- Chain codes represent a boundary by a connected sequence of straight-line segments of specified length and direction.
- Typically this representation is based on a 4- or 8-connectivity
- The direction of each segment is coded by numbers and the resulting chain code is a sequence of numbers

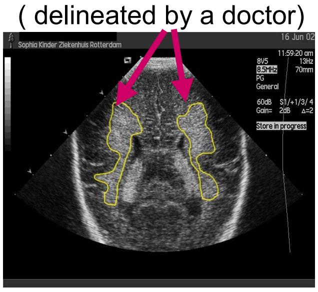




…Chain codes…

Sick regions

(delineated by a doctor)



ultrasound image

Can be interesting in cases where a number of contours need to be stored, like in medical follow-up examinations

- Advantage: a compressed contour representation
- Disadvantages:
  - chain code depends on the starting point
    → can be solved: treat the chain code as a circular sequence and redefining the starting point so that the resulting sequence of numbers is the smallest possible integer
  - Operations such as scaling and rotation result in different contours that in practice cannot be normalized (due to a finite grid) and hence in different chain codes.
    → this problem cannot be completely solved but its effect can be reduced by resampling to a coarser grid before chain coding and by a proper orientation of the resampling grid

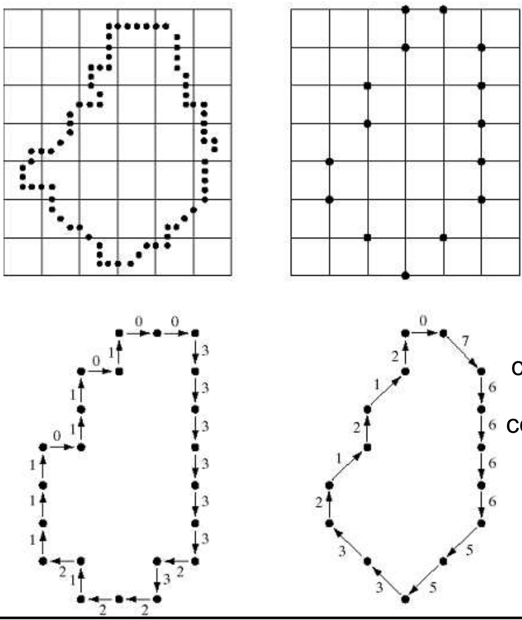
…Chain codes

a digital contour with resampling grid superimposed

resampled

chain code for 4-connectivity

chain code for 8-connectivity

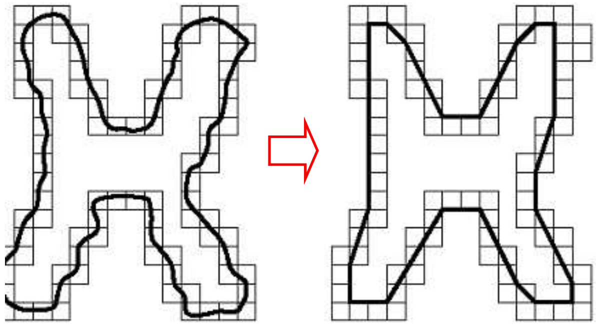


Poligonal approximations…

- Digital boundaries carry information which may be superfluous for certain applications.
- Boundary approximations can be sufficient in such cases.
- Linear piecewise (polygonal) approximations are the most frequently used ones
- These approximations produce a polygon which closely resembles original boundary line
- We will illustrate two examples of polygonal approximations called perimeter polygons and polygonal approximation by splitting

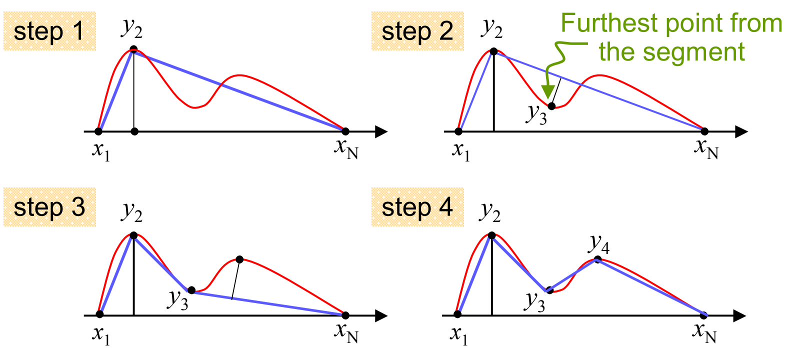
Minimum perimeter polygons

One type of polygonal approximation is minimum perimeter polygons
Think of the object boundary rubber band and imagine it is allowed to shrink until it is tightened.



Polygonal approximation by splitting…

- The optimal piecewise linear approximation means estimating the polygon vertices in such a way that total error is minimized. This can be a non-trivial and computationally intensive optimization approach!
- I In practice splitting techniques, which are fast are often used
- Splitting techniques divide a curve segment recursively until each curve segment is approximated with a linear segment with acceptable error. Mean square error or maximal error is used as a criterion;



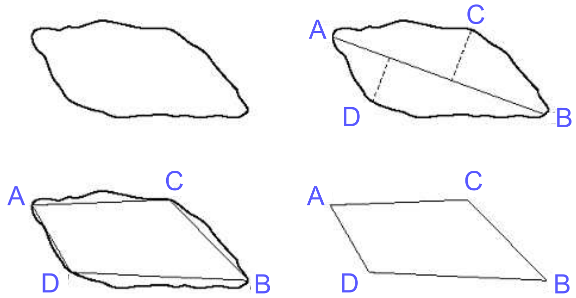
… Polygonal approximation by splitting

A practical splitting procedure can be:

step 1 Look for the two points that are furthest apart (A and B) and join them by a line segment AB

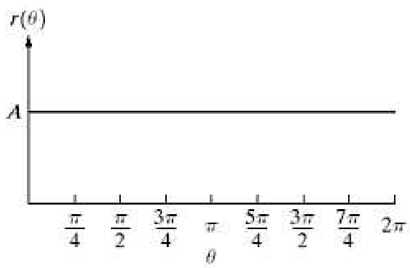
step 2 Look for the boundary points that are furthest apart from AB in both parts of the boundary (C and D)

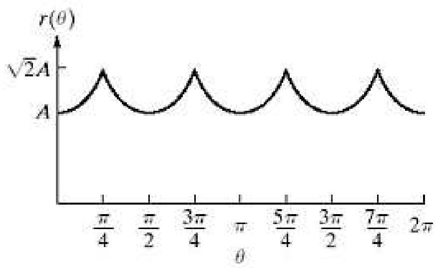
step 3 connect the points by line segments (AC, CB, BD, DA). For each segment check if the distance between its points and the corresponding points of the true boundary is smaller than a threshold. If yes, stop, otherwise subdivide the segment further.



Signatures

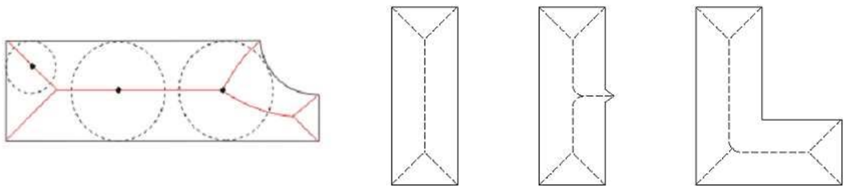
A signature is a 1-D functional representation of a boundary.





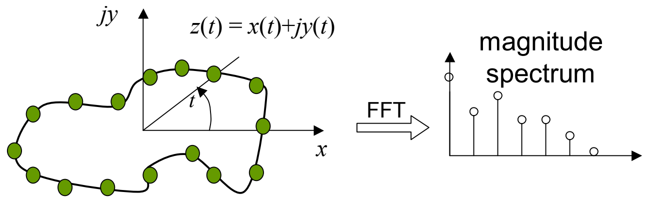
Skeletons

- A medial axis or skeleton representation is a popular tool in object recognition.
- Skeletons provide a compact and often highly intuitive representation.
- Shape skeletons have been used , e.g., in object recognition and representation, industrial inspection (e.g., inspection of printed circuit board) and in medical imaging.
- The skeleton of a region is often defined via the medial axis transformation (MAT).The MAT of a region R with border B is as follows: for each point p of R, we find its closest neighbor in B. If p has more then one such points, it is said to belong to the medial axis (skeleton) of R.



Boundary descriptors

Fourier descriptors (FDs)



A closed curve can be described by the curve coordinates $x(t)$, $y(t)$. The waveformes $x(t)$, $y(t)$ are periodic with period 2π. The waveforms can be sampled and combined to produce a complex periodic waveform with period N

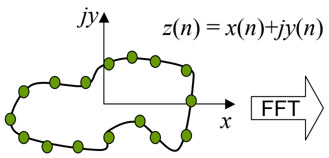
$$z(n)=x(n)+jy(n) \qquad n=0,1,\ldots,N-1$$

The Fourier transform coefficients of this signal are the Fourier descriptors of the curve

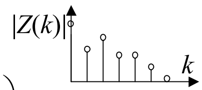
$$Z(k)=\sum_{n=0}^{N-1} z(n)\exp\left(-\frac{j2\pi nk}{N}\right)$$

Matlab demo fd1

Some properties of Fourier descriptors



$$Z(k)=\sum_{n=0}^{N-1} z(n)\exp\left(-\frac{j2\pi nk}{N}\right)$$



$Z(0)$ represents the center of gravity of the curve

Translation $z_t(n) = z(n)+z_0$, $z_0= x_0+jy_0$ affects only $Z(0)$: $Z_t(0)=Z(0)+z_0$

Rotation $z_r(n) = z(n)e^{j\theta}$ → phase shift of the coefficients $Z_r(0)=Z(k)\, e^{j\theta}$

Scaling $z_s(n) = az(n)$ → scaling the coeffs. by same amount $Z_s(0)=aZ(k)$

Change starting point $z_t(n) = z(n_1\text{-}n_0)$ → modulation $Z_t(k)=Z(k)e^{-j2\pi n_0 k/N}$

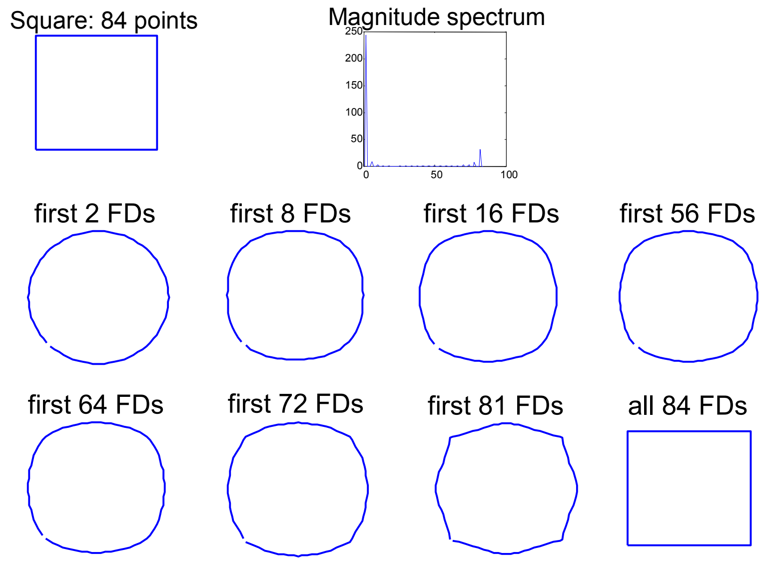
The coefficient magnitude $|Z(k)|$ is rotation invariant and the magnitude of the N-1 coefficients $|Z(k)|$, $k=1,\ldots,N-1$ is translation invariant as well.

$$E=\sum_{k=0}^{N-1}(Z_1(k)-Z_2(k))^2$$

Can be used as error measure for matching curves z_1(t) and z_2(t)

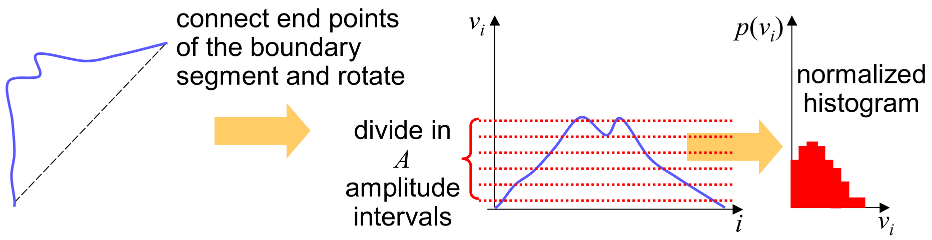
Matlab demo fd2

Example: Shape reconstruction from FDs



Matlab demo fd3

Statistical moments…



The shape of a boundary can now be described by simple moments such as mean, variance,… The n-th moment of v about its mean m is:

$$\mu_n(v) = \sum_{i=0}^{K-1} (v_i - m)^n p(v_i)$$

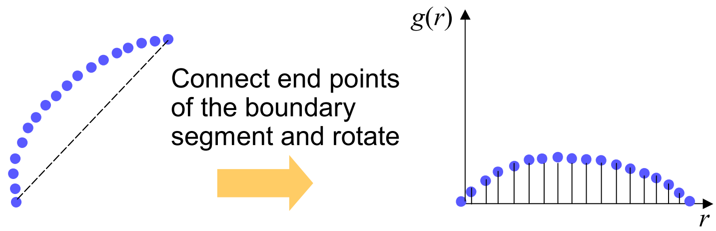
$$m = \sum_{i=0}^{A} v_i p(v_i)$$

$p(v_i)$ is the probability of occurrence of value v_i. These are estimated by the values of the normalized histogram (normalized = area under is 1)

Using only first few moments is usually sufficient to differentiate between (clearly distinct) different shapes

…Statistical moments

An alternative approach is to normalize (scale) the samples after rotation such that the total area equals 1 and to treat this as a histogram.



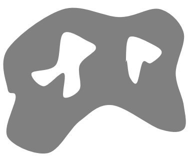
The moments are now defined as

$$\mu_n(v) = \sum_{i=0}^{K-1} (r_i - m)^n g(r_i) \qquad \text{with the mean} \qquad m = \sum_{i=0}^{K-1} r_i g(r_i)$$

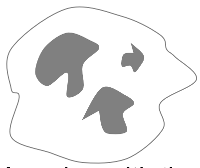
Now K is the number of points on the boundary, and the n-th moment $\mu_n(r)$ is directly related to the shape of $g(r)$. The second moment measures the spread of the curve and the third moment measures its symmetry.

Topological descriptors

Topological properties are used for higher-level and global descriptions of regions in the image. An example of a topological property is the number of "holes" or the number of connected components in the region.

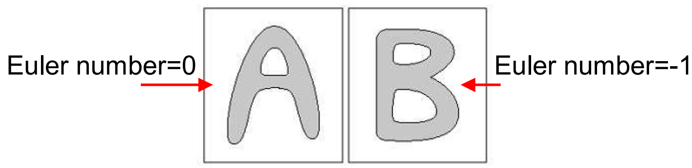


A region with two holes



A region with three connected components

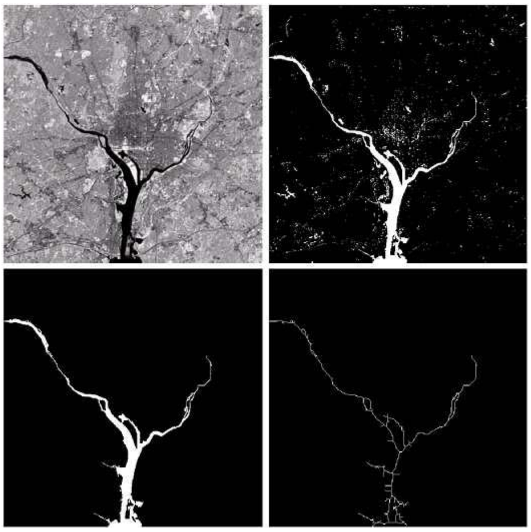
One topological descriptor is Euler number E, defined in terms of the number of holes H and the number of the connected components C as: E=C-H



An example of an image representation

original image

Thresholded image

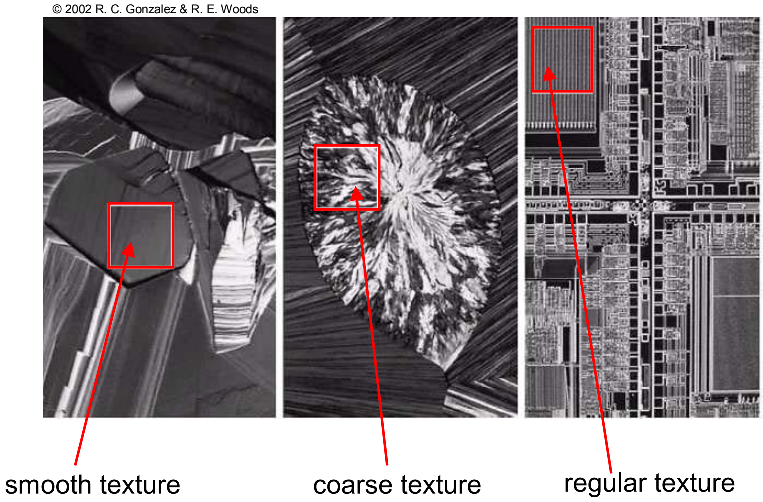


The largest connected component

skeleton

Texture representation

Texture descriptors



Statistical texture descriptors…

One of the simplest approaches is using statistical moments of the gray level histogram. Let z denote gray levels and $p(z_i)$, I=0,1,…,L-1 the corresponding histogram, where L is the number of distinct gray levels. The n-th moment of z about the mean m is

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

Note $\mu_0 = 1$ and $\mu_1 = 0$.

The second moment (the variance $\mu_2(z)=\sigma^2(z)$) is often used in texture description. The third moment $\mu_3(z)$ describes the skewness (asymmetry) of the histogram and the fourth moment $\mu_4(z)$ its relative flatness. Other useful texture descriptors based on the histogram include

- a measure of uniformity $U = \sum_{i=0}^{L-1} p^2(z_i)$
- average entropy $e = -\sum_{i=0}^{L-1} p(z_i)\log_2(z_i)$

… Statistical texture descriptors

Histogram based descriptors are limited in the sense that they cannot express information about relative positions of pixel values with respect to each other. One approach to solve this is to use a two-dimensional histogram called co-occurrence matrix.

The co-occurrence matrix counts the number of grey value transitions in a given direction and at a given distance d

Example:

$$\begin{matrix} 0 & 0 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 & 1 \\ 2 & 2 & 1 & 0 & 0 \\ 1 & 1 & 0 & 2 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{matrix}$$

Co-occurrence matrix for d=1, horizontal right

$$C = \begin{bmatrix} 4 & 3 & 1 \\ 4 & 3 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Some useful descriptors derived from the co-occurrence matrix

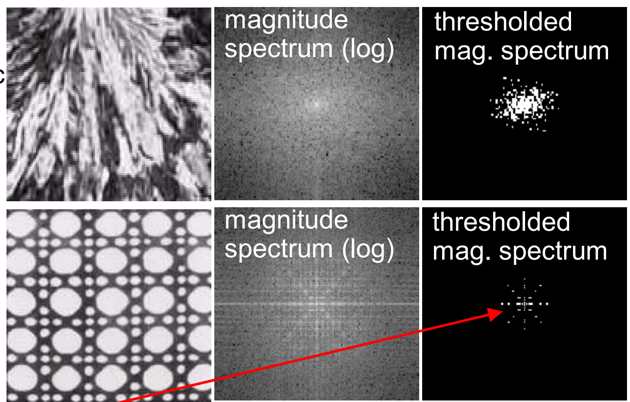
Maximimum probability	$\max_{i,j}(c_{i,j})$	uniformity	$\sum_i \sum_j c_{i,j}^2$
contrast	$\sum_i \sum_j (i-j)c_{i,j}$	entropy	$-\sum_i \sum_j c_{i,j} \log_2 c_{i,j}$

Spectral texture descriptors…

Spectral approaches are powerful for describing periodic 2D patterns

non-periodic texture

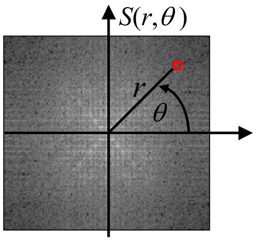
periodic texture



Prominent peaks of the Fourier spectrum indicate the principal direction of the texture patterns and their fundamental spatial period

Periodic components can be removed by filtering the spectrum, leaving non-periodic elements, which can be described by by statistical techniques.

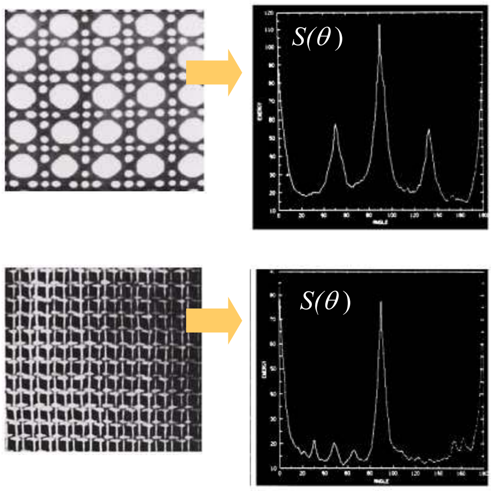
…Spectral texture descriptors



From the polar representation $S(r,\theta)$ of the spectrum, the following 1-D features can be derived

$$S(r) = \sum_{\theta=0}^{\pi} S_\theta(r)$$

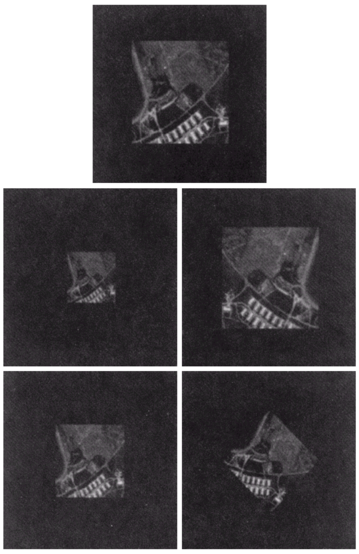
$$S(\theta) = \sum_{r=1}^{R_0} S_r(\theta)$$



Moments of 2D functions

Use of moments

- Suppose the following problem: we have a prototype of a scene (or an object, face,…) that we want recognize in various parts of an image sequence.
- This object or a scene can be in general deformed and rotated and scaled in various parts of the image sequence.
- In order to match this object with the prototype we need to extract certain features that are invariant under certain types of degradations and geometric transformations.
- Moments offer features that are robust and invariant under transformations such as translation, rotation, scaling and mirroring.



Moments of 2D functions

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

$$m_{p,q} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y)dxdy \qquad p,q=0,1,2,\ldots$$

Under certain conditions that are fulfilled by most of the 2-D functions of practical interest, the sequence of all moments $\{m_{p,q}\}$ of a function $f(x,y)$ is uniquely determined by $f(x,y)$ and the sequence $\{m_{p,q}\}$ uniquely determines $f(x,y)$

The central moments are defined as

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

where

$$\bar{x} = \frac{m_{1,0}}{m_{0,0}} \quad \text{and} \quad \bar{y} = \frac{m_{0,1}}{m_{0,0}}$$

The central moments describe the shape independent of translation

Moments of a digital image

For digital images the moments are defined analogously to those of 2D continuous functions, but replacing the integrals by finite sums:

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

with $\bar{x} = \frac{m_{1,0}}{m_{0,0}}$ and $\bar{y} = \frac{m_{0,1}}{m_{0,0}}$

The normalized central moments $\eta_{p,q}$ are defined as

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^{\gamma}}$$

with

$$\gamma = \frac{p+q}{2} + 1$$

Idea: define a measure not affected by translation and scaling

Invariant moments for shape recognition

Seven invariant moments can be derived from the 2nd and 3rd normalized central moments and are invariant to translation, rotation and scaling

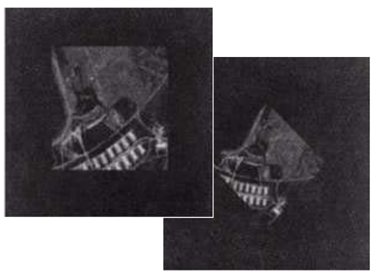
The first four of these invariant moments are

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - 3\eta_{03})^2$$

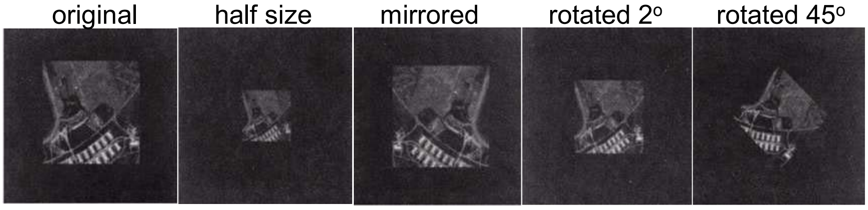
$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$



The expressions for ϕ_5, ϕ_6 and ϕ_7 are much longer (see, e.g., the book of Gonzales&Woods)

These invariant moments are useful when we need to recognize an object or a scene fragment that can be rotated, scaled or translated w.r.t. the prototype.

Invariant moments: Example



	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

Principal component analysis

Principal components analysis (PCA) is a technique for reducing the dimensionality of a multidimensional dataset.

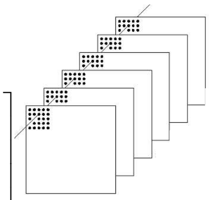
For example, suppose a set of 1000 hyperspectral images. These images look similar but differ from each other at some positions (i.e., some of the interesting details appear in different image at different places) Since the images are highly correlated we do not wish to spend time analyzing them all but rather rearranging the information into much fewer components that contain the interesting information.

PCA is an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

PCA is often called Hotteling transform or discrete Karhunen-Loève (KLT) transform

Calculating PCA

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ . \\ x_n \end{bmatrix}$$



Step 1: Calculate mean vector

$$\mathbf{m_x} = \frac{1}{L}\sum_{k=1}^{L}\mathbf{x}_k$$

and covariance matrix: transpose

$$\mathbf{C_x} = \frac{1}{L}\sum_{k=1}^{L}\mathbf{x}_k\mathbf{x}_k^T - \mathbf{m}_k\mathbf{m}_k^T$$

Step 2: Find eigen vectors of $\mathbf{C_x}$ (vectors **e** for which $\mathbf{C_x e} = \lambda \mathbf{e}$ where λ is a scalar, called eigen value)

Step 3: Form matrix $\mathbf{A}$ such that rows of this matrix are eigen vectors of $\mathbf{C_x}$ sorted such that the first row of $\mathbf{A}$ is the eigen vector corresponding to the largest eigen value and the last row of $\mathbf{A}$ is the eigen vector corresponding to the smallest eigen value

Step 4: Transform the data as $\mathbf{y} = \mathbf{A}(\mathbf{x} - \mathbf{m_x})$

Note: the covariance matrix of $\mathbf{y}$ is diagonal

$$\mathbf{C_y} = \begin{bmatrix} \lambda_1 & 0 & . & 0 \\ 0 & \lambda_2 & . & 0 \\ 0 & 0 & . & 0 \\ 0 & 0 & 0 & \lambda_n \end{bmatrix}$$

Principal component analysis: Example

input images resulting PCA componets