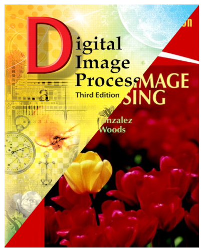
Lecture 7
Object Descriptors

Anders Brun
Uppsala University

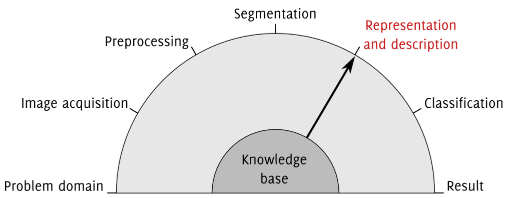
Reading instructions



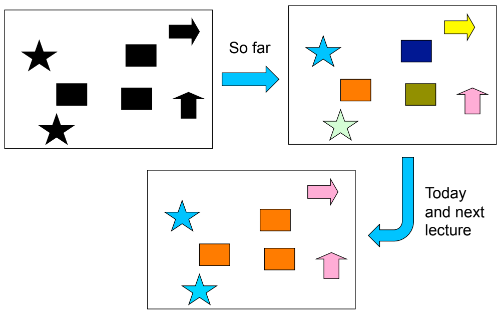
Chapter 11.1 – 11.4 in G-W

The Image Analysis Chain

- Our progress in the analysis chain



The Next Steps



Representations and descriptors

The next step after segmentation is to **represent** objects to make it possible to **describe** them

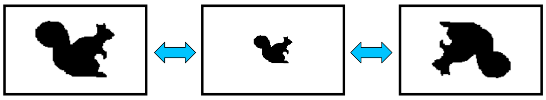
- External (boundary):
  - Representation: Polygon of the boundary
  - Description: The circumference
- Internal (regional)
  - Representation: Pixels inside the object
  - Description: The average color

Representations and descriptors

- The Representation of the Object
  - An encoding of the object
  - Truthful but possibly approximate
- A Descriptor of the Object:
  - Only an aspect of the object
  - Suitable for classification
  - Invariant to e.g. noise, translation, …
- (We can compute descriptors from the representation, but not vice versa)

Scale, rotation and translation

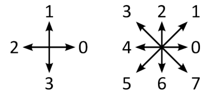
- Often we want descriptors that are invariant of scale, rotation and translation:

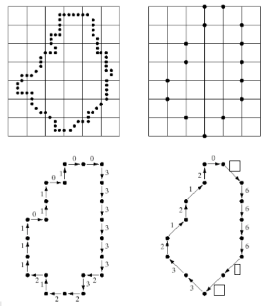


- However, not always. In Optical Character Recognition (OCR) rotation and scale is important (e.g. ‘P’ and ‘d’)

Chain Coding

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





Chain Coding

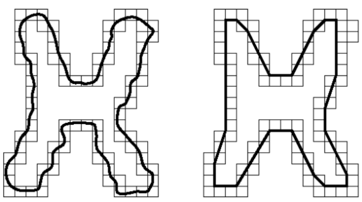
- Code become very long and noise sensitive
  - Use larger grid spacing, 0710 = 00
- Scale dependent
  - Choose appropriate grid spacing
- Start point determines result
  - Treat code as circular (minimum magnitude integer) 754310 → 075431
- Depends on rotation
  - Calculate difference code (counterclockwise)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

Polygonal Approximations

- 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

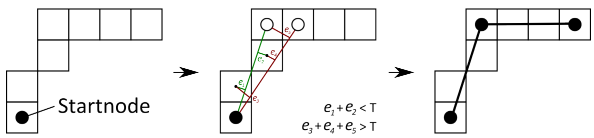
Polygonal Approximations

- **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



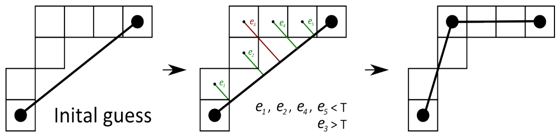
Polygonal Approximations

- **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

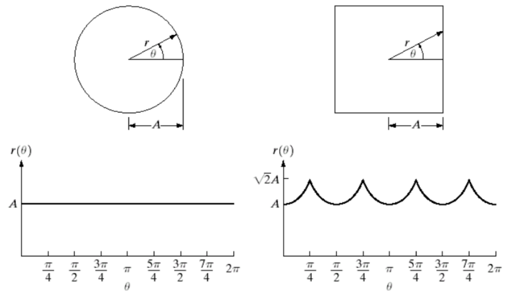


Polygonal Approximations

- **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



Signatures

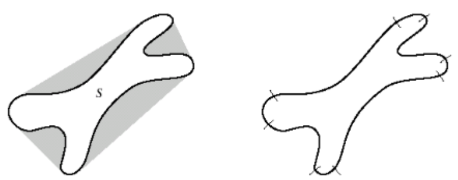
- A 1D representation of a boundary
- Could be implemented in different ways
  - Distance from centre point to border as a function of angle
  - Angle between the tangent in each point and a reference line (slope density function)
- Independent of translation, but not rotation & scaling. Possible solutions: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

Boundary segments

- 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

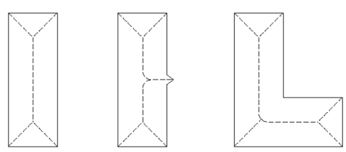
Boundary segments

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



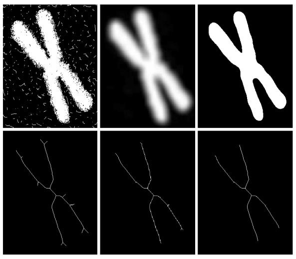
Skeletons

- 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

- After representation, the next step is to **describe** our regions so that we later can **classify** them (next lecture)
- A description is an aspect of the representation
- What descriptor is useful for classification of
  - adults / children
  - pears / bananas / tomatoes

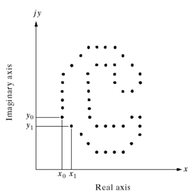
Simple boundary descriptors

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

Fourier descriptors

- Redefine the x- & y-coordinates of the boundary as the real and imaginary parts respectively of a complex number
  - f(k) = x(k) + i y(k), k = 0 … (K-1)
- The DFT generates the K Fourier descriptors
- Inverse transformation of the K descriptors regenerate the original boundary
- Inverse transformation of the P<K first descriptors generates an approximation of the boundary

Fourier descriptors



- Represent the boundary as a sequence of coordinates
- Treat each coordinate pair as a complex number

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

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

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)

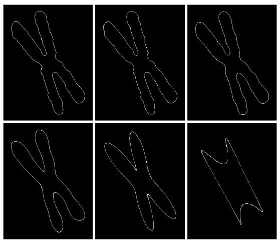
$$s(k) = \frac{1}{K} \sum_{u=0}^{K-1} a(u) e^{j2\pi uk/K}, k = 0, 1, 2, \ldots, K-1$$

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

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

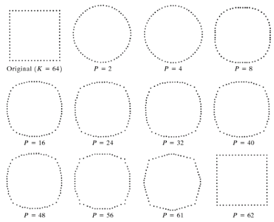
Fourier descriptors

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



Fourier descriptors

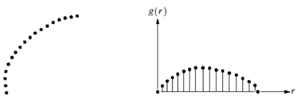
- This boundary consists of 64 point, P is the number of descriptors used in the reconstruction



Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University
UPPSALA UNIVERSITET SLU

Statistical moments

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

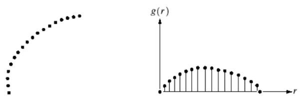


Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University
UPPSALA UNIVERSITET SLU

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:

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



Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University
UPPSALA UNIVERSITET SLU

Simple 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.

Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University
UPPSALA UNIVERSITET SLU

Topological descriptors

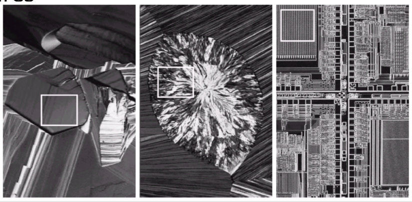
- Topology = The study of the properties of a figure that are unaffected by any deformation
- Topological descriptors
  - Number of holes in a region, H
  - Number of connected components, C
  - Euler number, E = C – H

A B

Centre for Image Analysis
Swedish University of Agricultural Sciences
Uppsala University
UPPSALA UNIVERSITET SLU

Texture

- 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

- 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, i.e. the probability of finding a gray value corresponding to the i:th gray level z_i.

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

2nd moment : Variance of z

3rd moment : Skewness

4th moment : Relative flatness

Histogram based descriptors

Uniformity and average entropy is computed similarly:

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

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

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

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

Co-occurrence matrix

- 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 Descriptors

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

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

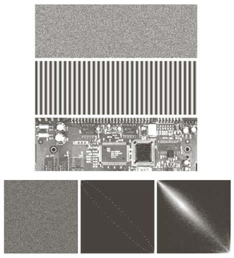
Co-occurrence matrix Descriptors

- 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

- Match image with a co-occurrence matrix!

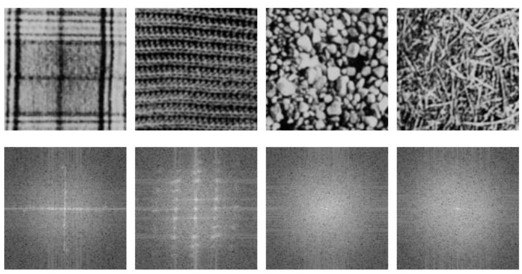


Spectral Analysis

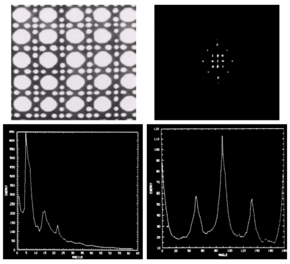
- Peaks in the Fourier spectrum give information about direction and spatial period patterns
- The spectrum can be described using polar coordinates S(r,θ)
- For each angle θ, S(r,θ) is a 1D function $S_\theta(r)$
- Similarly, for each frequency r, $S_r(\theta)$ is a 1D function
- A global description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



Spectral Analysis



S(r) S(θ)

Central Moments

- 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

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

Central Moments

- 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

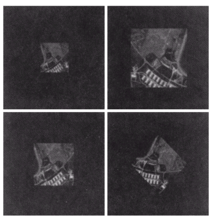
$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}$$

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

How to Design Descriptors?

- Find descriptors that are invariant to things that are unimportant for your task:
  - I.e. Noise, scale, blur, …
- Find descriptors that are ekvivariant with the your task
  - height, to classify adults / children
  - color and shape to separate bananas, pears and tomatoes

Conclusions

- We went from object boundaries and image regions to a set of invariant numbers that we call descriptors (or features)
- Next lecture we take a look at classifiers!
- Exercises 11.19 and 11.25 are recommended