Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

MIN-Fakultät
Fachbereich Informatik
Arbeitsbereich SAV/BV (KOGS)

Image Processing 1 (IP1)
Bildverarbeitung 1

Lecture 13 – Grouping and Shape Features

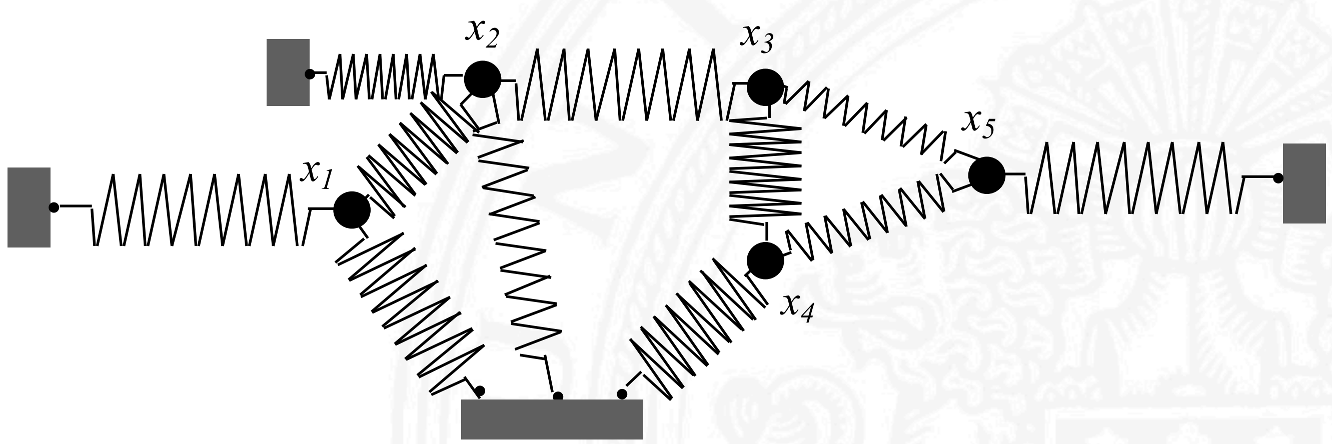
Winter Semester 2015/16

Slides: Prof. Bernd Neumann
Slightly revised by: Dr. Benjamin Seppke & Prof. Siegfried Stiehl

Grouping by Relaxation

Relaxation methods seek a solution by stepwise minimization ("relaxation") of constraints.

Analogy with spring system:

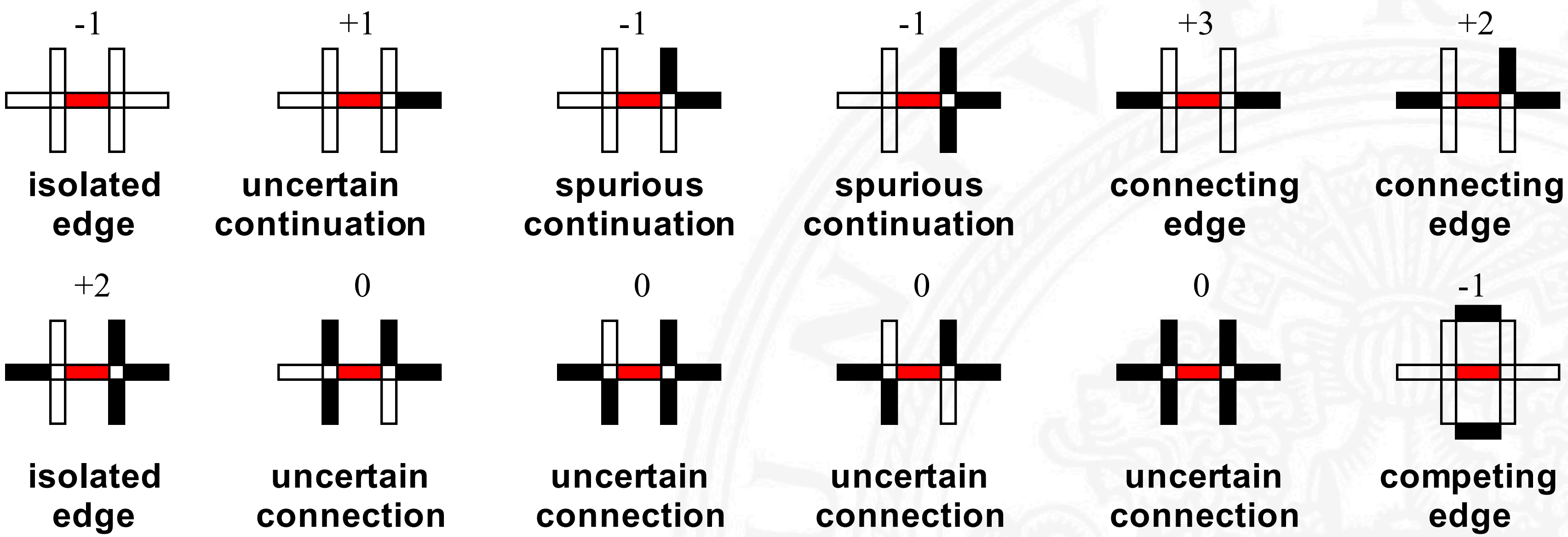


Variables x_i take on values (= positions) where springs are maximally relaxed corresponding to a state of global minimal energy. Hence relaxation is often realized by "energy minimization".

Contexts for Edge Relaxation

Iterative modification of edge strengths using context-dependent compatibility rules.

Context types:



Each context contributes with weight $w_j = w_0 \times \{-1 \ldots +2\}$ to an interative modification of the edge strength of the central element.

Modification Rule for Edge Relaxation

P_i^k edge strength in position i after iteration k

Q_{ij}^k strength of context j for position i after iteration k

w_j weight factor of context j

$Q_{ij}^k = \prod P_m^k \cdot \prod (1 - P_m)$ edge context strength

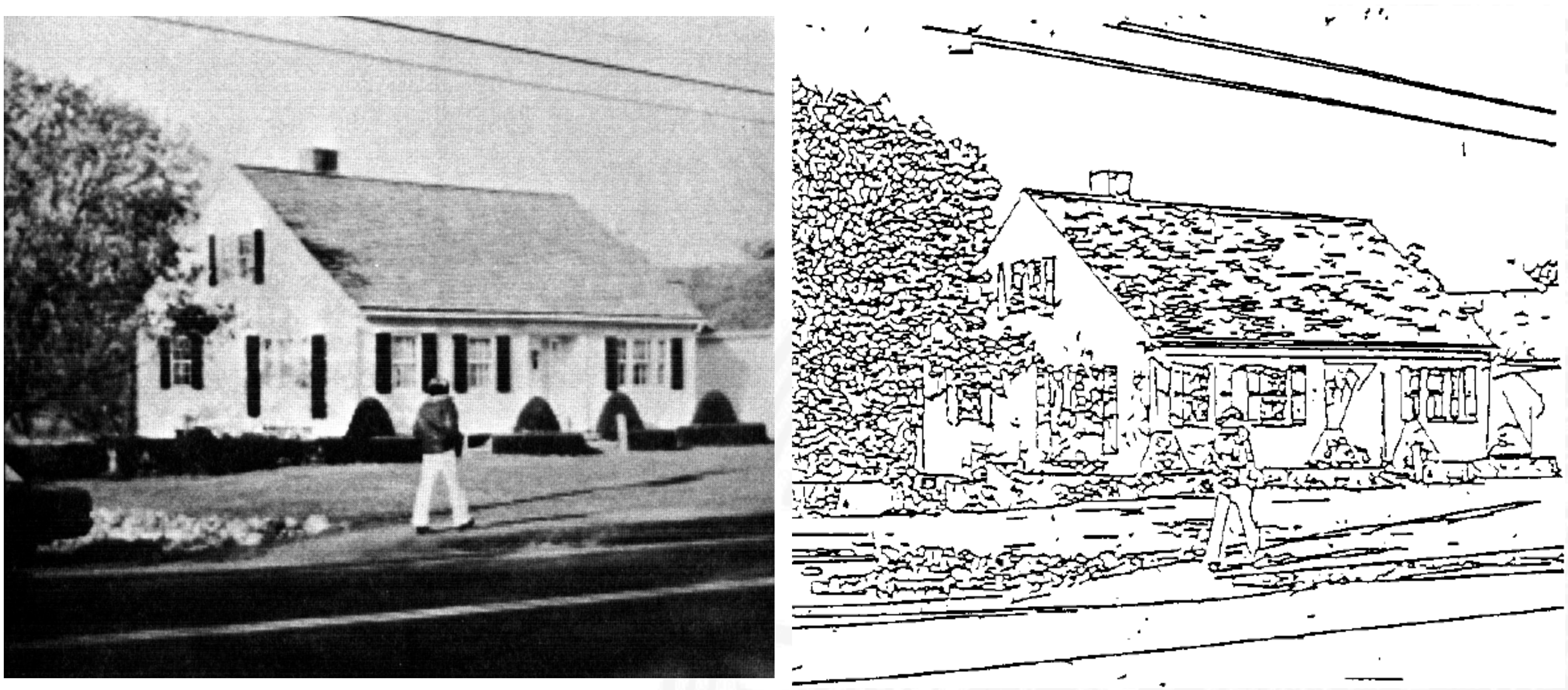
m, n ranging over all supporting and not supporting edge positions of context j, respectively.

$$P_i^{k+1} = P_i^k \frac{1 + \Delta P_i^k}{1 + P_i^k \Delta P_i^k}$$ edge strength modification rule

$$\Delta P_i^k = \sum_{j=1}^{N} w_j Q_{ij}^k$$ edge strength increment

There is empirical evidence (but no proof) that for most edge images this relaxation procedure converges within $10 \ldots 20$ iterations.

Example of Edge-finding by Relaxation



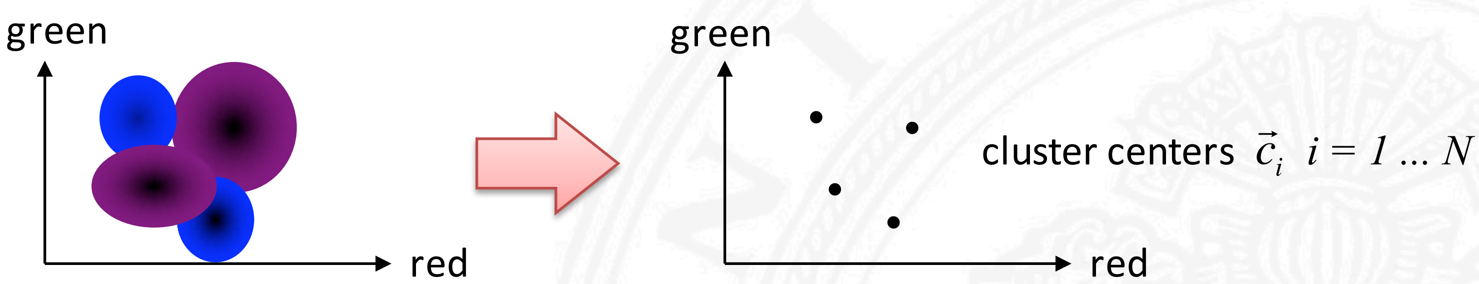
Landhouse scene from VISIONS project, 1982

Histogram-based Segmentation with Relaxation I

Basic idea:

Use relaxation to introduce a local similarity constraint into histogram-based region segmentation.

1. Determine cluster centers by multi-dimensional histogram analysis



2. Label each pixel by cluster-membership probabilities p_i, $1 = 1 \ldots N$

$$p_i = \frac{1/d_i}{\sum_{k=1}^{N} 1/d_i}$$

d_i is Euclidean distance between the feature vector of the pixel and cluster center $\vec{c}_i$

Histogram-based Labelling with Relaxation II

3. Iterative relaxation of the $p_i(j)$ of all pixels j:
   - equal labels of neighbouring pixels support each other
   - unequal labels of neighbouring pixels inhibit each other

$$q_i(j) = \sum_{k \in D(j)} [w^+ p_i(k) - w^-(1 - p_i(k))]$$

D(j) is neighbourhood of pixel j

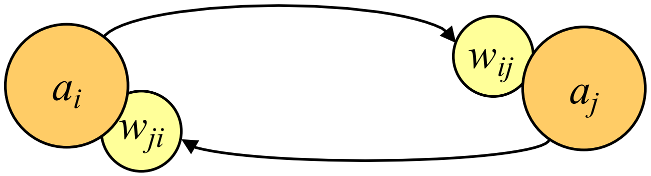
$$p_i'(j) = \frac{p_i(j) + q_i(j)}{\sum_n (p_n(j) + q_n(j))}$$

new probability $p_i´(j)$ of pixel j to belong to cluster i

4. Region assignment of each pixel according to its maximal membership probability: $\max p_i$
5. Recursive application of the procedure to individual regions

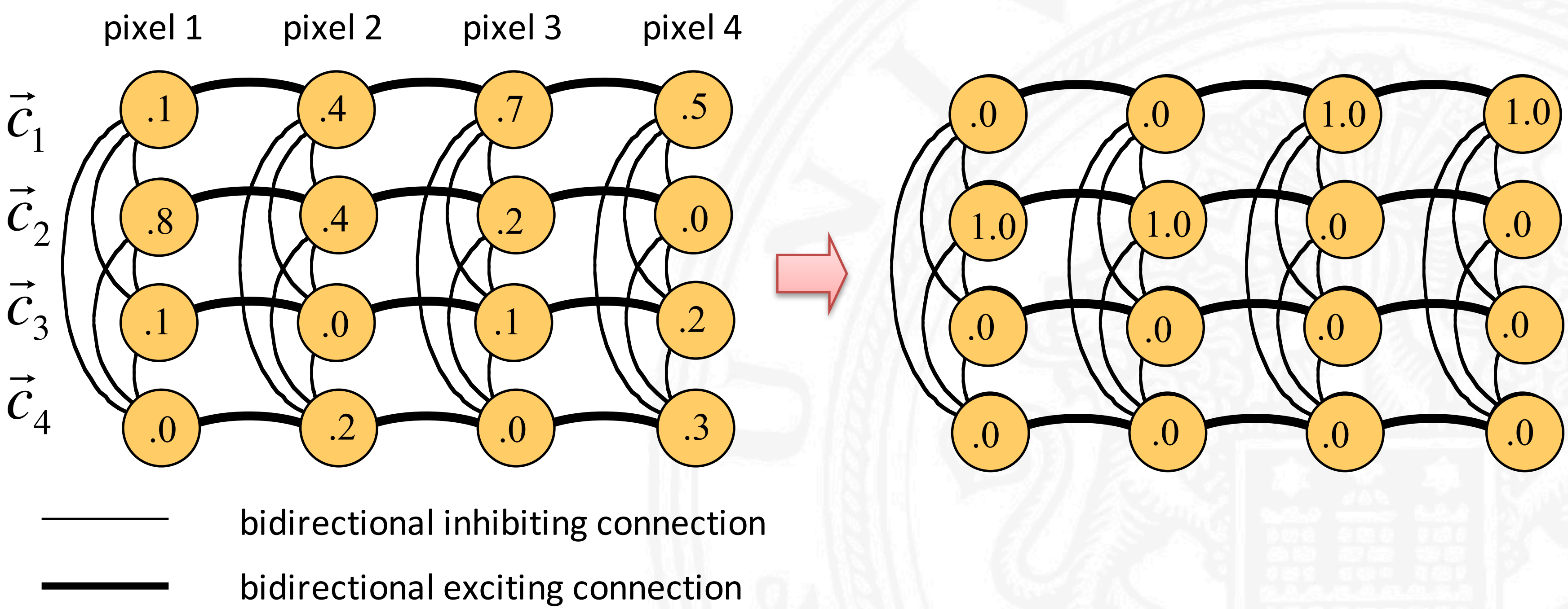
Relaxation with a Neural Network

Principle:



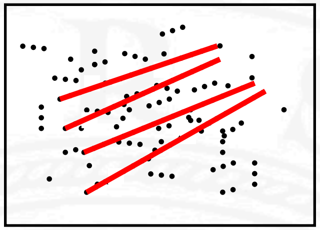
cells influence each other´s activation via exciting or inhibiting weights

Relaxation labelling of 4 pixels:



Hough Transform I

Robust method for fitting straight lines, circles or other geometric figures which can be described analytically.

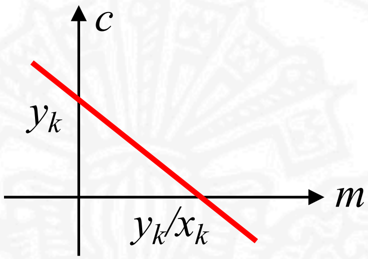


Given: Edge points in an image

Wanted: Straight lines supported by the edge points

An edge point (x_k, y_k) supports all straight lines $y = mx + c$ with parameters m and c such that $y_k = mx_k + c$.

The locus of the parameter combinations for straight lines through (x_k, y_k) is a straight line in parameter space.



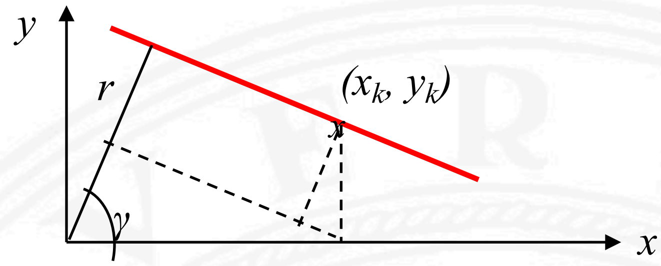
Principle of Hough transform for straight line fitting:

- Provide accumulator array for quantized straight line parameter combinations
- For each edge point, increase accumulator cells for all parameter combinations supported by the edge point
- Maxima in accumulator array correspond to straight lines in the image

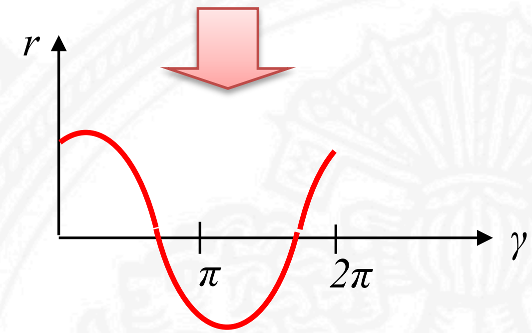
Hough Transform II

For straight line finding, the parameter pair (r, γ) is commonly used because it avoids infinite parameter values:

$$x_k \cos(\gamma) + y_k \sin(\gamma) = r$$



Each edge point (x_k, y_k) corresponds to a sinusoidal in parameter space:



Important improvement by exploiting direction information at edge points:

(x_k, y_k, φ) ⇒ $x_k \cos(\gamma) + y_k \sin(\gamma) = r$ restricted to $\varphi - \delta \leq \gamma \leq \varphi + \delta$

gradient direction (for φ)

direction tolerance (for δ)

Hough Transform III

Same method may be applied to other parameterizable shapes, e.g.

- circles: $(x_k - x_0)^2 + (y_k - y_0)^2 = r^2$ 3 parameters x_0, y_0, r

- ellipses

$$\left(\frac{(x_k - x_0)\cos\gamma + (y_k - y_0)\sin\gamma}{a}\right)^2 + \left(\frac{(y_k - y_0)\cos\gamma - (x_k - x_0)\sin\gamma}{b}\right)^2 = 1$$

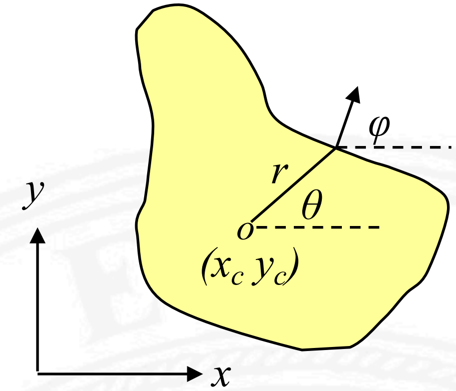
5 parameters x_0, y_0, a, b, γ

Accumulator arrays grow exponentially with number of parameters

→ quantization must be chosen with care

Generalized Hough Transform

- shapes are described by edge elements $(r\ \theta\ \ \varphi)$ relative to an arbitrary reference point $(x_c\ y_c)$
- φ is used as index into $(r\ \theta)$ pairs of a shape description
- edge point coordinates $(x_k\ y_k)$ and gradient direction φ_k determine possible reference point locations
- likely reference point locations are determined via maxima in accumulator array



φ_1: $\{(r_{11}\ \theta_{11})\ (r_{12}\ \theta_{12}) \dots \}$

φ_2: $\{(r_{21}\ \theta_{11})\ (r_{22}\ \theta_{12}) \dots \}$

$\vdots$

φ_N: $\{(r_{N1}\ \theta_{11})\ (r_{N2}\ \theta_{12}) \dots \}$

$$(x_k\ y_k\ \varphi_k) \Rightarrow \{(x_c\ y_c)\} = \{\ (x_k - r_i(\varphi_k)\ \cos\ \theta_i(\varphi_k),\ \ (y_k - r_i(\varphi)\ \sin\ \theta_i(\varphi_k))\ \}$$

counter cell in accumulator array

Region Description for Recognition

- For object recognition, descriptions of regions in an image have to be compared with descriptions of regions of meaningful objects (models).
- The general problem of object recognition will be treated later.
- Here we learn basic region description techniques for later stages in image analysis (including recognition).
- Typically, region descriptions suppress (abstract from) irrelevant details and expose relevant properties. What is "relevant" depends on the task.

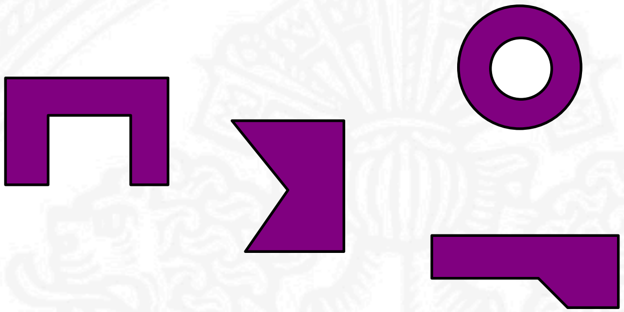
<u>Example</u>: OCR (Optical Character Recognition)

region **abstraction 1** **abstraction 2** **abstraction 3**

Simple 2D Shape Features

For industrial recognition tasks it is often required to distinguish
- **a small number of different shapes**
- **viewed from a small number of different view points**
- **with a small computational effort.**

In such cases simple 2D shape features may be useful, such as:
- area
- boxing rectangle
- boundary length
- compactness
- second-order momentums
- polar signature
- templates



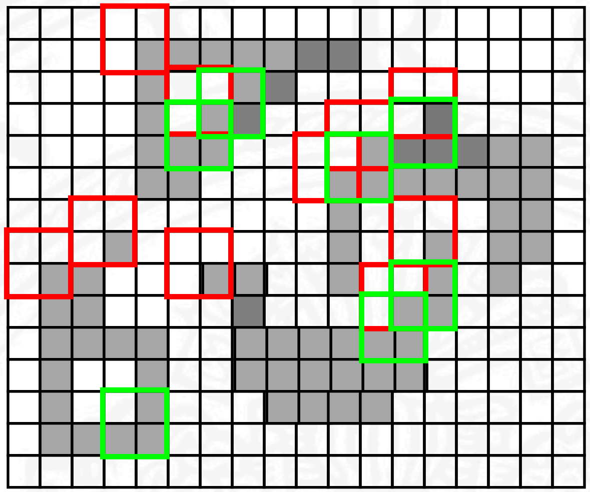
Features may or may not have invariance properties:
- 2D translation invariance
- 2D rotation invariance
- scale invariance

Euler Number

The Euler number is the difference between the number of disjoint regions and the number of holes in an image.

P = number of parts
H = number of holes
$E = P - H$

Example:
$P = 5$
$H = 2$
$E = 3$

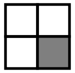


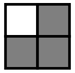
Surprisingly, E (but not P or H) can be computed by simple local operators.

Operators for regions with asymmetric connectivity:

4-connected NE and SW

8-connected NW and SE

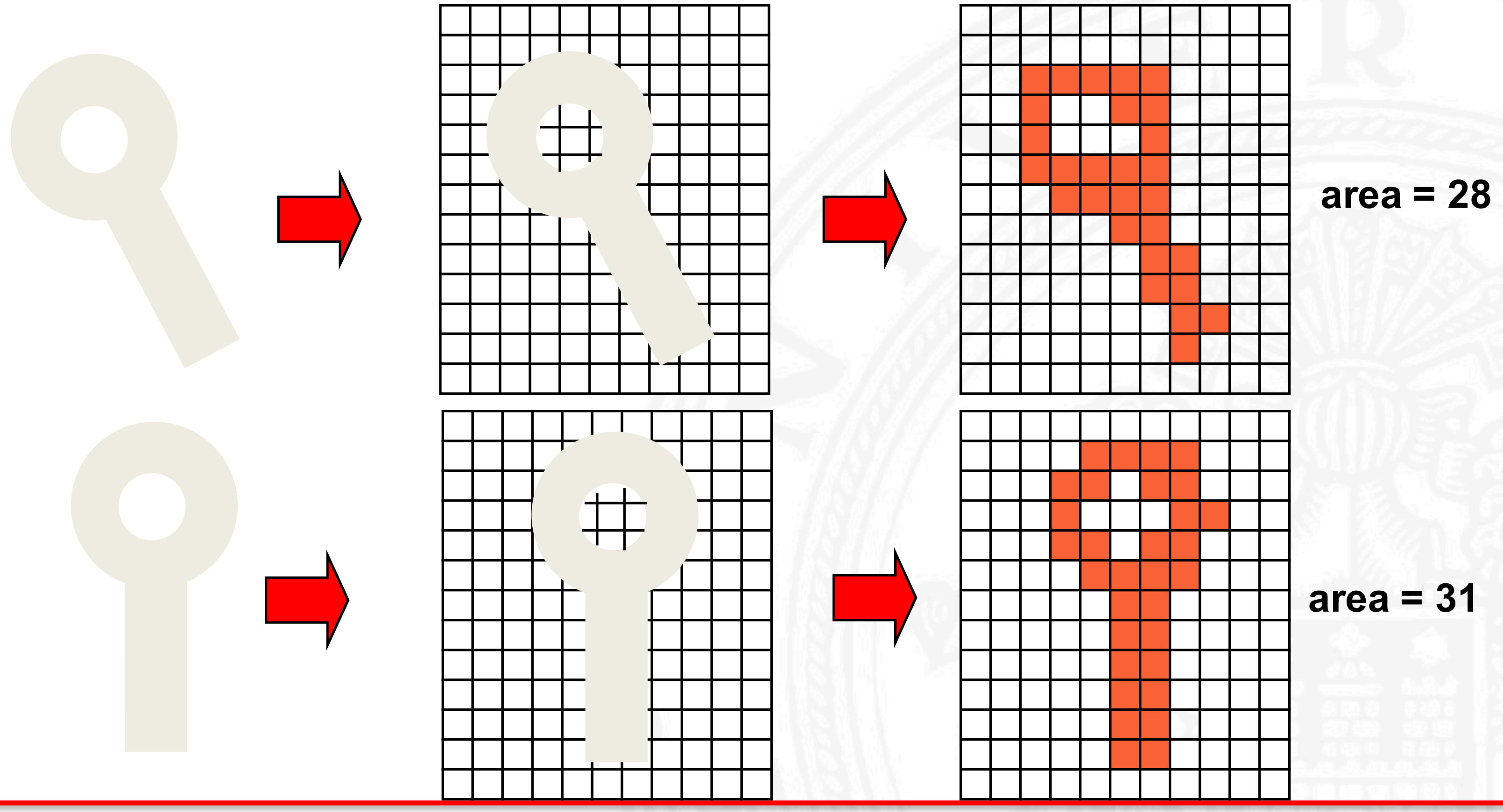
pattern1 = 

pattern2 = 

E = (count of pattern1) - (count of pattern2)

Area

The area of a digital region is defined as the number of pixels of the region. For an arbitrarily fine resolution, area is translation and rotation invariant. In praxis, discretization effects may cause considerably variations.

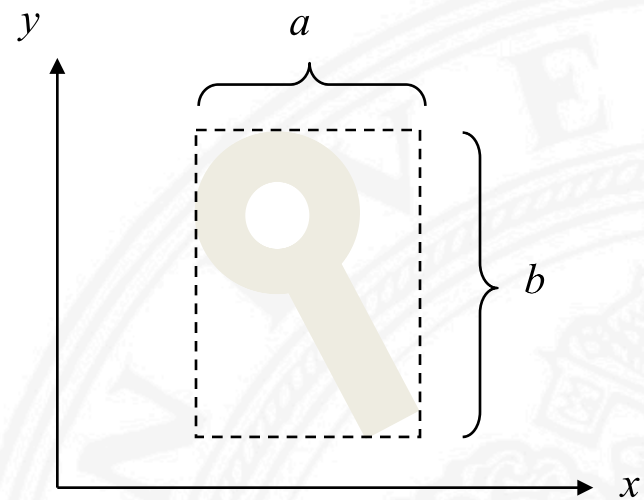


Boxing Rectangle

(a.k.a. Bounding Rectangle or Bounding Box)

Boxing rectangle = width of a shape in x- and y-direction

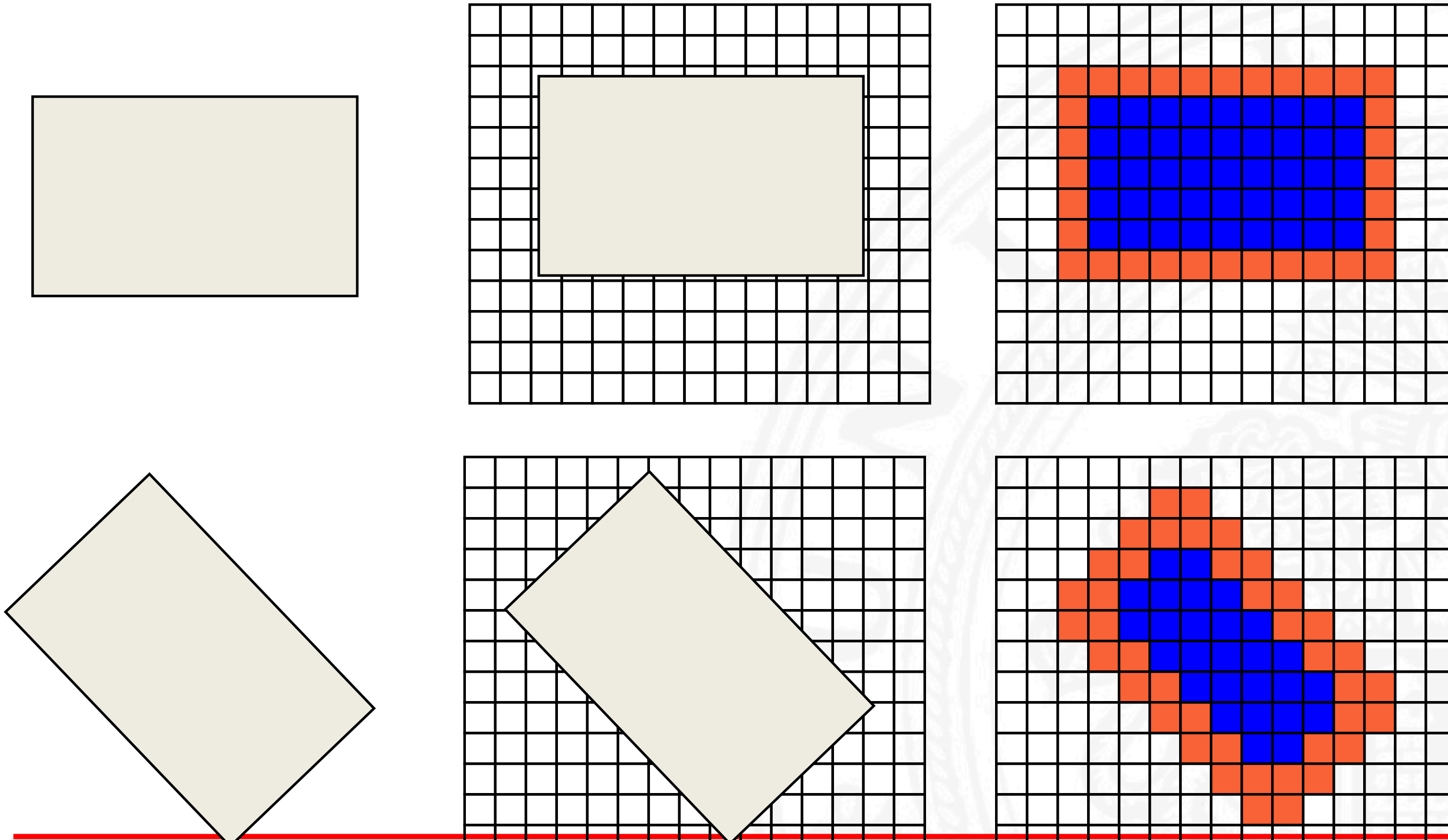
- easy to compute
- <u>not</u> rotation invariant



To achieve rotation invariance, the rectangle must be fitted parallel to an innate orientation of the shape. Orientation can be determined as the axis of least inertia (see second order moments).

Boundary Length

The boundary length is defined as the number of pixels which constitute the boundary of a shape.



area = 77

boundary length = 32

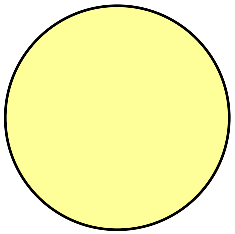
area = 69

boundary length = 40

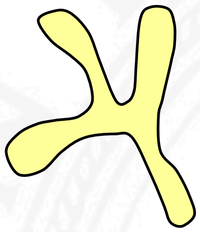
Compactness

$$\text{(non-)compactness} = \frac{(\text{boundary length})^2}{\text{area}}$$

Compactness describes <u>analog</u> shapes independent of linear transformations.



very compact



not very compact

Compactness for <u>discrete</u> shapes is in general <u>not</u> translation, rotation or scale invariant due to discretization effects.

Center of Gravity

Consider a 2D shape evenly covered with mass. Physical concepts such as:

- center of gravity
- moments of inertia

may be applied.

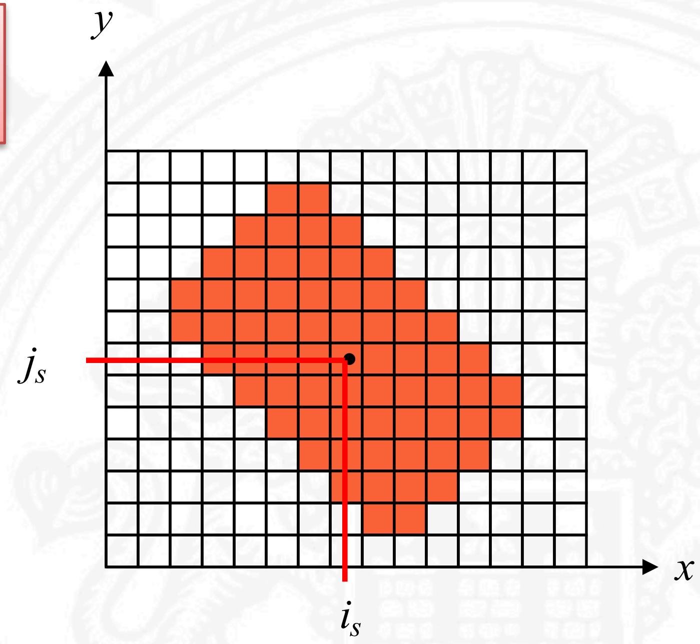
The center of gravity is the location where first-order moments sum to zero.

Center-of-gravity coordinates:

D = digital region

$$\sum_{ij\in D} (i - i_s) = 0 \qquad \sum_{ij\in D} (j - j_s) = 0$$

$$\Rightarrow \quad i_s = \frac{1}{|D|}\sum_{ij\in D} i \qquad j_s = \frac{1}{|D|}\sum_{ij\in D} j$$



Second-order Moments

Second-order moments ("moments of inertia") measure the distribution of mass relative to axes through the center of gravity.

$$m_x = \sum_{ij \in D} (\text{i-i}_s)^2 = \sum_{ij \in D} i^2 - i_s^2 |D|$$

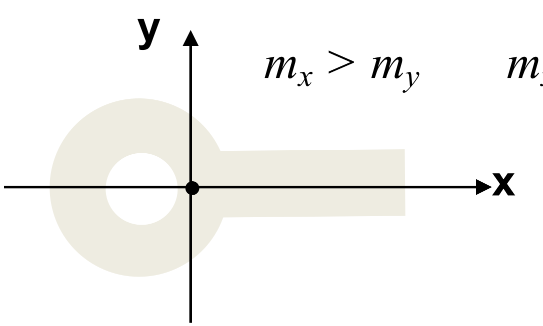
moment of inertia relative to y-axis through center of gravity

$$m_y = \sum_{ij \in D} (\text{j-j}_s)^2 = \sum_{ij \in D} j^2 - j_s^2 |D|$$

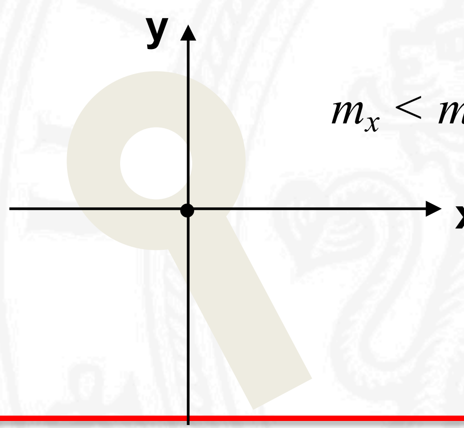
moment of inertia relative to x-axis through center of gravity

$$m_{xy} = \sum_{ij \in D} (\text{i-i}_s)(\text{j-j}_s) = \sum_{ij \in D} ij - i_s j_s |D|$$

"mixed"moment of inertia relative to x- and y-axis through center of gravity,
zero if x- and y-axis are "main axes"



$m_x > m_y \qquad m_{xy} = 0$



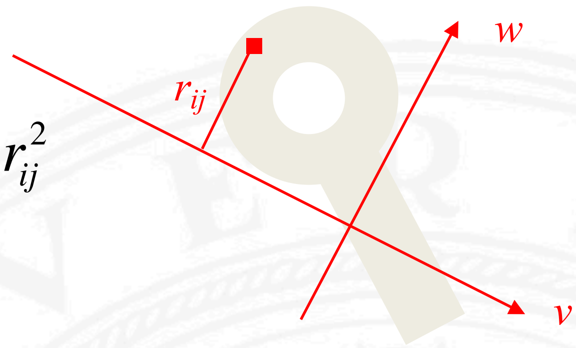
$m_x < m_y \qquad m_{xy} \neq 0$

Axis of Minimal Inertia

The axis of minimal inertia can be used as an innate orientation of a 2D shape.

Inertia (= second order moment) relative to an axis is the sum of the squared distances between all pixels of the shape and the axis.

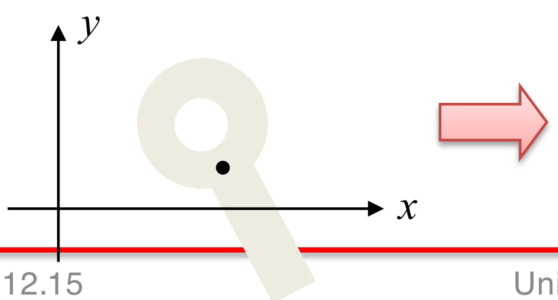
$$m_v = \sum_{ij \in D} r_{ij}^2$$

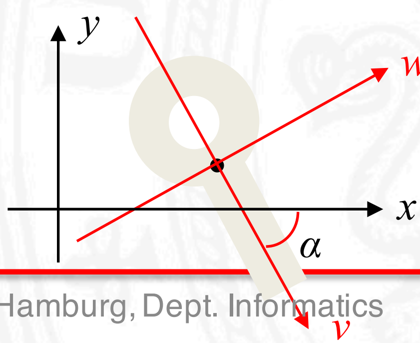


1. The axis of least inertia passes through the center of gravity
2. The mixed moment m_{vw} relative to the axes v and w must be zero

If the mixed moment is nonzero, the axis must be turned by the angle α:

$$\tan 2\alpha = \frac{2m_{xy}}{m_y - m_x}$$

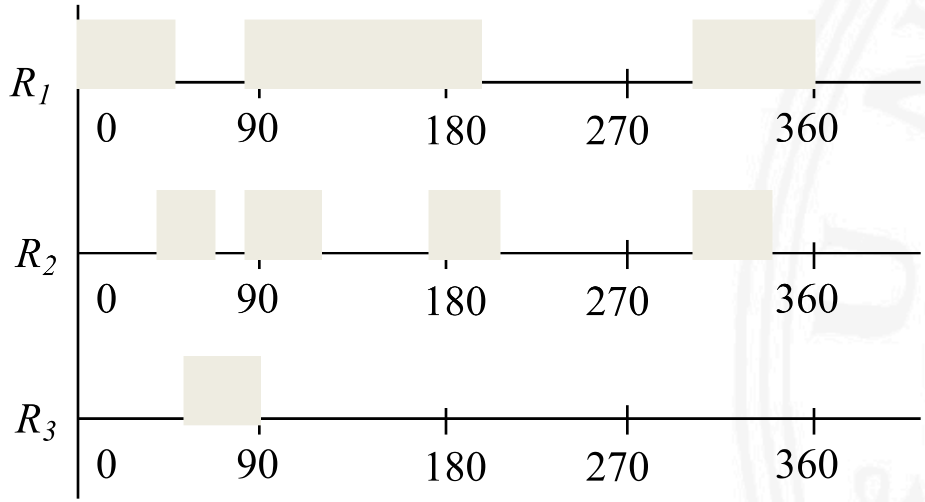


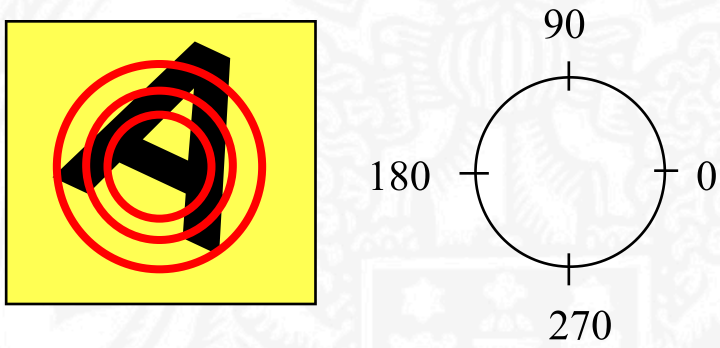


Polar Signature

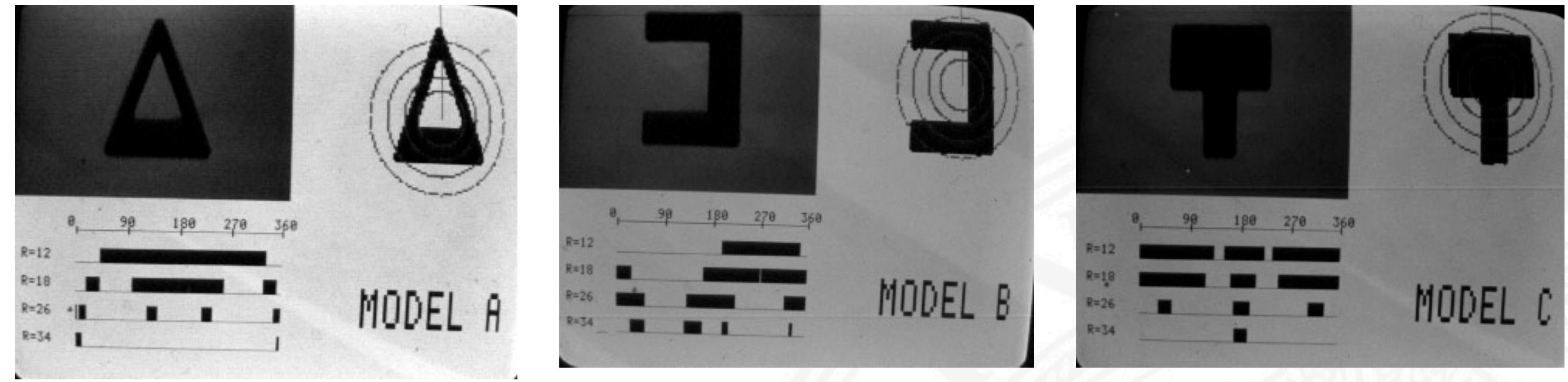
The polar signature records the angular segments where circles around the center of gravity lie within a shape.

- scalable from coarse to fine by appropriate number of circles
- radii of circles must be chosen judiciously
- translation-invariant
- rotation-invariance can be achieved by cyclic shifting

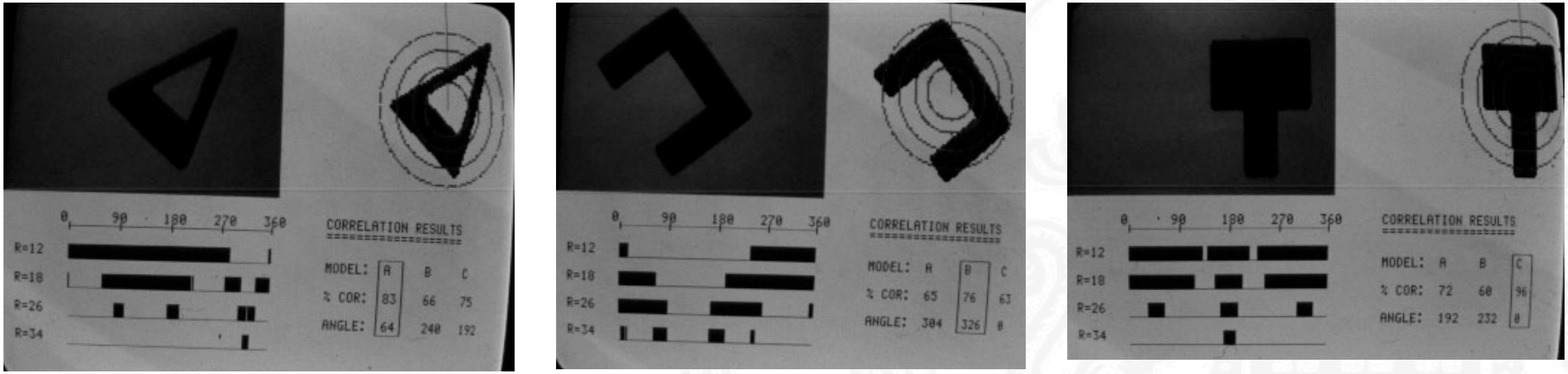




Object Recognition Using the Polar Signature



Model signatures



Recognition results

Convex Hull

A region R is convex if the straight-line segment x_1x_2 between any two points of R lies completely inside of R.

For an arbitrary region R, the convex hull H is the smallest convex region which contains R.

Example of shape with convex hull:

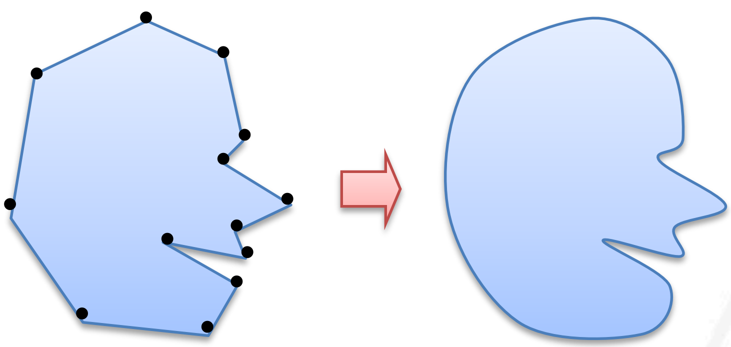
Intuitive convex hull algorithm:

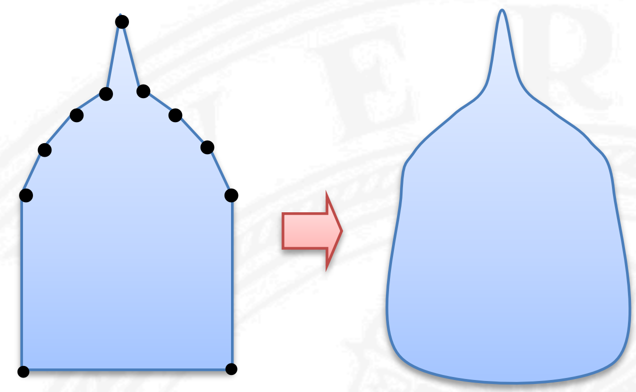
1. Pick lowest and left-most boundary point of R as starting point $P_k = P_1$. Set direction of previous line segment of convex-hull boundary to $\vec{v} = (0\ \ {-1})^T$
2. Follow boundary of R from current point P_k in an anti-clockwise direction and compute angle θ_n of line P_kP_n for all boundary points P_n after P_k. The point P_q with $\theta_q = \min\{\theta_n\}$ is a vertex of the convex hull boundary.
3. Set $P_k = P_q$ and $\vec{v} = (P_k\ P_n)^T$ and repeat 2) and 3) until $P_k = P_1$.

There are numerous convex hull algorithms in the literature. The most efficient is O(N) [Melkman 87], see Sonka et al. "Image Processing ...".

B-Splines I

B-splines are piecewise polynomial curves which provide an approximation of a polygon based on vertices.





precision depends on distances of vertices

Important properties:

- eye-pleasing smooth approximation of control polygon
- change of control polygon vertex influences only small neighbourhood
- curve is twice differentiable (e.g. has well-defined curvature)
- easy to compute

B-Splines II

B-splines can be defined by means of a parametric (closed) curve with one free parameter s:

$$\vec{x}(s) = \sum_{i=0}^{N+1} \vec{v}_i B_i(s)$$

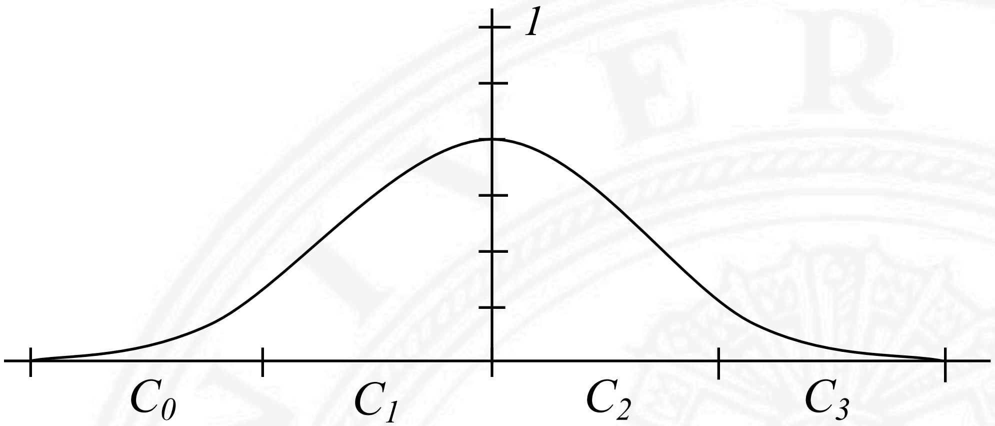
with:

- s parameter, changing linearly from i to $i+1$ between vertices $\vec{v}_i$ and $\vec{v}_{i+1}$
- $\vec{v}_i$ vertices of control polygon
- $B_i(s)$ base functions, nonzero only in $[i-2\,,\, i+2]$

B-Splines III

Each base function $B_i(s)$ consists of four parts:

$$\left.\begin{aligned}
C_0(t) &= \frac{t^3}{6}\\
C_1(t) &= \frac{-3t^3+3t^2+3t+1}{6}\\
C_2(t) &= \frac{3t^3-6t^2+4}{6}\\
C_3(t) &= \frac{-3t^3+3t^2-3t+1}{6}
\end{aligned}\right\}$$



The resulting curve equation is:

$$\vec{x}(s) = C_3(s-i)\vec{v}_{i-1} + C_2(s-i)\vec{v}_i + C_1(s-i)\vec{v}_{i+1} + C_0(s-i)\vec{v}_{i+2}$$

Example: s= 7.7, i=7

$$\vec{x}(7.7) = C_3(0.7)\vec{v}_6 + C_2(0.7)\vec{v}_7 + C_1(0.7)\vec{v}_8 + C_0(0.7)\vec{v}_9$$

Shape Description by Fourier Expansion I

The curvature function k(s) of a region is necessarily periodic:

$k(s) = k(s+L)$ $L =$ length of boundary

Hence k(s) can be expanded by a Fourier series with coefficients:

$$c_n = \frac{1}{L}\int_0^L k(s)\exp\left(-\frac{2\pi i n}{L}s\right)ds$$

To avoid problems with curvature discontinuities at corners, it is useful to consider the slope intrinsic function:

$$\boxed{\theta'(s) = \theta(s) - \frac{2\pi s}{L} - \mu}$$

with:

$$\theta(s) = \int_0^s k(\varsigma)\,d\varsigma$$

tangent angle

normalization (to achieve periodicity)

$$\mu = \frac{1}{L}\int_0^L \left[\theta(s) - \frac{2\pi s}{L}\right]ds$$

mean (dependent on starting point)

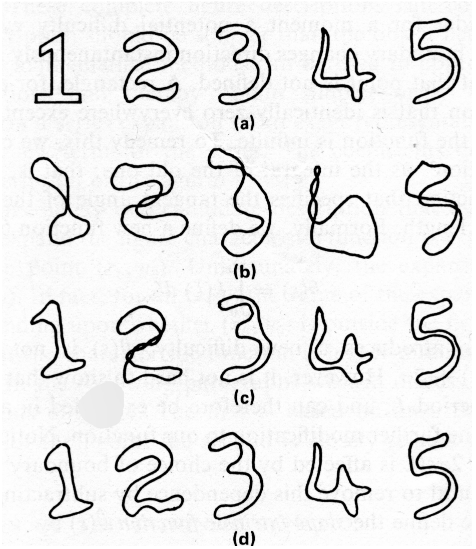
Shape Description by Fourier Expansion II

The shape of a contour can be approximately represented by a limited number of harmonics of the Fourier expansion of the slope intrinsic function $\theta'(s)$:

$$c_n = \frac{1}{L}\int_0^L \theta'(s)\exp(-\frac{2\pi i n}{L}s)ds$$

Example:

(from Duda and Hart 73: Pattern Classification and Scene Analysis)



original

5 harmonics

10 harmonics

15 harmonics

Caution:
It is questionable whether the approximations by a limited number of harmonics capture the most frequent deviations from the normal.