Object / 2D image descriptors

Václav Hlaváč

Czech Technical University in Prague
Czech Institute of Informatics, Robotics and Cybernetics
Praha 6, Jugoslávských partyzánů 1580/3, Czech Republic
http://people.ciirc.cvut.cz/hlavac, vaclav.hlavac@cvut.cz
also Center for Machine Perception, http://cmp.felk.cvut.cz

Lecture plan

1. Problems in describing objects in images mathematically.
2. A taxonomy of image/objects descriptions.
3. Simple 2D object descriptors (region-based, boundary-based).
4. Matching region of interest.

Computer vision / segmentation is hard

- Many pictures are difficult to interpret.
- A large part of the brain is devoted to vision.
- $\approx$ 50 years of research in computer vision and we are still nowhere near a solution.

Problems in describing objects in images mathematically:

- Ill-definedness.
- Ill-posedness.
- Intractability.

Ill-definedness

- Scene models used in general recovery tasks are not fully defined.
  - ‘Piesewise’ simple means nothing unless we impose a lower bound on the piece sizes.
  - ‘Noise’ is not easy to model (or to distinguish from the ‘signal’); Noise is often not Gaussian!
- Many object classes easily recognizable by humans do not have simple definitions (As chairs, bushes, dogs, . . .)

Functional description, categorization

H. Bülthof's counterexample

Functional description, issues

Domain

- Suitable for man-made objects.

Problems

- The object function is often difficult to extract.
- Mapping shape to function.
- Even shape is difficult to extract.

Ill-posedness

- Recovery problems are usually underconstrained – e.g., ambiguity of illumination / photometry / geometry.
- (Questionable) approach – add constraints, e.g.,
  - smoothness (regularization),
  - minimal description length principle.

  - Critique: the actual scene may not satisfy the constraints!

Intractability

- Recovery and recognition tasks are of combinatorial complexity.
- Parallelism can speed up the early stages of the vision process (e.g., image operations).
  However, a little is known about how to speed up the potentially combinatorial stages.
- Applications force us to solve vision problems in real time with inadequate algorithmic/computational resources $\Rightarrow$ suboptimality.

What can be done?

- Define your domain! Work in a domain that can be adequate (e.g., specialized). Explore semantics of the domain.
- Improve inputs,
  - Sensory redundancy (multisensor fusion, active vision).
  - Processing redundancy (consensus).
- Take your time. Use adequate computational resources.

A taxonomy of image/objects descriptions

- Object detection and recognition, i.e., the pattern recognition approach. Shape is difficult to express mathematically.
- Alignment ($\approx$ correspondence problem).
- Invariants.
- Decomposition into parts (expressing structure, relational graph).
- Functional description.

Four possibilities

Description (local/global) vs. entire image/object

	Image	Object
Global descr.	◆ Fourier transform (or other linear integral transform). ◆ Principal component analysis. ◆ No matching of individual shapes.	◆ Matching of a single, whole, object. ◆ Features: simple descriptors, moments, . . . ◆ Drawbacks: perfect segmentation nee-ded, sensitive to noise and occlusion.
Local descr.	Deliberately empty.	◆ No object segmentation needed. ◆ Matching is based on local descriptors. ◆ Features: interest points, corners, local structures, curvature.

Region identification, formulation

Called also connected component labeling

Input – binary image where all object' pixels $= 1$ and background pixels $= 0$.

Output – each region (connected component) has an unique label.

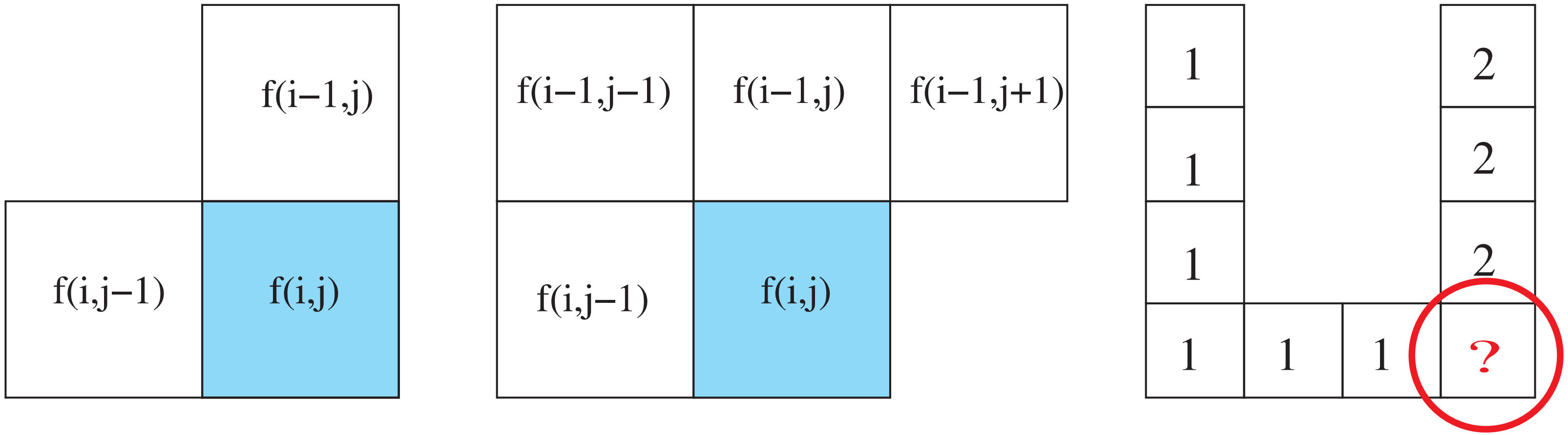
Two algorithms

1. Two passes algorithm.
2. Recursive filling of regions.

Connected components labeling
Two passes algorithm

Pass 1: identifies # of connected components based shifting a mask checking local neighborhood.

Pass 2: resolves label conflicts which are not seen locally.



4-neigborhood mask

8-neghborhood mask

conflict of labels

Simple 2D object descriptors

2 basic approaches to describe a region

- Region-based.
- Boundary-based.
  - Straight lines (chain codes, polylines).
  - Curved lines (circles, elipses, 2D polynomials, B-splines).
  - Ψ-S description, bending energy, chord distribution.
  - Fourier transform of boundaries.

Simple region descriptors

Area $=\#$ of pixels.

If the resolution increases while capturing the image then the area converges to the correct value.

If real area (e.g., in m^2) is sought then multiplication with the appropriate normalization constant is needed.

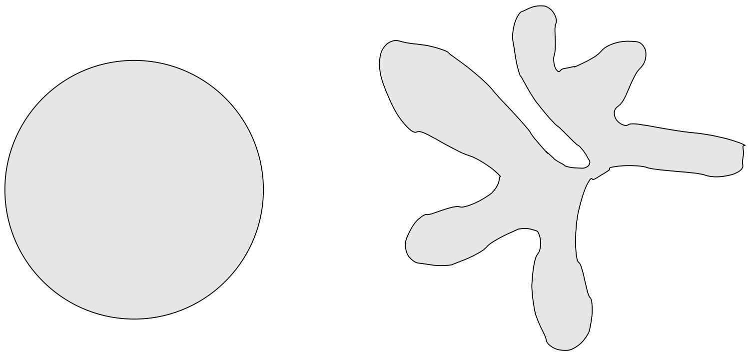
Perimeter $=\#$ of boundary pixels.

Compensation for 8-neighborhood shortening possible, $\sqrt{2}$ instead of 2. To be explained with chain code.

If the resolution increases while capturing the image then the perimeter converges to ∞. *How long is the coastline of Britain?*

Compactness

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



- *Called also circularity in the literature.*
- Troubles with digitization artifacts because the perimeter calculation is sensitive to noise.

R. Haralick's circularity

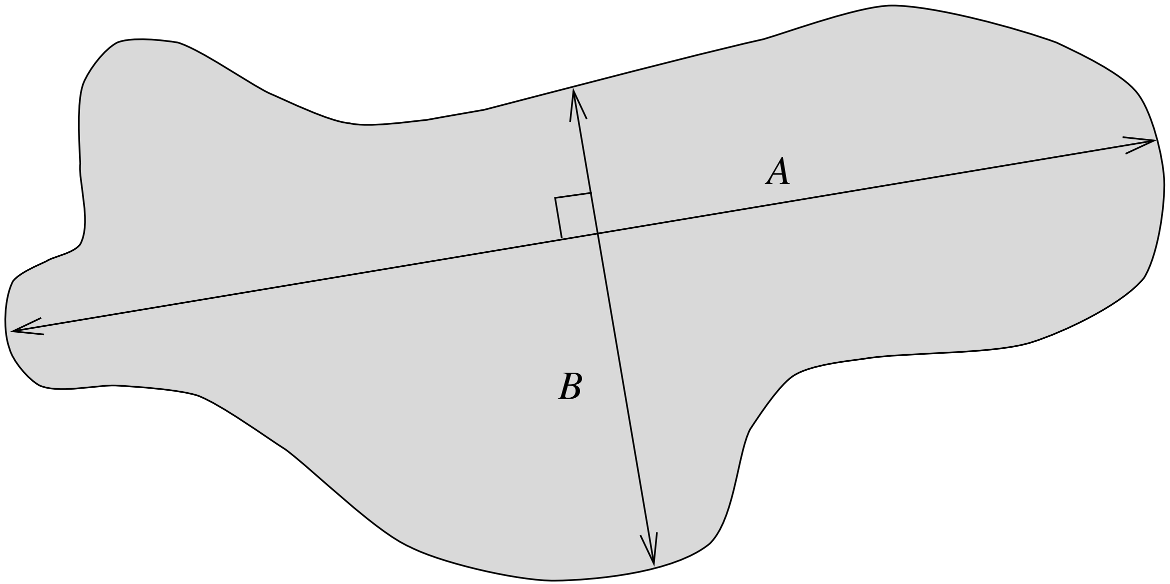
- Overcomes digitization artifacts mentioned with the compactness.
- Let R be the distance between the centroid of the region and any point on the regions boundary. Let μ_R be the mean of R and σ_R be the standard deviation of R.

$$\textit{Haralic's circularity} \quad \mathrm{Circ}_H = \frac{\mu_R}{\sigma_R}$$

- As the region becomes more compact (circular), the Circ_H increases.
- It is orientation and size independent.
- Circ_H values for discretized regions follow well values for the corresponding continuous regions.
- *R. M. Haralick: A measure for circularity of digital figures. IEEE Transactions on Systems, Man and Cybernetics, Issue 4, July 1974, pp. 394 - 396.*

Eccentricity

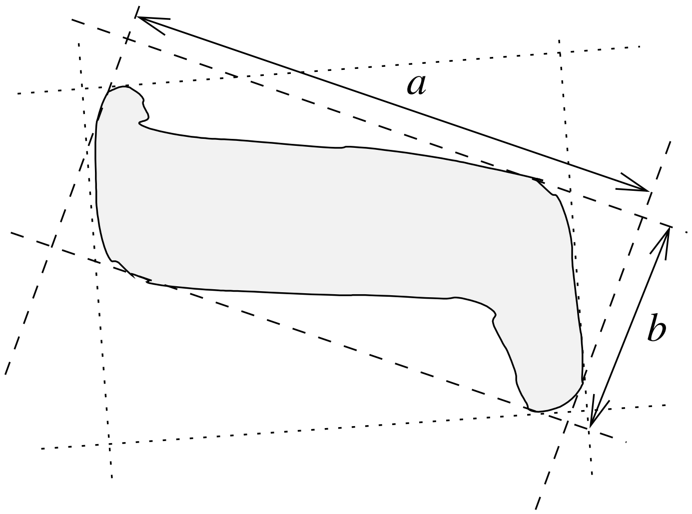
The simplest eccentricity characteristic is the ratio of the length of the maximum chord A to the maximum chord B which is perpendicular to A (the ratio of major and minor axes of an object).



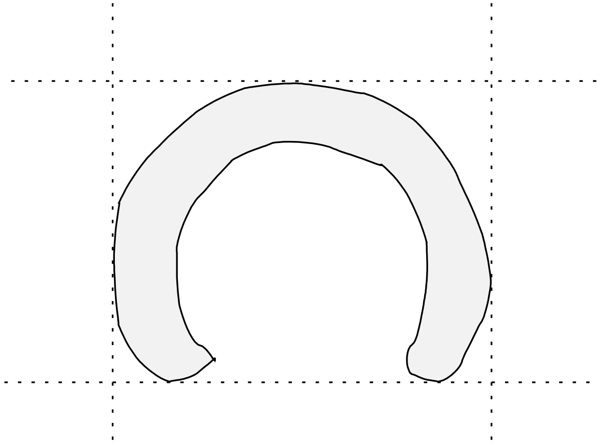
Elongatedness

The elongatedness is the ratio between the length and width of the region bounding rectangle of minimal area.

The minimal bounding rectangle is located by turning bounding rectangle in discrete steps until a minimum is located.



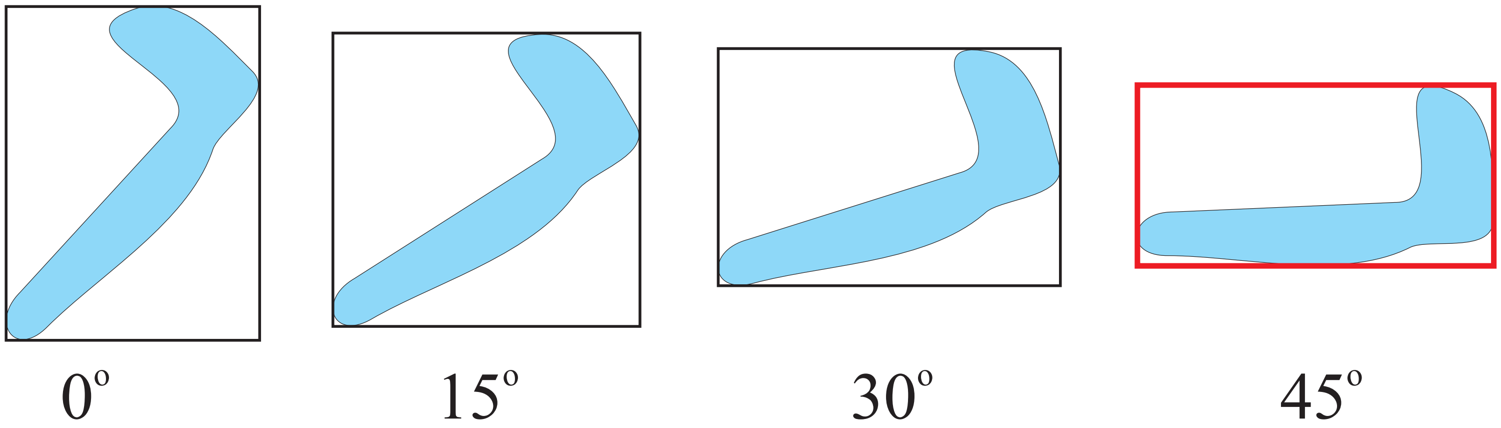
suitable



not suitable

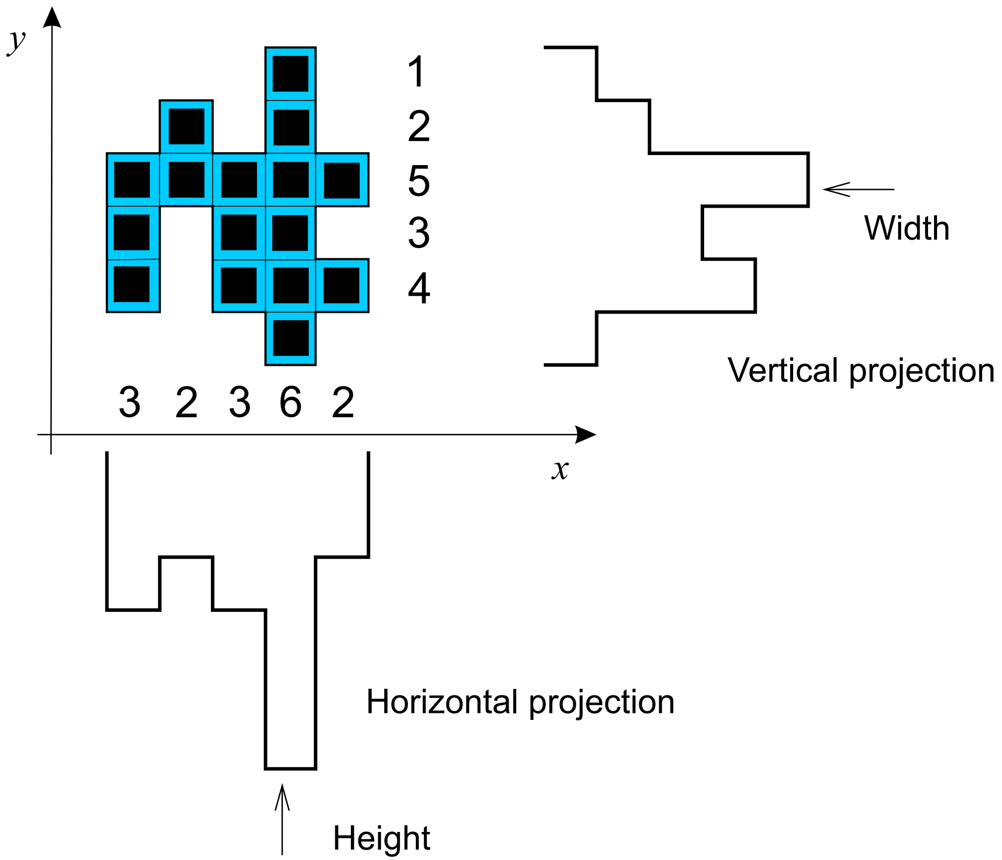
Rectangularity

- Consider the region and its bounding rectangle in orientation k in several discrete steps, e.g., $k \in \{0^o, 15^o, 30^o, 45^o, \ldots, 90^o\}$.
- $F_k = \dfrac{\textit{area of the region}}{\textit{area of the bounding rectangle in orientation } k}$
- $\textit{rectangularity} = \max\limits_k F_k$



Projections of a region

- Horizontal projection
  $$p_h(x) = \sum_y f(x, y).$$
- Vertical projection
  $$p_v(y) = \sum_x f(x, y).$$



Moments

- Let $f(x, y)$ be a gray scale (continuous) image containing a single object.
- The object is uniquely described by an infinite sequence of moments m_{pq},

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q \, f(x, y) \, \mathrm{d}x \, \mathrm{d}y \; .$$

- A discrete version is needed in a digital image $f(i, j)$,

$$m_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} i^p j^q \, f(i, j) \; .$$

Moment invariants

While describing an object globally in the image, the invariance to various transformations is often needed, e.g., to

- Translation (central moments).
- Scale.
- Rotation.
- Reflection.

Central moments

- The aim is to achieve invariance to translation.
- Center of gravity x_c, y_c of an object: $x_c = m_{10}/m_{00}$, $y_c = m_{01}/m_{00}$.
- Central moments

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - x_c)^p \, (y - y_c)^q \, f(x, y) \, \mathrm{d}x \, \mathrm{d}y$$

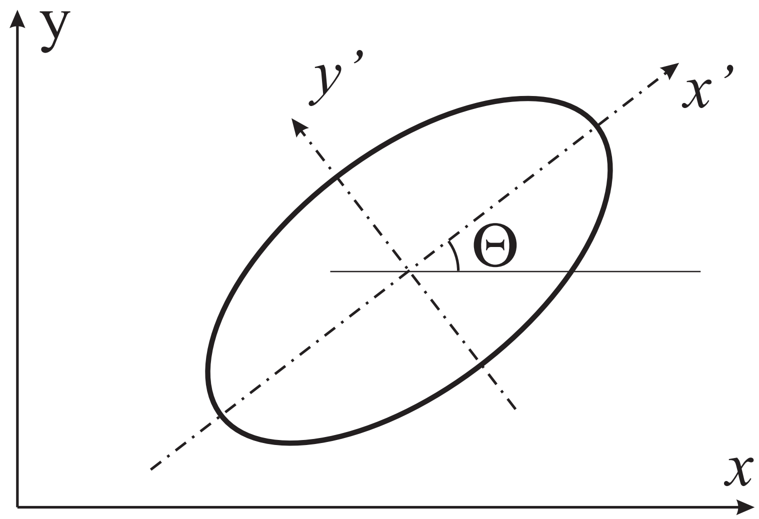
$$\mu_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} (i - x_c)^p \, (j - y_c)^q \, f(i, j)$$

Moment invariants under translation; Central moments

$$
\begin{aligned}
\text{Area } \mu_{00} &= m_{00} = \mu \\
\text{Centroid coordinates } \mu_{10} &= \mu_{10} = 0 \\
\text{Horizontal centralness } \mu_{20} &= m_{20} - \mu\, x_c^2 \\
\text{Diagonality } \mu_{11} &= m_{11} - \mu\, x_c\, y_c \\
\text{Vertical centralness } \mu_{02} &= m_{02} - \mu\, x_c\, y_c \\
\text{Horizontal imbalance } \mu_{30} &= m_{30} - 3\, m_{20}\, x_c + 2\, \mu\, x_c^3 \\
\text{Vertical divergence } \mu_{21} &= m_{21} - m_{20}\, y_c - 2\, m_{11}\, x_c + 2\, \mu\, x_c^2\, y_c \\
\text{Horizontal divergence } \mu_{12} &= m_{12} - m_{02}\, x_c - 2\, m_{11}\, y_c + 2\, \mu\, x_c\, y_c^2 \\
\text{Vertical imbalance } \mu_{03} &= m_{03} - 3\, m_{02}\, y_c + 2\, \mu\, y_c^3 \\
\text{Object spread or size } S &= \mu_{02} + \mu_{20}
\end{aligned}
$$

Moment use example: Object orientation

Orientation angle $\Theta = \frac{1}{2} \arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right)$



Moment invariants under scaling

A uniform scaling by a factor α is assumed.

$$\eta_{p,q} = \frac{\frac{\mu_{p,q}}{\alpha^{(p+q+2)}}}{\mu_{0,0}^{\frac{(p+q+2)}{2}}}$$

Moment invariants under translation, rotation, scale

$$
\begin{aligned}
\varphi_1 &= \mu_{20} + \mu_{02} \\
\varphi_2 &= (\mu_{20} - \mu_{02})^2 + 4\,\mu_{11}^2 \\
\varphi_3 &= (\mu_{30} - 3\,\mu_{12})^2 + (3\,\mu_{21} - \mu_{03})^2 \\
\varphi_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} - \mu_{03})^2 \\
\varphi_5 &= (\mu_{30} - 3\,\mu_{12}) + (\mu_{30} + \mu_{12})\,\left((\mu_{30} + \mu_{12})^2 - 3\,(\mu_{21} + \mu_{03})^2\right) \\
&\quad + (3\,\mu_{21} - \mu_{03}) + (\mu_{21} - \mu_{03})\,\left(3\,(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right) \\
\varphi_6 &= (\mu_{20} - \mu_{02})\left((\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right) + \; 4\,\mu_{11}\,(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03})
\end{aligned}
$$

In addition, invariant under reflection symmetry

$$
\begin{aligned}
\varphi_7 &= (3\,\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})\left((\mu_{30} + \mu_{12})^2 - 3\,(\mu_{21} + \mu_{03})^2\right) \\
&\quad + (\mu_{30} - 3\,\mu_{12})(\mu_{30} + \mu_{12})\left(3\,(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2\right)
\end{aligned}
$$

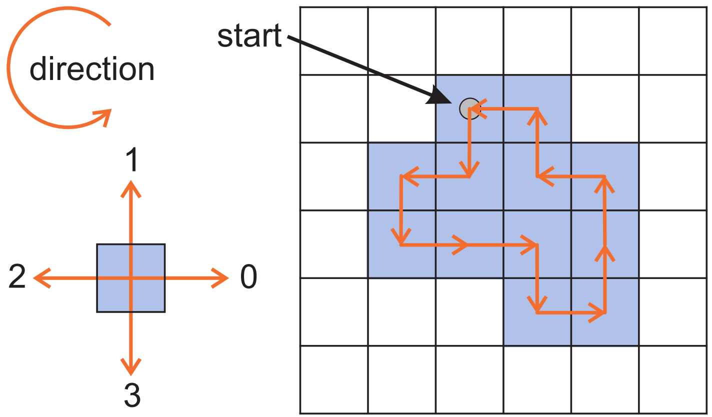
Border-based descriptors

- Chain code.
- Fourier descriptors.
- Chord distribution.
- B-splines.

Chain code of a region boundary

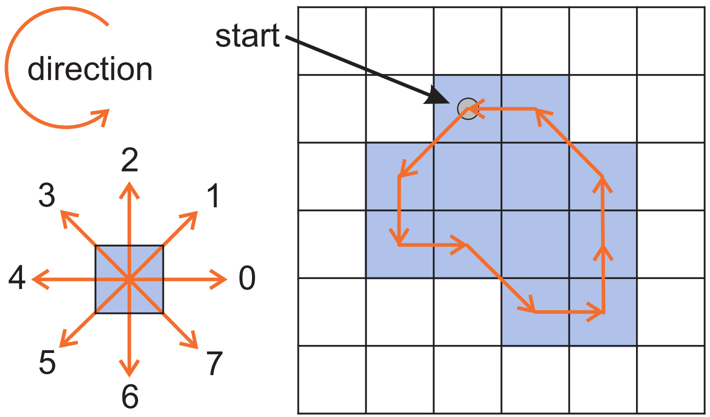
- The chain code (H. Freeman 1961) is a special case of the region boundary polygonal representation. The individual polygonal segments are of length 1 of the used neighborhood relation (4, 8, 6 -neighbors).
- A starting point is given, e.g. the most top-left pixel.
- The anti-clockwise direction is assumed while traversing the region boundary.
- Fast implementation: a 3×3 neighborhood and look into 256-lookup table.
- Disadvantage: the chain code depends on a starting point.

4-neighborhood



Chain code: 3 2 3 0 0 3 0 1 1 2 1 2

8-neighbourhood

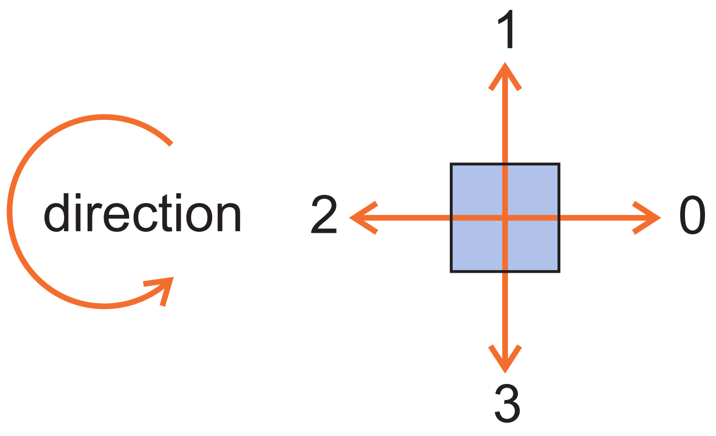


Chain code: 5 6 0 7 0 2 2 3 4

Derivative dd of the chain code

- Derivative dd (also the first difference) of the chain code yields the rotation invariance up to 90^o for 4-neighborhood or up to 45^o for 8-neighborhood.
- Derivative dd = the number of direction changes in counterclockwise direction needed to rotate from the old direction d_{old} to the new direction d_{new}.

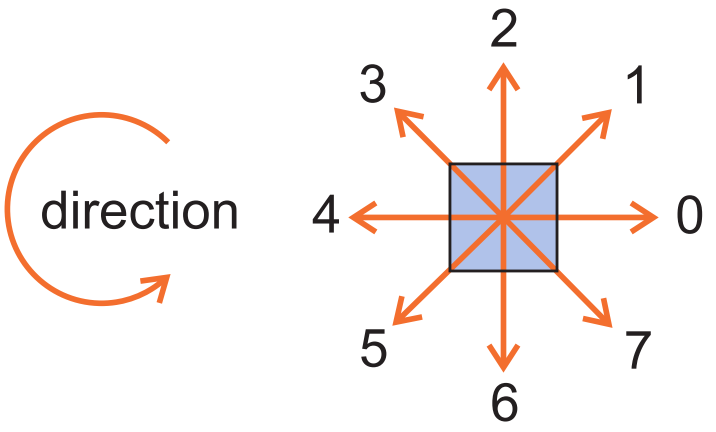
4-neighborhood



if $d_{\mathrm{new}} \geq d_{\mathrm{old}}$ then $dd = d_{\mathrm{new}} - d_{\mathrm{old}}$

if $d_{\mathrm{new}} < d_{\mathrm{old}}$ then $dd = 4 + d_{\mathrm{new}} - d_{\mathrm{old}}$

8-neighborhood

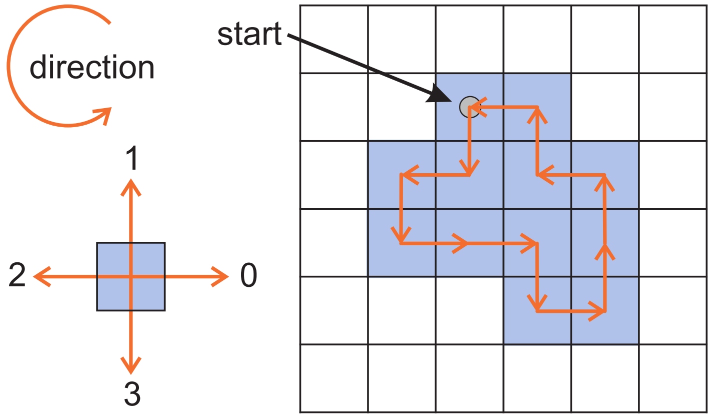


if $d_{\mathrm{new}} \geq d_{\mathrm{old}}$ then $dd = d_{\mathrm{new}} - d_{\mathrm{old}}$

if $d_{\mathrm{new}} < d_{\mathrm{old}}$ then $dd = 8 + d_{\mathrm{new}} - d_{\mathrm{old}}$

Chain code derivative dd, example

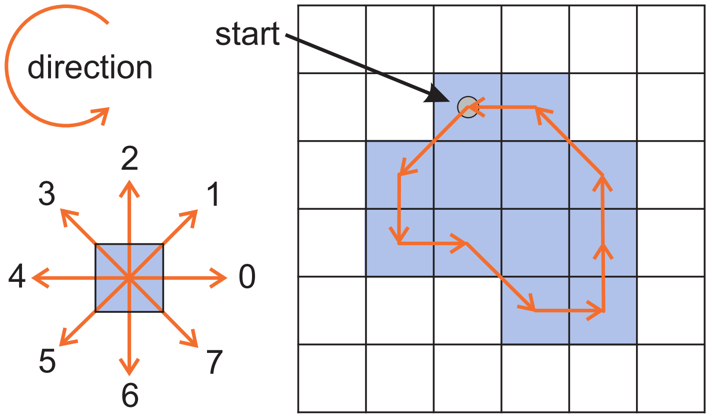
4-neighborhood



if $d_{\mathrm{new}} \geq d_{\mathrm{old}}$ then $dd = d_{\mathrm{new}} - d_{\mathrm{old}}$
if $d_{\mathrm{new}} < d_{\mathrm{old}}$ then $dd = 4 + d_{\mathrm{new}} - d_{\mathrm{old}}$

Chain code: 3 2 3 0 0 3 0 1 1 2 1 2
Derivative dd: 3 1 1 0 3 1 1 0 1 3 1 1

8-neighborhood



if $d_{\mathrm{new}} \geq d_{\mathrm{old}}$ then $dd = d_{\mathrm{new}} - d_{\mathrm{old}}$
if $d_{\mathrm{new}} < d_{\mathrm{old}}$ then $dd = 8 + d_{\mathrm{new}} - d_{\mathrm{old}}$

Chain code: 5 6 0 7 0 2 2 3 4
Derivative dd: 1 2 7 1 2 0 1 1 1

Curve length or region boundary perimeter from a chain code

4-neighborhood chain code: Curve length $=$ $\#$ of chain codes.

8-neighborhood chain code: Curve length $=$ $\#$ of even chain codes $+ \sqrt{2}$ ($\#$ of odd chain codes).

Naturally, 4-neighborhood length $>$ 8-neighborhood length.

Fourier descriptors

- Preprocessing: to express the boundary pixels as a sequence of complex numbers

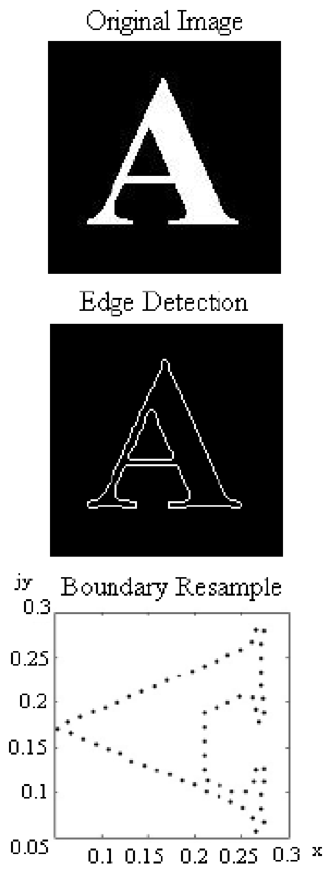
$$s(k) = x(k) + jy(k)\,, \;\; k = 0, 1, \ldots, K-1\,, \;\; k = \sqrt{-1}\,.$$

- The Discrete Fourier Transform (DFT) of this complex sequence is

$$z(u) = \frac{1}{K} \sum_{k=0}^{K-1} s(k)\, e^{\frac{-jxuk}{K}}\,, \quad u = 1, \ldots, K-1\,.$$

- There is a transformation to achieve invariance to scale, rotation (and translation).

$$c(u-2) = \frac{z(u)}{z(1)}\,, \quad u = 2, 3, \ldots, K-1\,.$$



Invariance of Fourier descriptors

Rotation Θ affects all coefficients by the some constant, $s(k)\,e^{j\Theta} \Leftrightarrow z(u)\,e^{j\Theta}$.

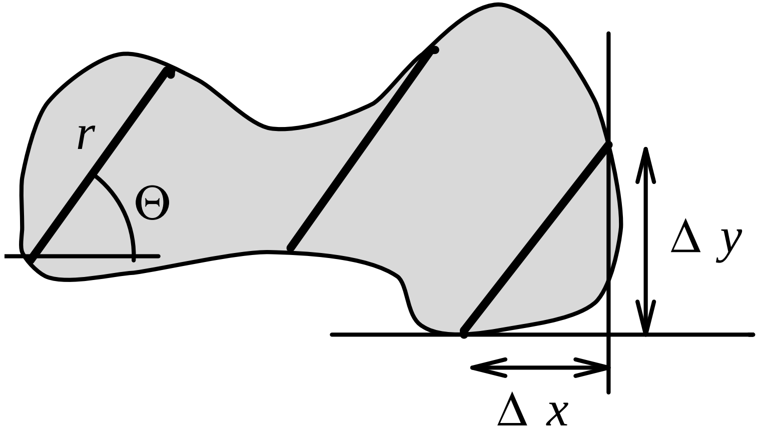
Translation Δ affects zero-th coefficient only, $s(k) + \Delta \Leftrightarrow z(u) + \Delta\,\delta(u)$.

Scaling by α affects all coefficients by the same constant, $\alpha\,s(k) \Leftrightarrow \alpha\,z(u)$.

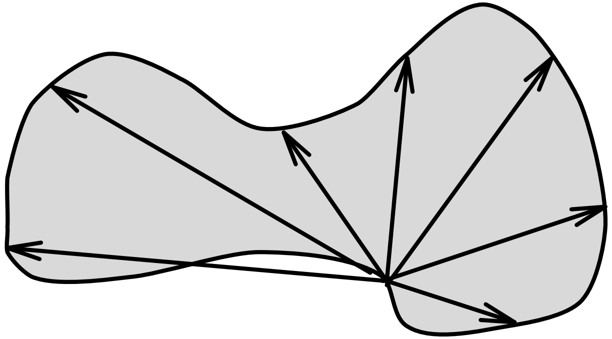
Shift of a starting point by k_0 affects the phase only, $s(k - k_0) \Leftrightarrow z(u)\,e^{-2\pi j\frac{k_0 u}{K}}$.

Chord distribution (1)

- Distribution of lengths and angles of all chords on a contour may be used for shape description.



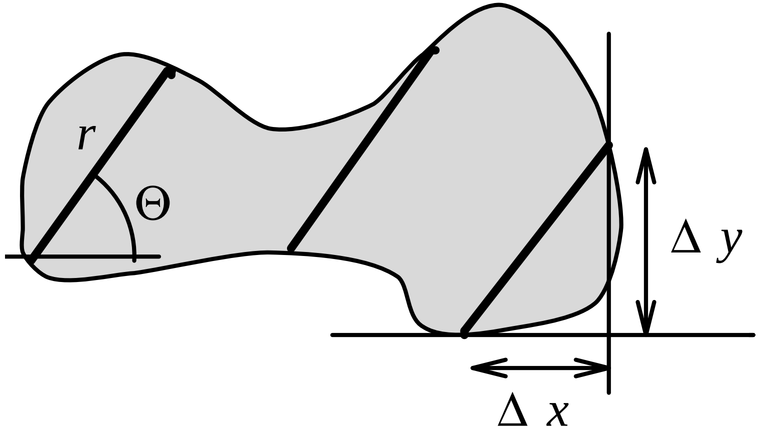
(a)



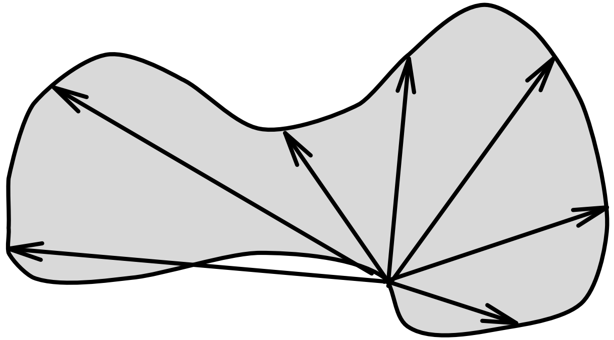
(b)

- $b(x, y) = 1$ contour pixels; $b(x, y) = 0$ other pixels.
- Chord distribution
  $h(\Delta x, \Delta y) = \sum_i \sum_j b(i, j)\, b(i + \Delta x, j + \Delta y)$.

Chord distribution (2)



(a)



(b)

- Radial distribution $h_r(r) = \int\limits_{-\pi/2}^{\pi/2} h(\Delta x, \Delta y)\, r \,\mathrm{d}\,\theta$,

  where $r = \sqrt{\Delta x^2 + \Delta y^2}$, $\theta = \sin^{-1}\left(\frac{\Delta y}{r}\right)$.
- Angular distribution $h_a(\theta) = \int\limits_{0}^{\max(r)} h(\Delta x, \Delta y)\,\mathrm{d}\,r$.
- Combination of $h_r(r)$ and $h_a(\theta)$ is a robust shape descriptor.

Region of interest, matching

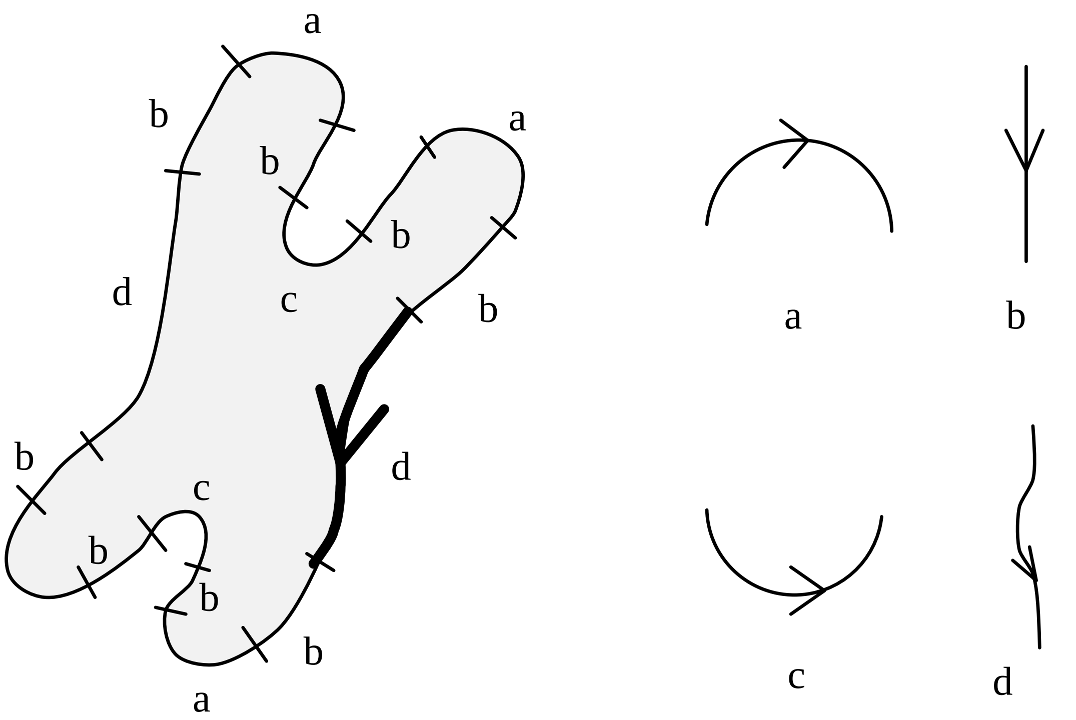
Matching (color) histograms is common, e.g., in tracking.

Let $h(q)$ be a histogram of gray levels q_i, $0 \leq i \leq L$.

Histogram features

- Mean $= \sum_0^L q_i h(q_i)$.
- Energy $= \sum_0^L (h(q_i))^2$.
- Entropy $= \sum_0^L h(q_i) \log_2 h(q_i)$.

Example – structural description of a chromosome



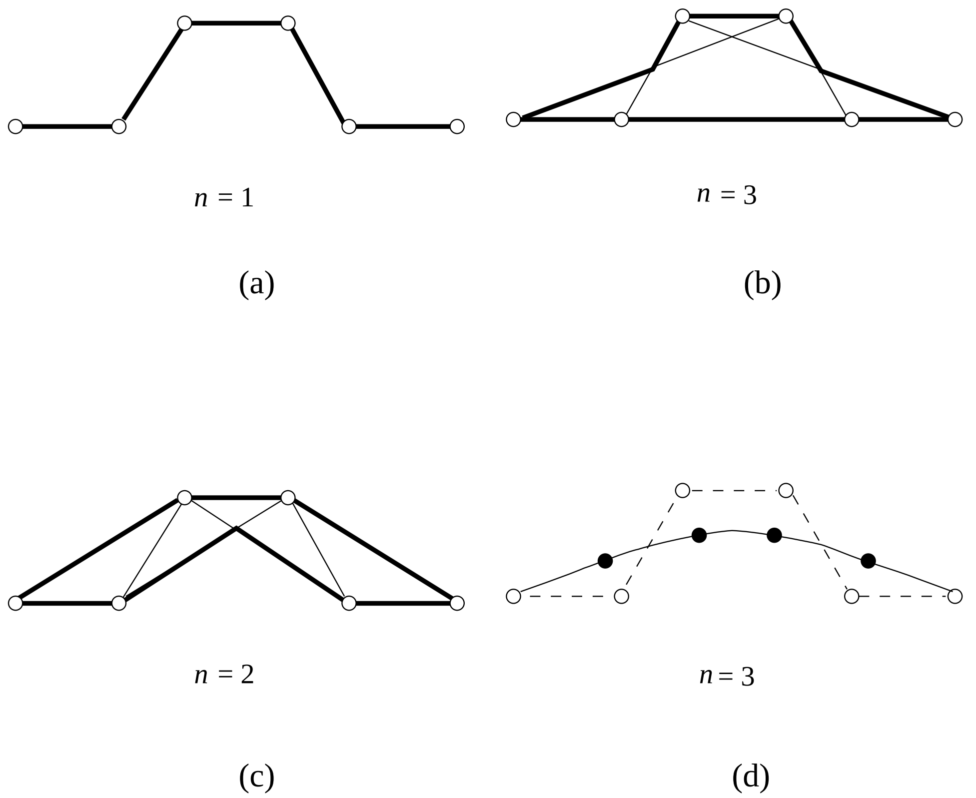
- Structural description of chromosomes by a chain of boundary segments
- Code word: d, b, a, b, c, b, a, b, d, b, a, b, c, b, a, b.

B-spline representation

- B-splines are piecewise polynomial approximation curves.
- B-splines are given by a control polygon.
- A curve is represented by control points.

- B-splines of the third-order are the most common (cope with the curvature change).
- Endpoints fixed by two control points.
- Shape controlled by two control points.

B-spline representation



A spline curve is always positioned inside a convex $n+1$-polygon for a B-spline of the n^{th} order.

B-spline (continued)

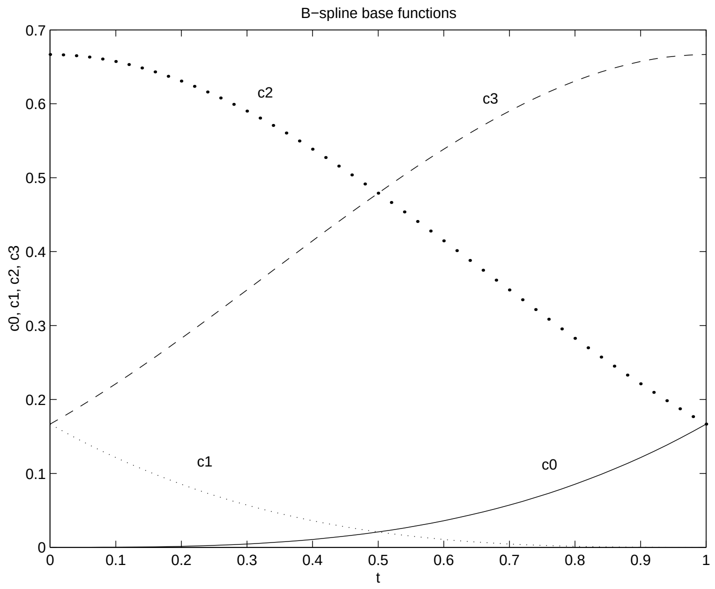
- Let $\mathbf{x}_i$, $i = 1, \ldots, n$ be points of a B-spline interpolation curve $\mathbf{x}(s)$.
- The parameter s changes linearly between points $\mathbf{x}_i$. That is, $\mathbf{x}_i = \mathbf{x}(i)$.
- B-splines $\mathbf{x}(s) = \sum_{i=0}^{n+1} \mathbf{v}_i \, B_i(s)$, where
  - $\mathbf{v}_i$ are control points of a control polygon.
  - n points $\mathbf{x}_i \Rightarrow n + 2$ points $\mathbf{v}_i$.
  - $B_i(s)$ are base functions.
- The start point $\mathbf{v}_0$ and the end point $\mathbf{v}_{n+1}$ are constrained by binding conditions. If the curvature to be zero at the curve start and the curve end then

$$
\begin{aligned}
\mathbf{v}_0 &= 2\,\mathbf{v}_1 - \mathbf{v}_2 \\
\mathbf{v}_{n+1} &= 2\,\mathbf{v}_n - \mathbf{v}_{n-1}
\end{aligned}
$$

B-spline base functions

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

$$
\begin{aligned}
\mathbf{x}(s) &= C_{i-1,3}(s)\mathbf{v}_{i-1} + C_{i,2}(s)\mathbf{v}_i \\
&+ C_{i+1,1}(s)\mathbf{v}_{i+1} + C_{i+2,0}(s)\mathbf{v}_{i+2}
\end{aligned}
$$



- Base functions are non-negative.
- Shape of base functions induces only their local influence to the shape of the approximated function.