Gray-level morphology.
Shape description

Today

- Gray scale morphology: an overview of basic principles
- We will start shape description and representation (Sonka Chapter 8)

- Note: some slides of this lecture are adapted from Brian Morse, Computer Vision I, http://morse.cs.byu.edu/650/home/index.php

Gray-scale morphology

- basic ideas of binary morphology extend to grayscale
- logical operations convert to similar arithmetic ones: union becomes maximum, intersection becomes minimum, etc.

Images, Functions, and Umbras

- We will discuss greyscale morphology by first considering one-variable functions (signals).
- The umbra of a function/signal/image f (x) is the set of all positions/values (x, v) such that value v is less than or equal to

$$f(x):\{(x,v) \mid v \le f(x)\}$$

- This construct allows us to now consider greyscale functions as sets, like we did for binary images.

Examples of umbra

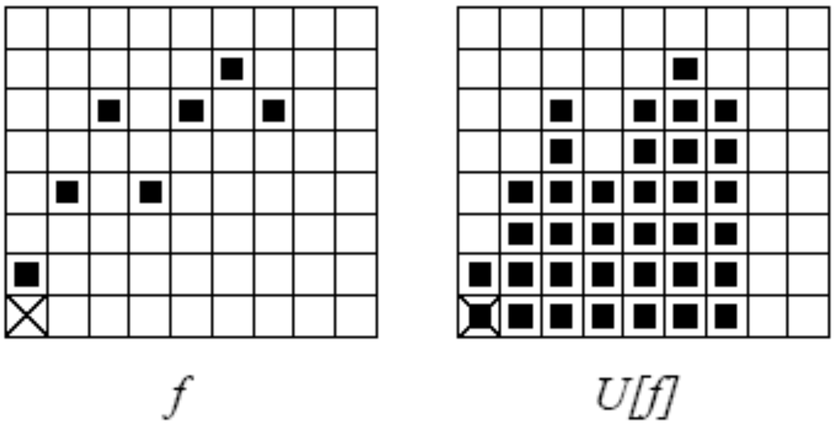


Figure 13.14: Example of a 1D function (left) and its umbra (right).

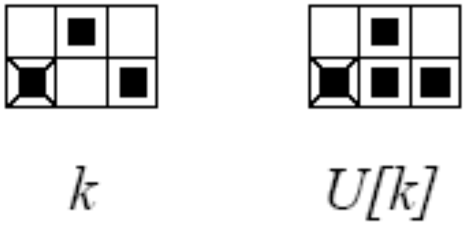


Figure 13.15: A structuring element: 1D function (left) and its umbra (right).

Grayscale dilation and erosion

- Grayscale morphology can thus be thought of as binary morphology of the umbras:

$$\text{Umbra}\,(A \oplus_g B) = \text{Umbra}\,(A) \oplus \text{Umbra}\,(B)$$
$$\text{Umbra}\,(A \ominus_g B) = \text{Umbra}\,(A) \ominus \text{Umbra}\,(B)$$

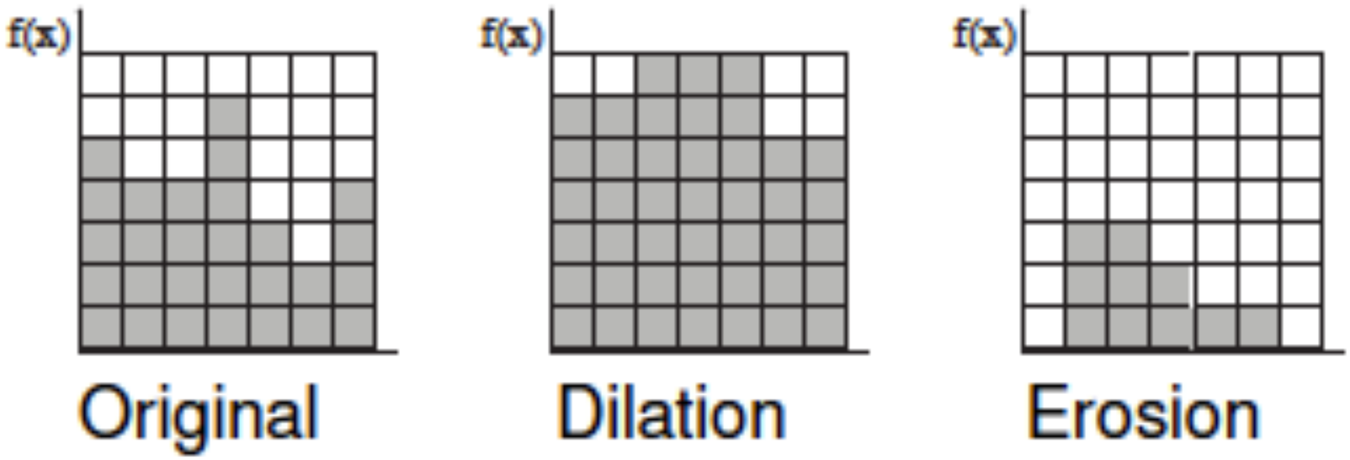
Example

- Let's dilate a 1-d function f(x) by the structuring element g(x) = 1 over the range $\{-1 \leq x \leq 1\}$, 0 otherwise.

- Each point in the structuring element causes the image to be increased by one intensity level (intensity dilation).
- However, the spatial spread of the structuring element causes the result at each output position x to be the maximum of the 3-pixel neighborhood. The net effect of these together is (for this case)

- $f(x) \oplus_g g(x) = \max\{f(x-1)+1, f(x)+1, f(x+1)+1\}$

Example (3-wide structuring element of all 1s):



Gray-scale dilation

$$f \oplus_g g = \max_z \{f(x - z) + g(z)\}$$

1. reflect the structuring element,
2. position the structuring element at position x
3. pointwise add the structuring element over the neighborhood, and
4. take the maximum of that result over the neighborhood.

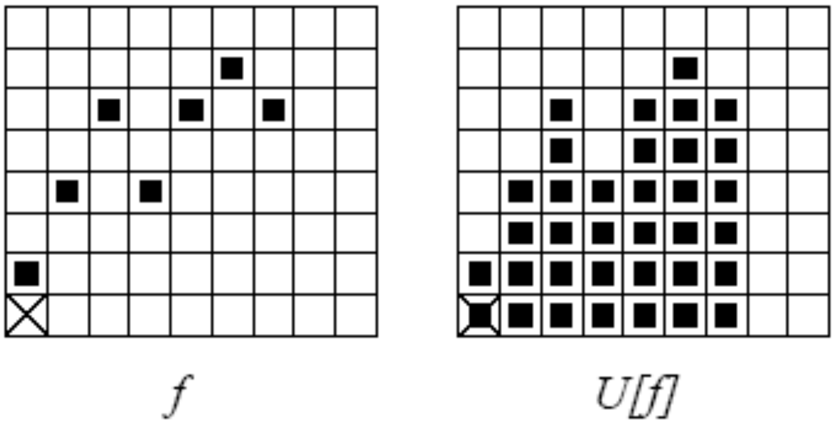


Figure 13.14: Example of a 1D function (left) and its umbra (right).

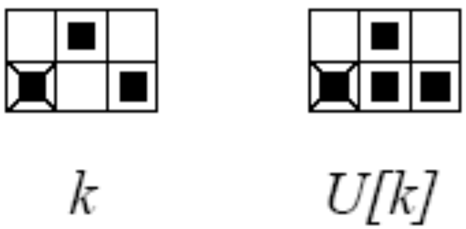


Figure 13.15: A structuring element: 1D function (left) and its umbra (right).

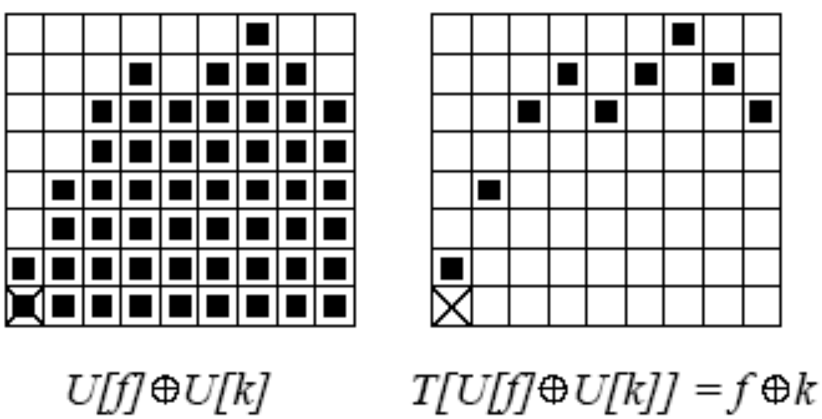


Figure 13.16: 1D example of gray-scale dilation. The umbras of the 1D function f and structuring element k are dilated first, $U[f] \oplus U[k]$. The top surface of this dilated set gives the result, $f \oplus k = T\left[U[f] \oplus U[k]\right]$.

Grey-scale erosion

$$f \ominus_g g = \min_z \{f(x+z) - g(z)\}$$

1. position the structuring element at position x
2. pointwise subtract the structuring element over the neighborhood, and
3. take the minimum of that result over the neighborhood.

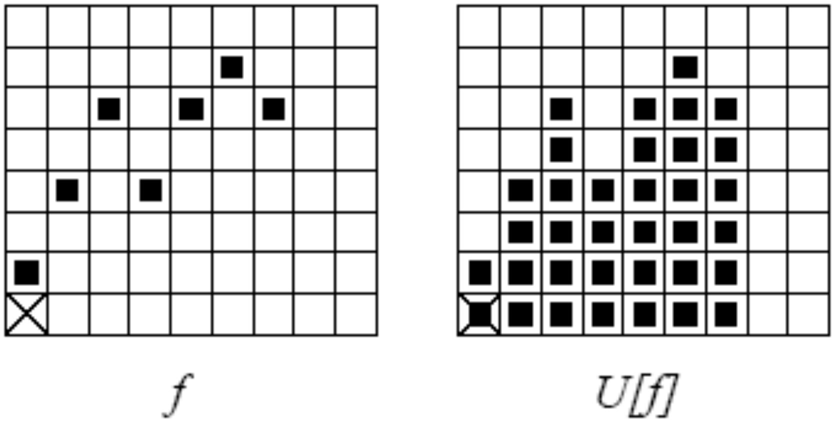


Figure 13.14: Example of a 1D function (left) and its umbra (right).

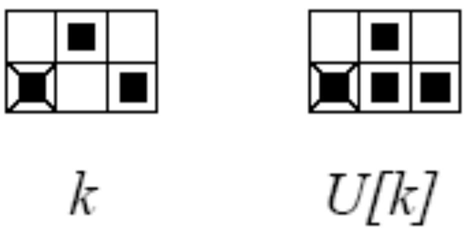


Figure 13.15: A structuring element: 1D function (left) and its umbra (right).

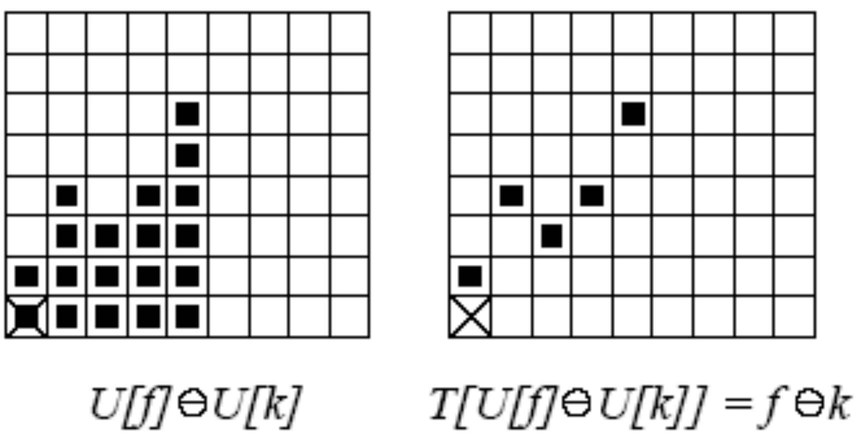
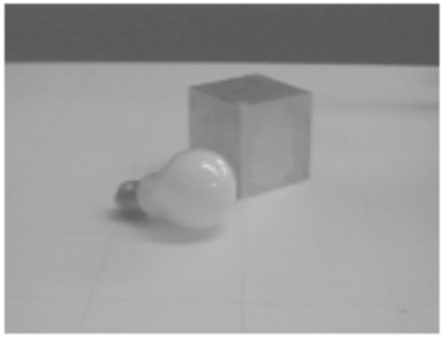


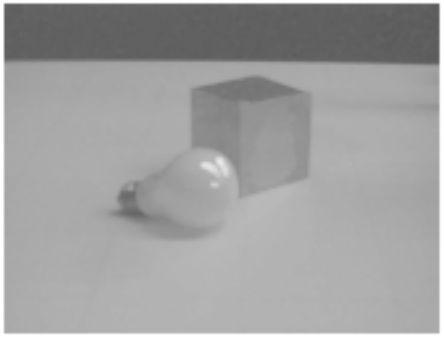
Figure 13.17: 1D example of gray-scale erosion. The umbras of 1D function f and structuring element k are eroded first, $U[f] \ominus U[k]$. The top surface of this eroded set gives the result, $f \ominus k = T\left[U[f] \ominus U[k]\right]$.

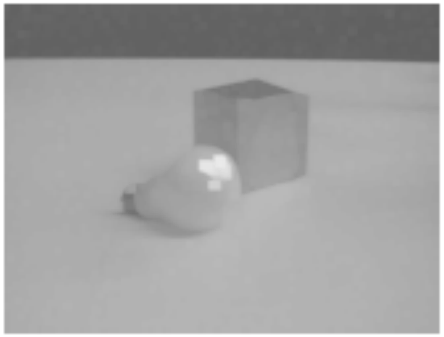
Umbras of 2D images

- We can extend these ideas to 2-dimensional images I(x, y) by considering the sets of positions (x, y) and possible intensities
- $I:\{(x,y,v \mid v \leq I(x,y)\}$

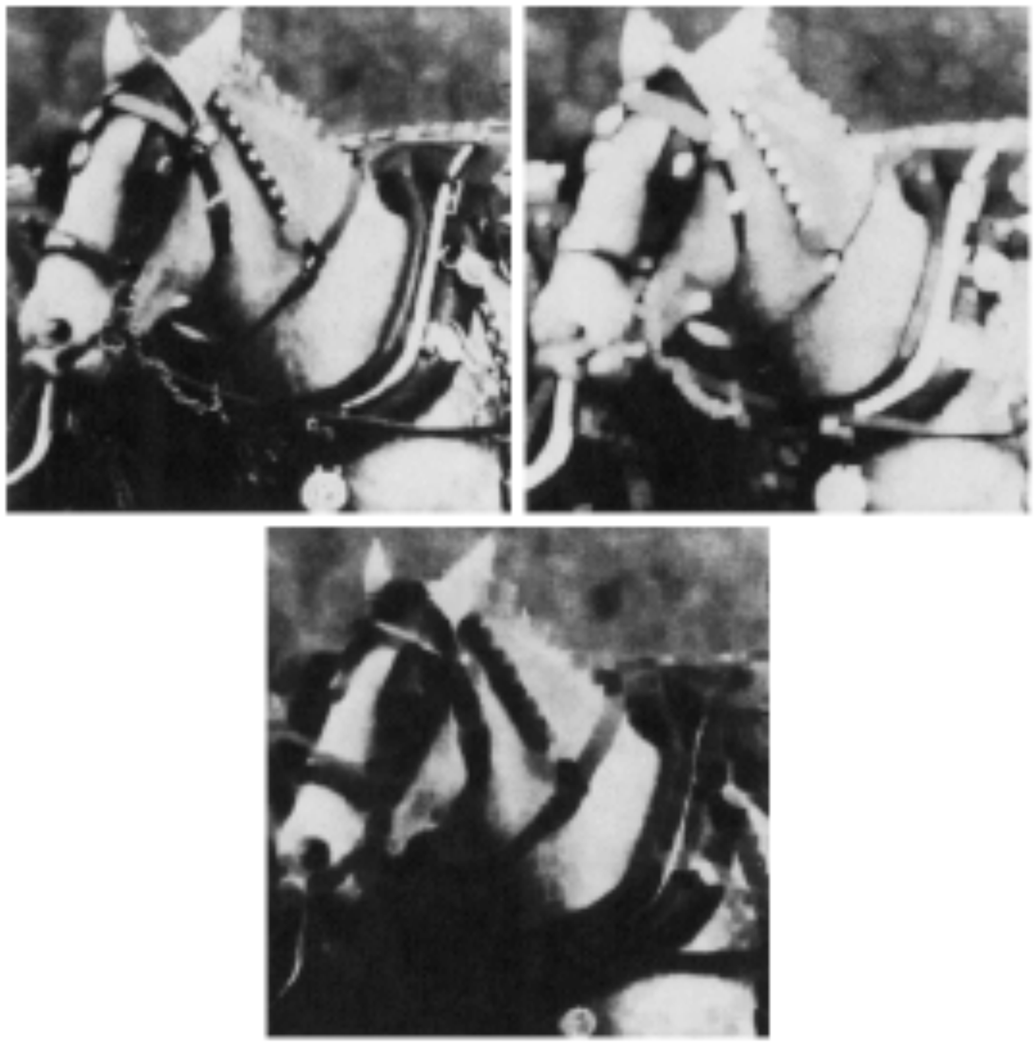
Gray-scale image dilation: example 1







Gray-scale image dilation and erosion: example 2



a b
c

FIGURE 9.29 (a) Original image. (b) Result of dilation. (c) Result of erosion. (Courtesy of Mr. A. Morris, Leica Cambridge, Ltd.)

Gray-scale opening

Erosion followed by dilation:

$$f \circ_g g = (f \ominus_g g) \oplus_g g$$

or for a constant structuring element:

$$f \circ_g g = \max_{(a \in g)} \min_{(b \in g)} f(x - a + b)$$

Gray-scale closing

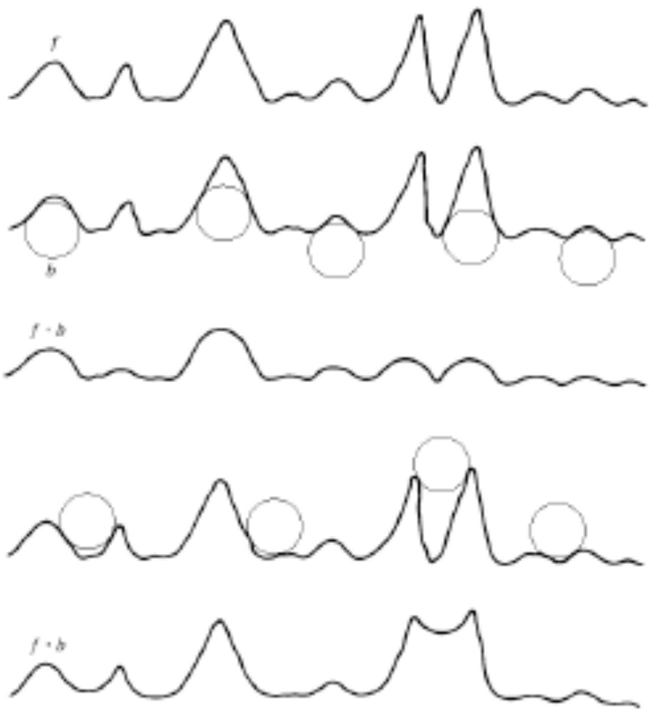
Dilation followed by erosion:

$$f \bullet_g g = (f \oplus_g g) \ominus_g g$$

or for a constant structuring element:

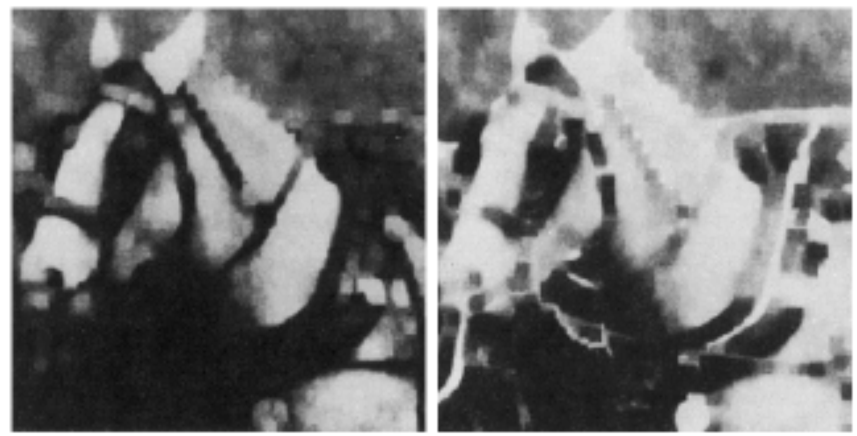
$$f \bullet_g g = \min_{(a \in g)} \max_{(b \in g)} f(x + a - b)$$

Gray-scale opening and closing: example



a
b
c
d
e

FIGURE 9.30 (a) A gray-scale scan line. (b) Positions of rolling ball for opening. (c) Result of opening. (d) Positions of rolling ball for closing. (e) Result of closing.



b

GURE 9.31 (a) Opening and (b) closing of Fig. 9.29(a). (Courtesy of Mr. A. Morris, eica Cambridge, Ltd.)

Shape representation

- Represents the process of transition from the image space to the feature space
- Region-based
- Contour-based

Also

Motion representation

Event representation etc.

Shape descriptors

- Desirable properties:
  - Compact representation
  - Invariant to as many transformations as possible (translation, rotation, scale, etc.)
  - Useful for matching—
    relatively insensitive to small variations

Region description and representation

Reading:

- Sonka 8.3
- optional
  - Bobick and Davis, “The recognition of human movement using temporal templates”, IEEE Transactions on Pattern Analysis and Machine Intelligence, March 2001.

Region Descriptors

- Reading: 8.3.1
- A descriptor is a number or set of numbers that describes some property of a shape
- Can't usually reconstruct the shape, but can be used to distinguish shapes
- Examples:
  - Area
  - Perimeter
  - Compactness
  - Eccentricity
  - Euler Number (count components and holes)
- Practical considerations:
  - Inside or outside borders?
  - 4- vs. 8-connected perimeters?
  - Complicated regions?

Compactness

$$compactness = \frac{(region\_border\_length)^2}{area} \qquad (6.40)$$

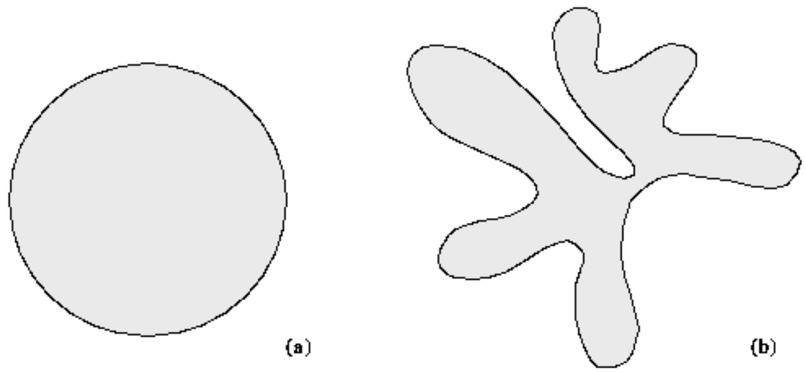


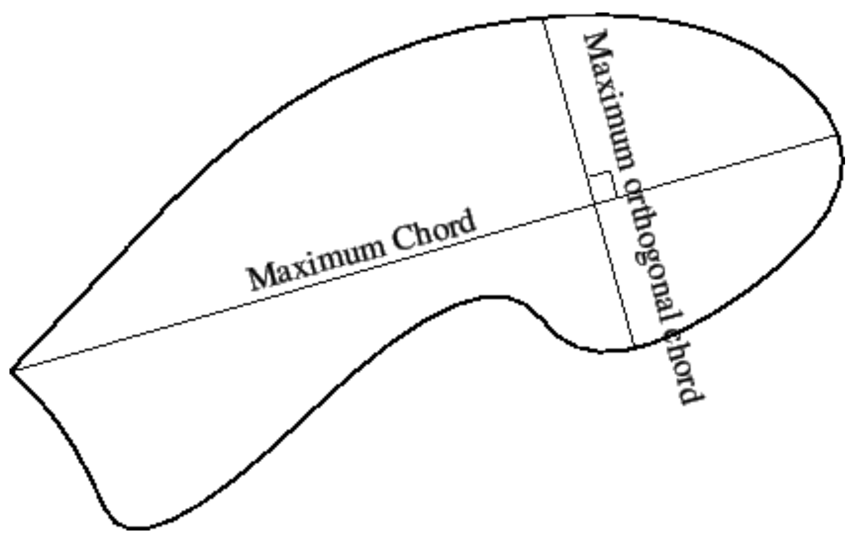
Figure 6.25 *Compactness: (a) Compact, (b) non-compact.*

The compactness measure is independent of linear transformations if the outer boundary is measured

Most compact shape: the circle : 4π compactness

Eccentricity

- The ratio of the longest chord compared to the chord perpendicular to it;
- measure of how non-circular a shape is



Rectangularity

http://www.mobileye.com/

Elongation

Can be measured as the ratio between the length and width of the region bounding rectangle

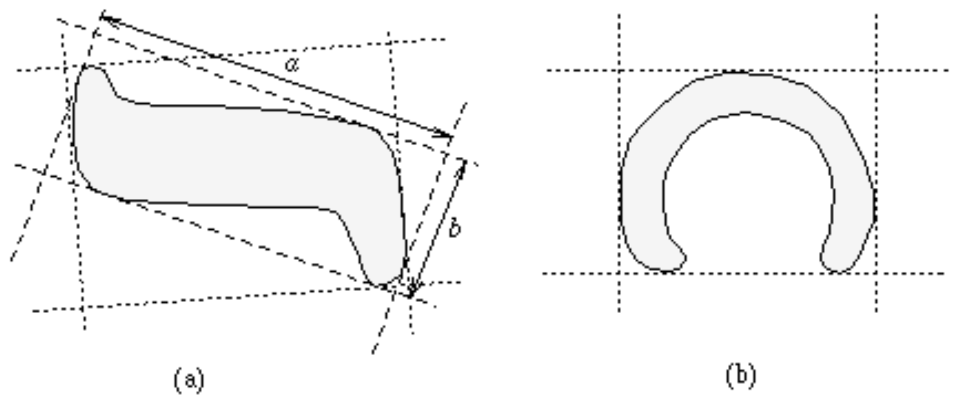


Figure 6.24 *Elongatedness: (a) Bounding rectangle gives acceptable results, (b) bounding rectangle cannot represent elongatedness.*

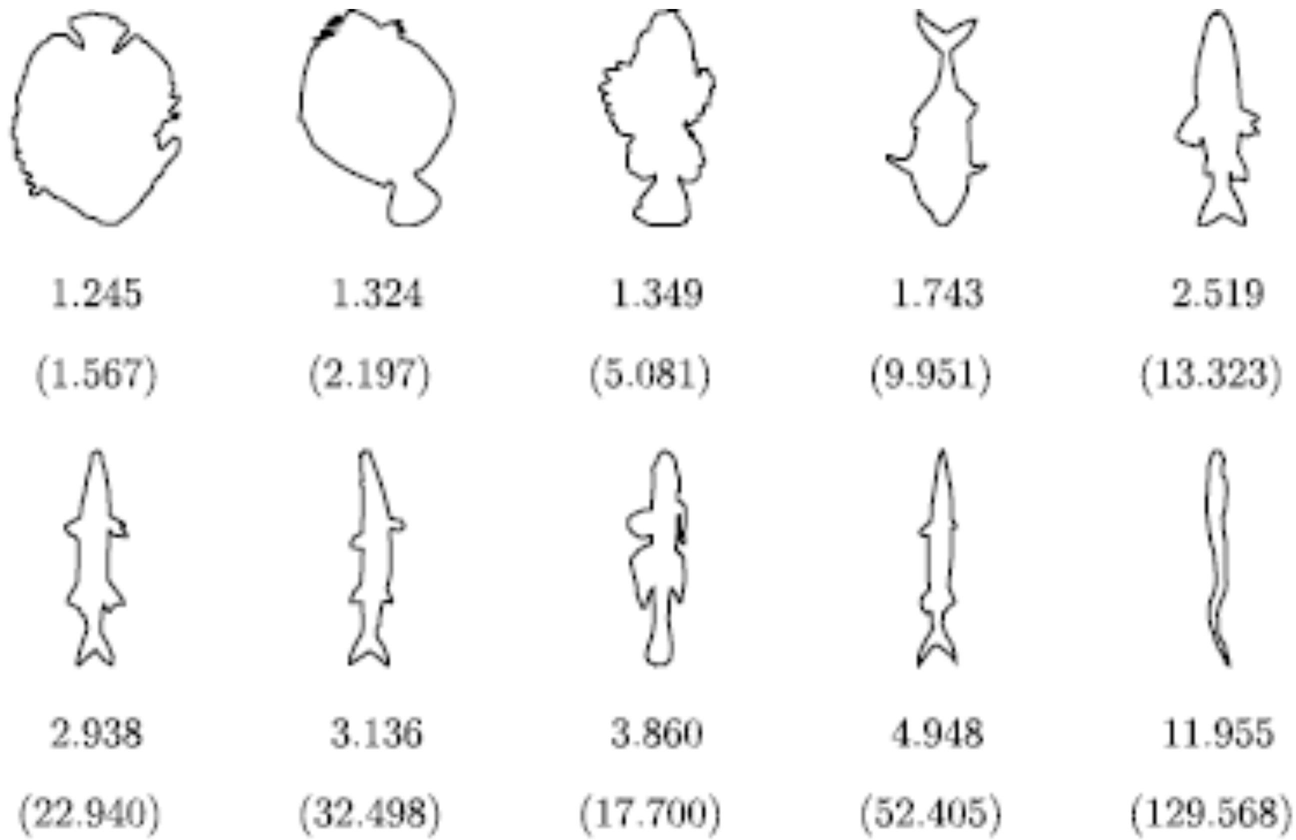
This measure can not be used in curved regions!

Elongation (morphological measure)

$$\text{Elongation} = \frac{\text{Area}}{(2d)^2}$$

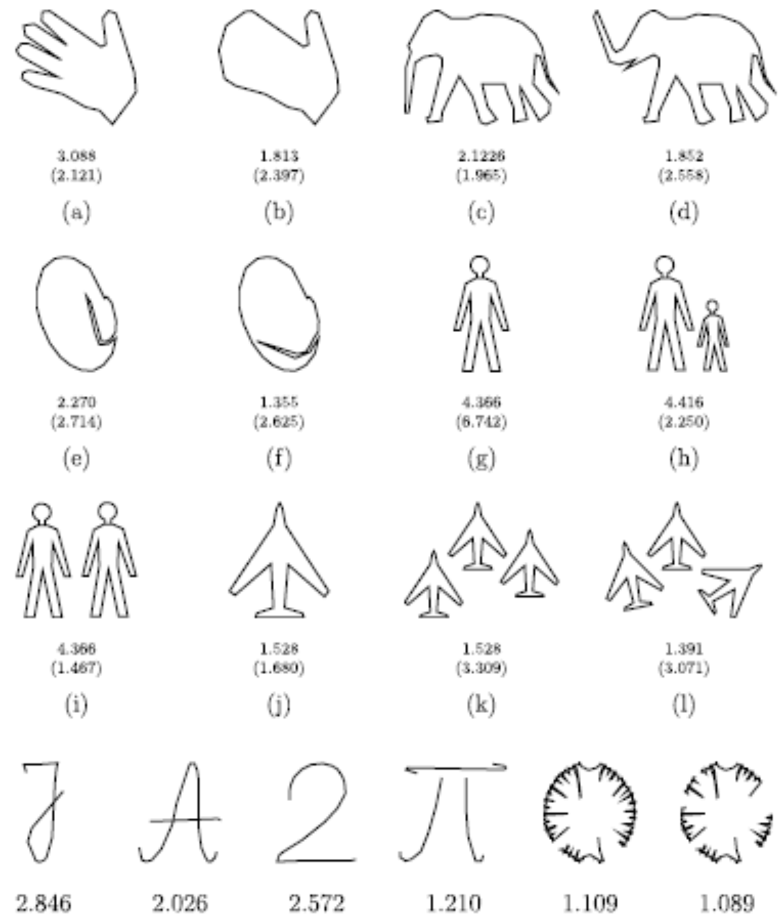
where d is the maximum number of erosions before the shape disappears (half the width of the object)

Examples of shapes that can be described via elongation (1)



Examples of shapes that can be described via elongation (2)

Stojmenovic, Žunic (2008)
Measuring elongation from shape boundary



Concavities

- Read section 8.3.3
- Differences between object and its convex hull are holes or concavities.
- For complex shapes: We can build a hierarchical representation of concavities (concavity tree)

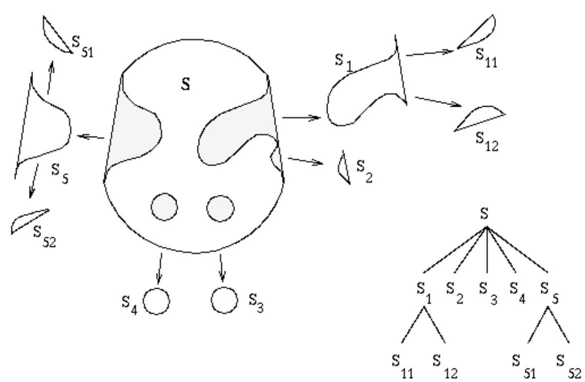
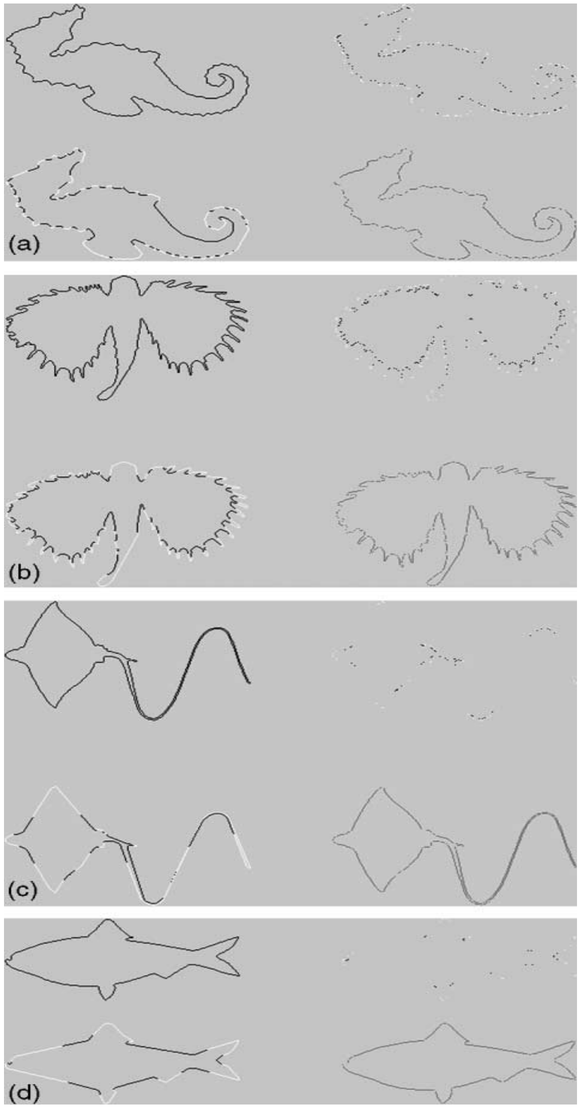


Figure 6.30 *Concavity tree construction: (a) Convex hull and concave residua, (b) concavity tree.*

Contour partitioning via concavity analysis

Cronin, “Visualizing concave and convex partitioning of 2D contours”,
Pattern Recognition Letters, 2003.



Statistical region descriptors: moments

- Read section 8.3.2
- Region moment representations interpret a normalized gray level image function as a probability density of a 2D random variable.
- Properties of this random variable can be described using statistical characteristics - **moments**.
- Assuming that non-zero pixel values represent regions, moments can be used for binary or gray level region description.

$$m_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} i^p j^q f(i,j) \qquad (6.42)$$

- where i,j are the region point co-ordinates (pixel co-ordinates in digitized images).

Moments

- Represent a global description of a shape layout
- Combine area, compactness, irregularity, and higher order descriptions together
- Associated with statistical pattern recognition
- Not able to handle shape occlusion

Central moments

- The n-th central moment for probability density function f:

$$\mu_n = \int_{-\infty}^{\infty} (x-\mu)^n \; f(x) \; dx$$

- Or for arbitrary discrete functions:

$$\mu_n = \sum_{x=1}^{N} (x-\mu)^n f(x)$$

- First two central moments:

$$\mu_0 = \text{Area}$$

$$\mu_1 = 0$$

Higher order moments

Mean	μ	$=$	m_1 / m_0
Variance	σ^2	$=$	μ_2 / m_0
Skew			μ_3 / m_0
Kertosis			μ_4 / m_0

Central and normalized bidimensional moments

$$\mu_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} (x - \overline{x})^p (y - \overline{y})^q P_{xy}$$

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

$$\gamma = \frac{p+q}{2} + 1 \qquad \forall (p+q) \geq 2$$

Symmetry analysis with moments

Figure 1.5: Axes of symmetry for typed characters.

Table 1.1: Typed characters η_{pq} values indicating symmetry.

Character	η_{11}	η_{20}	η_{02}	η_{21}	η_{12}	η_{30}	η_{03}
M	0	+	+	-	0	0	-
C	0	+	+	0	+	+	0

Example of shape descriptors applied to motion analysis

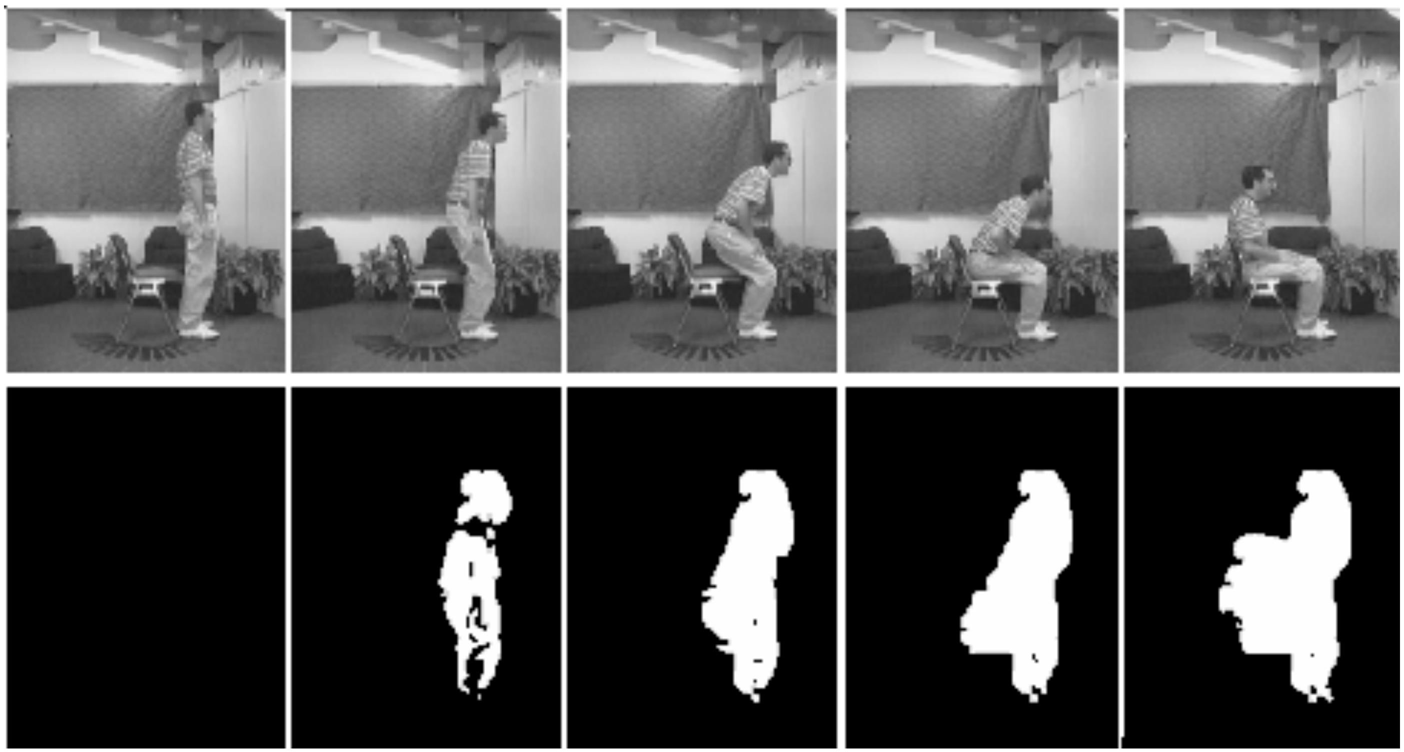
Bobick and Davis “The recognition of human movement using temporal templates”, IEEE Transactions on Pattern Analysis and Machine Intelligence, March 2001.

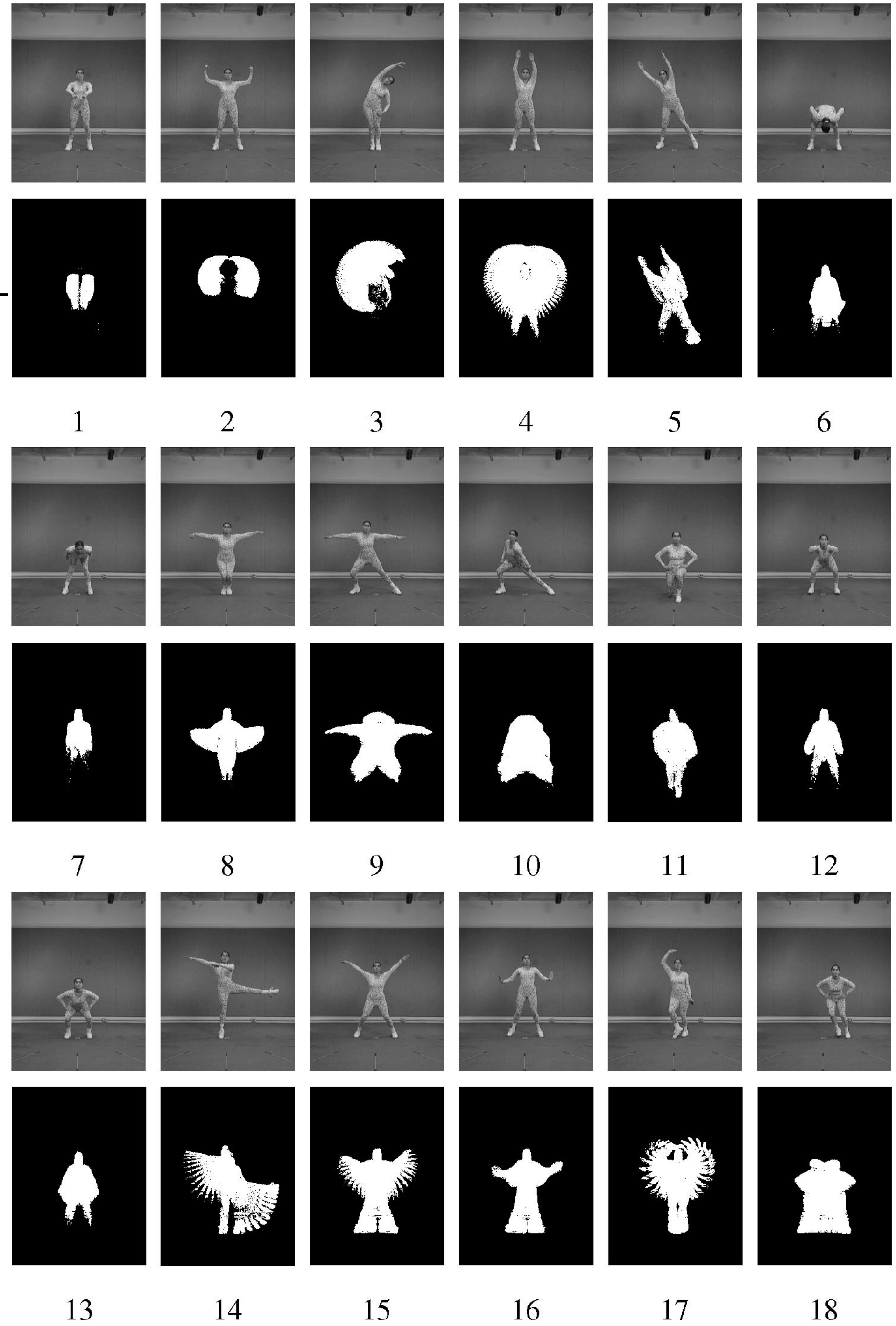
Main ideas:

- analyzing the shape of motion leads to action recognition
- the shape of motion is considered separately from the shape of the object in motion (here, a human silhouette)

Motion energy image (MEI)

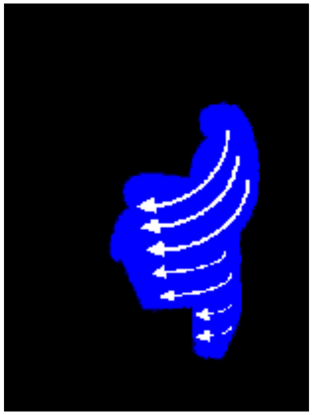
- a representation of the spatial distribution of motion (‘where’)

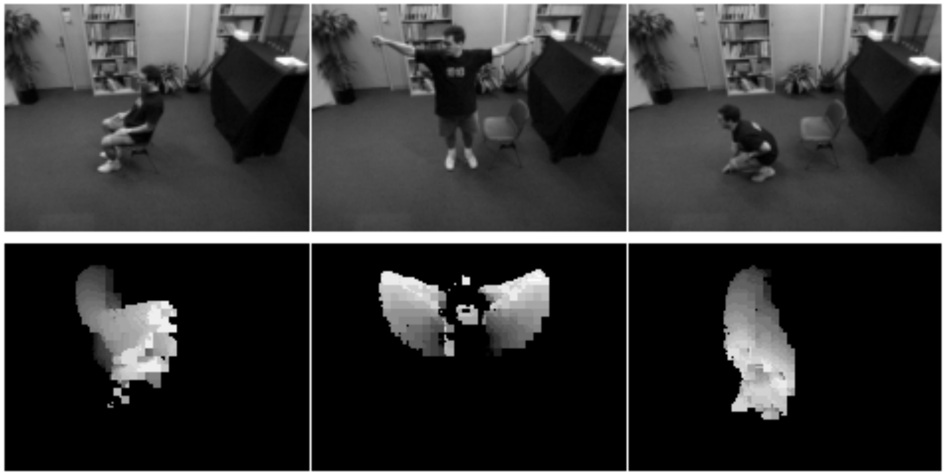


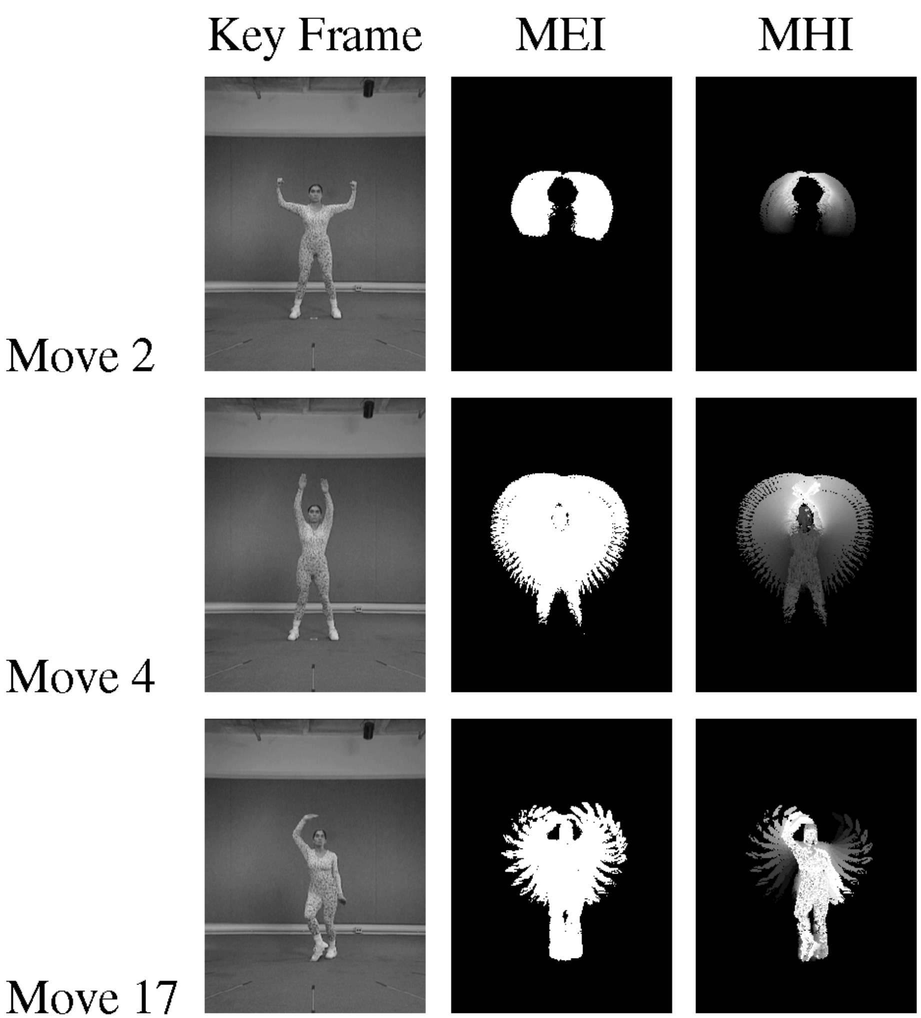

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18

Motion History Image (MHI)

- Describes how the motion evolves over a predefined length of time
- pixel intensity is a function of the motion history at that location, where brighter values correspond to more recent motion.





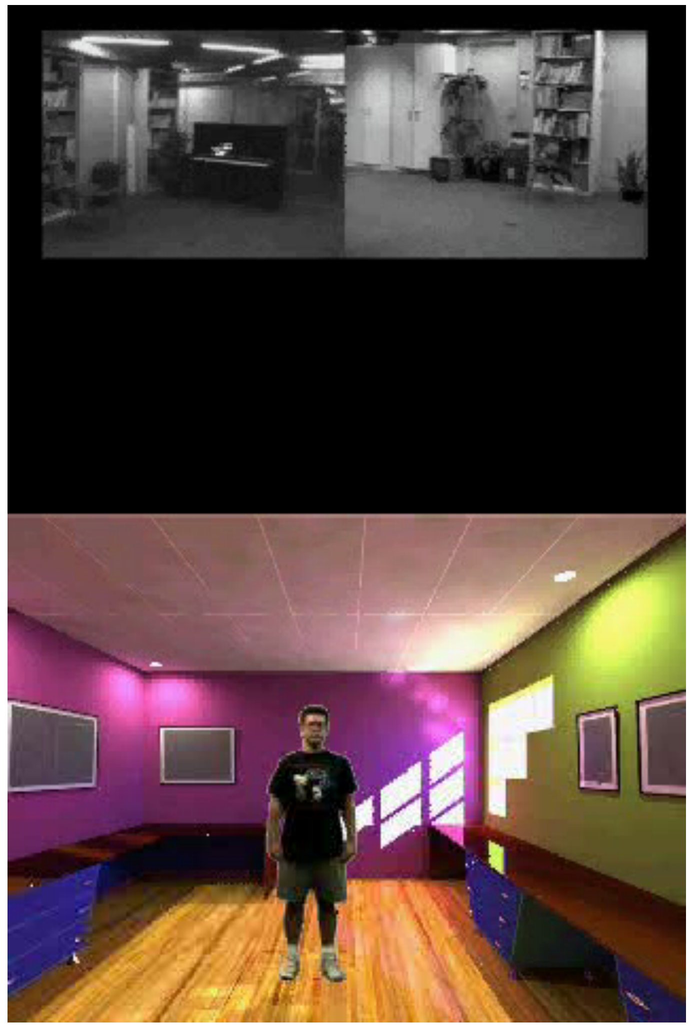

Key Frame
MEI
MHI
Move 2
Move 4
Move 17

How are these two templates described?

- 7 Hu moments are computed for each motion template/view; the Hu moments are invariant to rotation/scale/translation

$$
\begin{aligned}
I_1 &= \eta_{20} + \eta_{02} \\
I_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
I_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
I_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
I_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} \\
&\quad - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
I_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})] \\
I_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} \\
&\quad + \eta_{03})^2] + (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} \\
&\quad + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned}
$$

Motion recognition process



Contour-based shape description

- Reading
  - Sonka 8.2
  - Optional: Jeong and Radke, elliptical Fourier Descriptors

Simple geometric descriptors

- Boundary length (perimeter)
- Bending energy
  - Basic Idea: How much work do you have to do to bend a straight line into the shape?
  - Calculation:

$$\int_0^1 \kappa(s)^2 \; ds$$

    where $\kappa(s)$ is the curvature at point s

- Histograms of geometric properties
  - Orientations
  - Curvature
  - ...

Contour encoding: chains

- Encode the border of the region only.
- Only need to encode relative direction around the border.

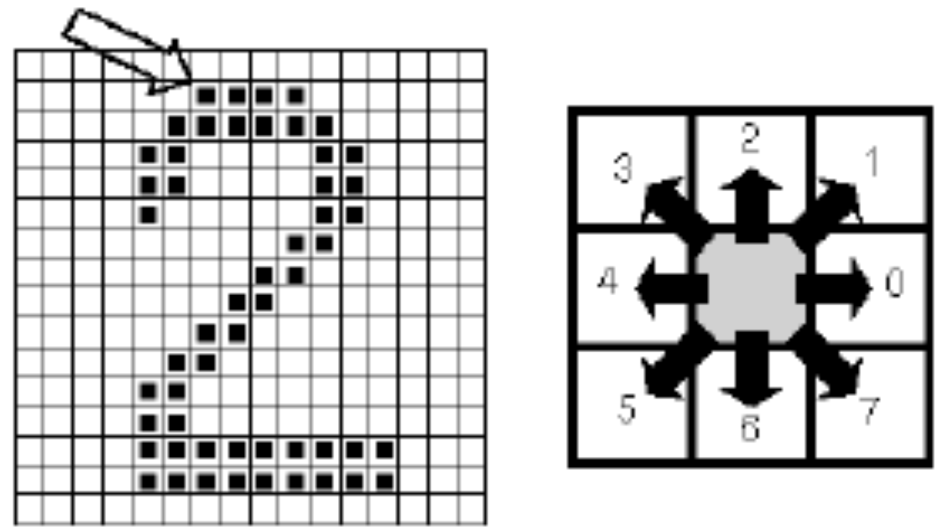
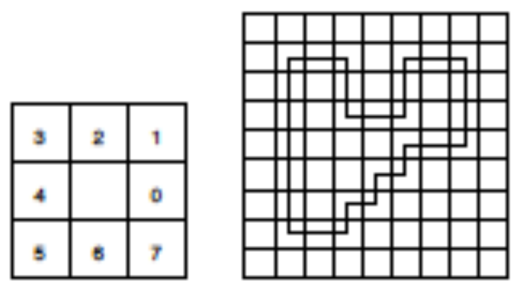


Figure 3.1 An example chain code; the reference pixel is marked by an arrow: 00077665555556600000006444444442221111112234445652211.

Chain codes and variations

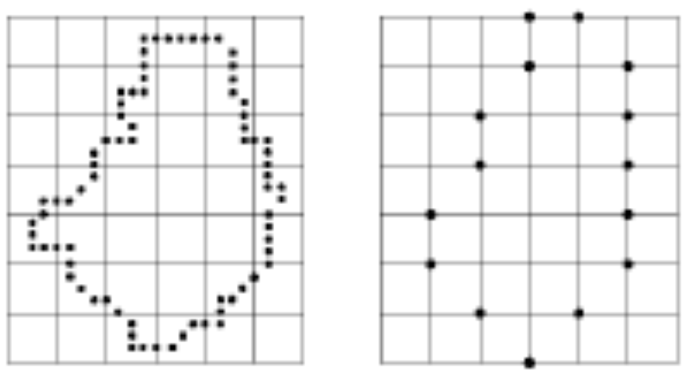
- Chain code: encode direction around the border between pixels
- Differential chain code: *change* in direction around the border (differences between chain code numbers modulo 4 or 8)
- Shape number: differential CCs normalized for starting point– rotate differential chain code to be as small a number as possible



Chain	0 0 6 6 0 0 2 2 0 0 6 6 6 4 4 6 4 6 4 6 4 4 2 2 2 2 2 2
Differential	6 0 6 0 2 0 2 0 6 0 6 0 0 6 0 2 6 2 6 2 6 0 6 0 0 0 0 0
Shape No.	0 0 0 0 0 6 0 6 0 2 0 2 0 6 0 6 0 0 6 0 2 6 2 6 2 6 0 6

Chain codes: smoothing and resampling

- Problem:
  Pixel grid and noise cause change in chain code (and its length)
- Approach:
  Smooth the shape and/or resample to some fixed number of points (code length)

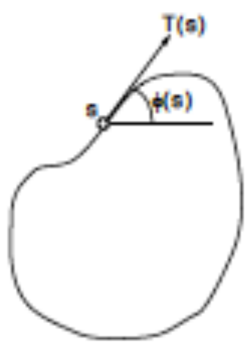


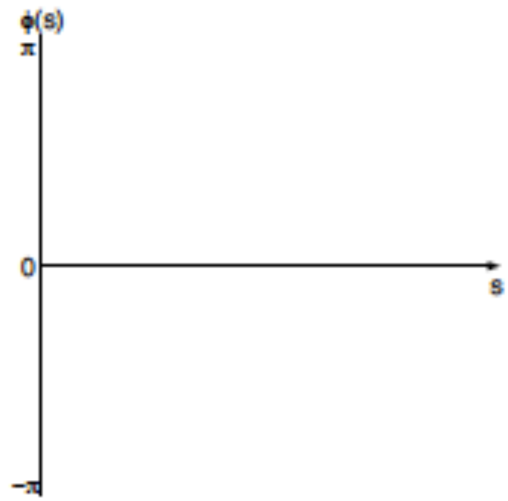
a b
c d

FIGURE 11.2 (a) Digital boundary with resampling grid superimposed. (b) Result of resampling. (c) 4-directional chain code. (d) 8-directional chain code.

Tangential Representations: ψ-s curves

- Encodes the tangent angle as a function of arclength
- Similar to differential chain codes, but not limited to grid (average orientation over small sections of the boundary)
- Sample a fixed number of points around the boundary



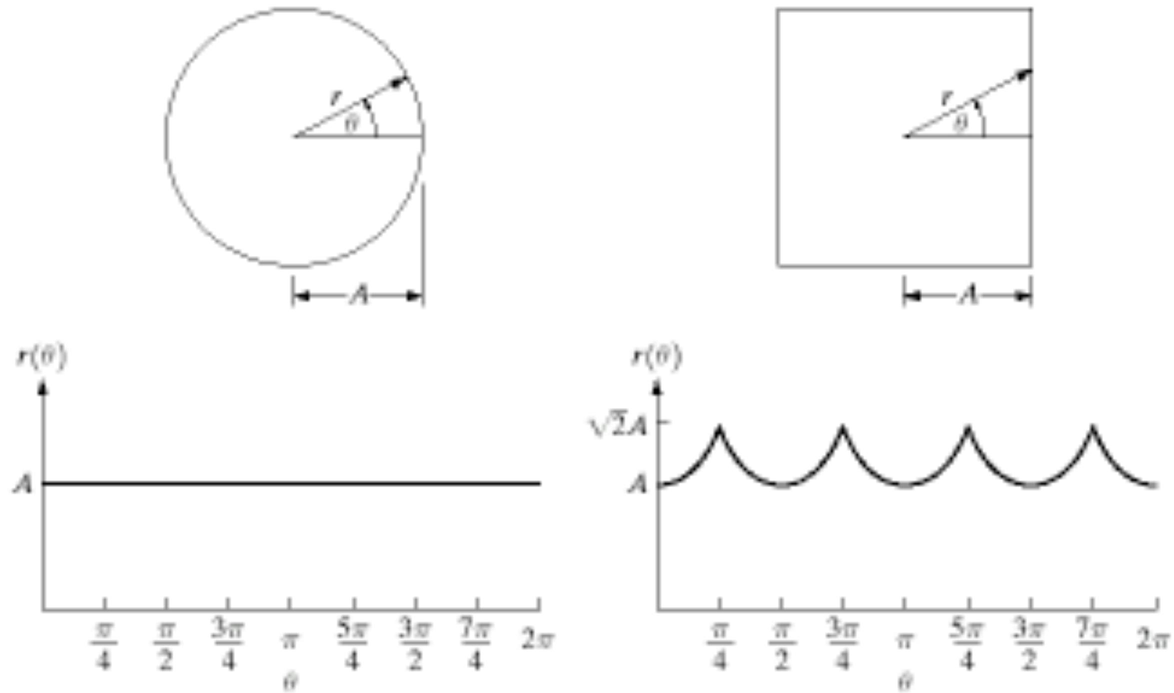


Radial Representations: *r*-*s* Curves

- Encodes the distance from the center as a function of arclength

a b

FIGURE 11.5 Distance-versus-angle signatures. In (a) $r(\theta)$ is constant. In (b), the signature consists of repetitions of the pattern $r(\theta) = A \sec\theta$ for $0 \leq \theta \leq \pi/4$ and $r(\theta) = A \csc\theta$ for $\pi/4 < \theta \leq \pi/2$.



Points of extreme curvature

- Another approach to describing a shape is to decompose it into parts based on *extrema of boundary curvature*
- Decompose into parts
  $\rightarrow$ compare parts
  $\rightarrow$ build relationship graphs
- Some evidence that this corresponds to human perception

Convex hulls

- Build the convex hull of the shape, look at where the shape touches its convex hull—gives you an idea of the "extremal" points or sections of the curve.

a b

FIGURE 11.6
(a) A region, S, and its convex deficiency (shaded).
(b) Partitioned boundary.

Fourier descriptors

- Can think of the boundary pixels (x_k, y_k) as a curve in the complex plane:

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

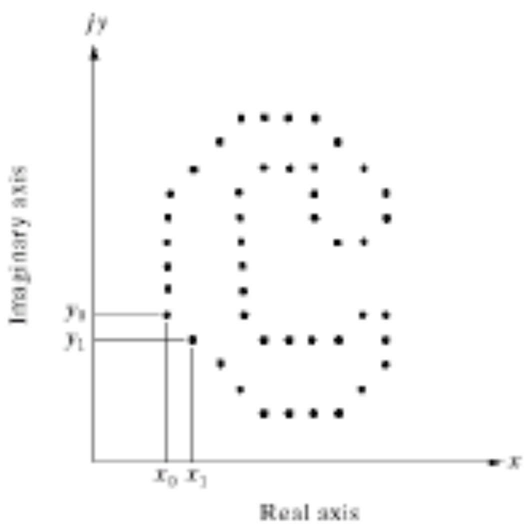


FIGURE 11.13 A digital boundary and its representation as a complex sequence. The points (x_0, y_0) and (x_1, y_1) shown are (arbitrarily) the first two points in the sequence.

Fourier descriptors (cont'd)

- A *Fourier descriptor* is the Fourier Transform of the complex-valued boundary curve:

$$a(\nu) = \mathcal{F}(s(k))$$

- Relatively insensitive to transformations:

Transformation	Boundary	Fourier Descriptor
Identity	$s(k)$	$a(u)$
Rotation	$s_r(k) = s(k)e^{j\theta}$	$a_r(u) = a(u)e^{j\theta}$
Translation	$s_t(k) = s(k) + \Delta_{xy}$	$a_t(u) = a(u) + \Delta_{xy}\delta(u)$
Scaling	$s_s(k) = \alpha s(k)$	$a_s(u) = \alpha a(u)$
Starting point	$s_p(k) = s(k - k_0)$	$a_p(u) = a(u)e^{-j2\pi k_0 u/K}$

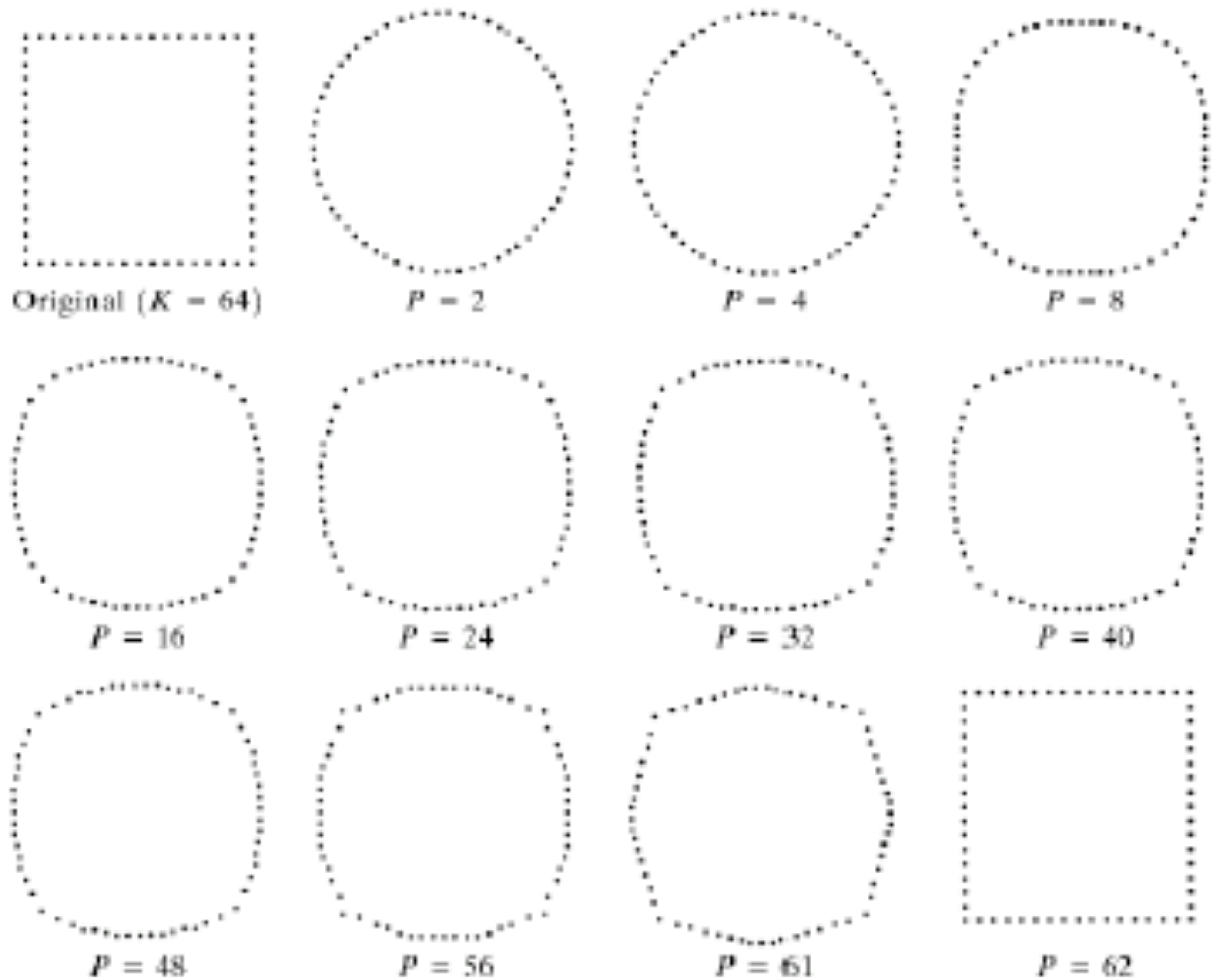
TABLE 11.1 Some basic properties of Fourier descriptors.

- Most of the time, just
  - use the magnitude (invariant to start point)
  - ignore the zero-frequency term (invariant to translation)
  - normalize sum (invariant to overall size)

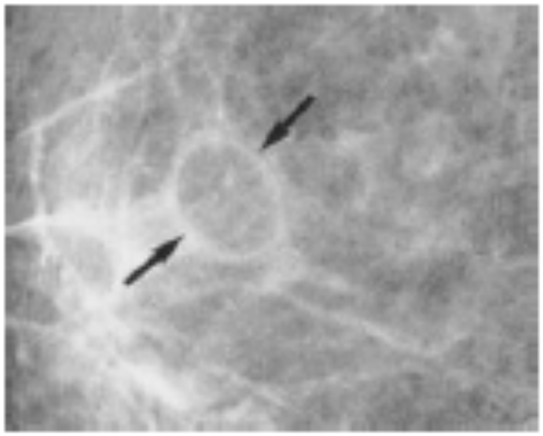
Why Fourier descriptors?

- Separates
  - low-frequency components of shape (general properties)
  - high-frequency ones (detail, small perturbations)
- Can filter shape!
  - Low-pass filtering the Fourier descriptor smooths the shape

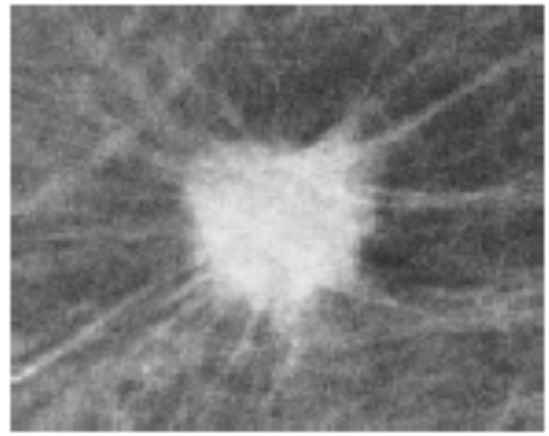
FIGURE 11.14 Examples of reconstruction from Fourier descriptors. P is the number of Fourier coefficients used in the reconstruction of the boundary.



Computer-aided diagnosis: Comparing two contours



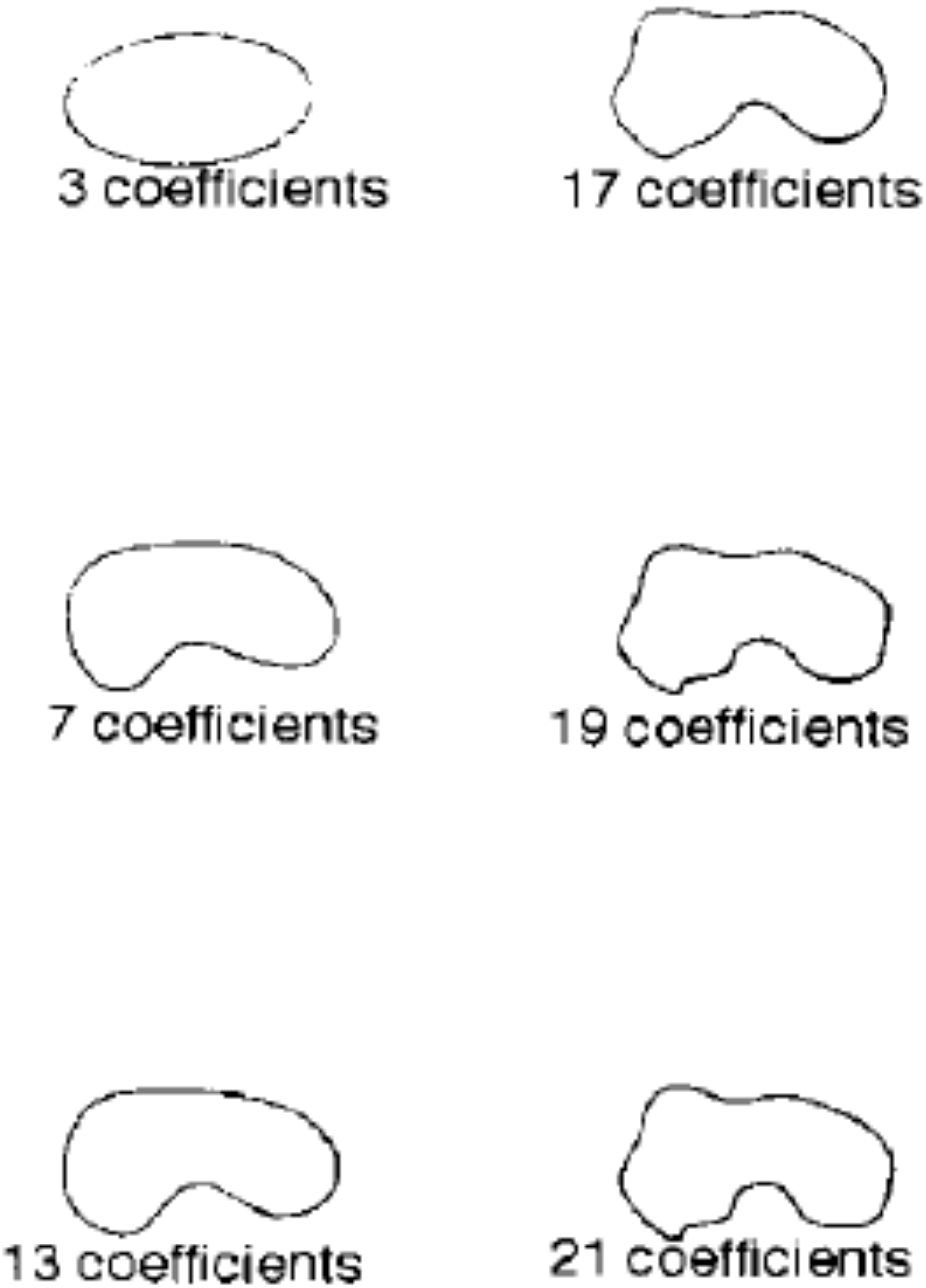
Circumscribed (benign) lesions in digital mammography



Spiculated lesions in (digital mammography)

The feature of interest: regularity of contour

-The circumscribed shape will have its Fourier coefficients at lower frequencies than the spiculated shape



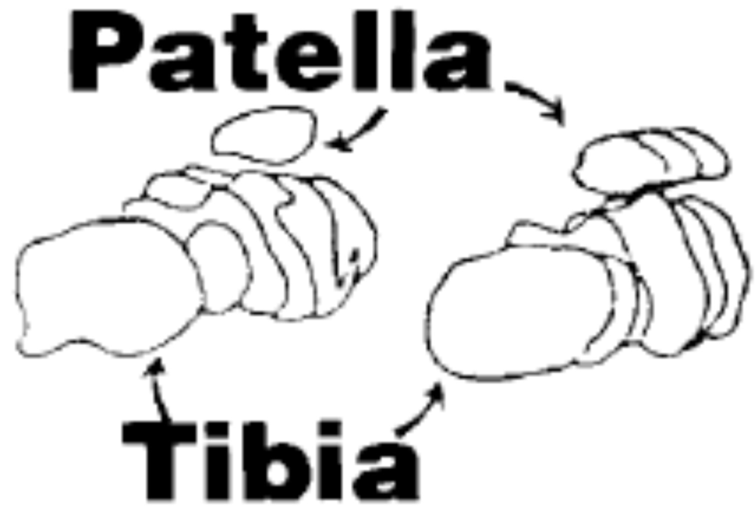


Fig. 6. Reconstruction of knee region anatomy with truncated Fourier series.

Fig. 8. Progressive approximations to a contour with increasing number of terms in the reconstruction.

Example of using contour descriptors

- Jeong and Radke, "Reslicing axially sampled 3D shapes using elliptic Fourier descriptors", Medical Image Analysis 2007

Main idea:

- Contour-based interpolation
- Interpolation between parallel slices from a 3D shape is necessary for reslicing and putting into correspondence organ shapes acquired from volumetric medical imagery

Rationale

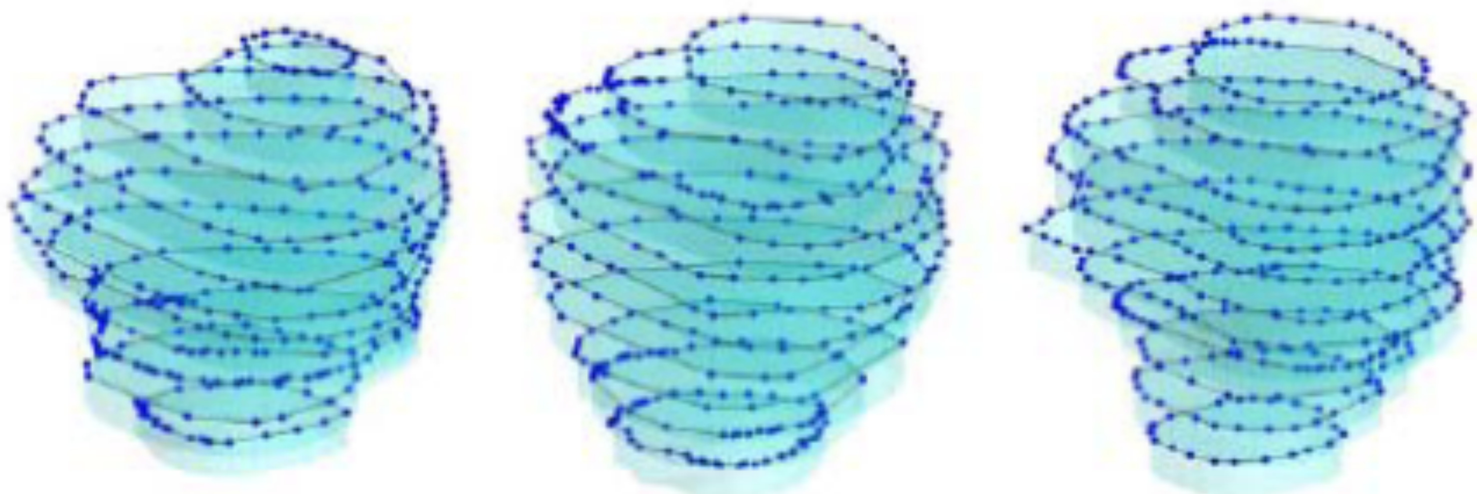


Fig. 1. Axial slices of one patient's prostate acquired on three different days of radiation treatment, contoured from CT imagery. The number of axial slices for each dataset and the number of sample points around each contour generally vary between datasets, and the spacing of the sample points around each contour is usually nonuniform.

Interpolation using Elliptic Fourier descriptors

- The main goal of the elliptical Fourier analysis is to approximate a closed contour as the sum of elliptical harmonics.

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} a_0 \\ c_0 \end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \begin{bmatrix} \cos kt \\ \sin kt \end{bmatrix}$$

$$a_0 = \frac{1}{2\pi} \sum_{j=0}^{R-1} x(t_j), \qquad c_0 = \frac{1}{2\pi} \sum_{j=0}^{R-1} y(t_j),$$

$$a_k = \frac{1}{\pi} \sum_{j=0}^{R-1} x(t_j) \cos kt_j, \quad b_k = \frac{1}{\pi} \sum_{j=0}^{R-1} x(t_j) \sin kt_j,$$

$$c_k = \frac{1}{\pi} \sum_{j=0}^{R-1} y(t_j) \cos kt_j, \quad d_k = \frac{1}{\pi} \sum_{j=0}^{R-1} y(t_j) \sin kt_j.$$

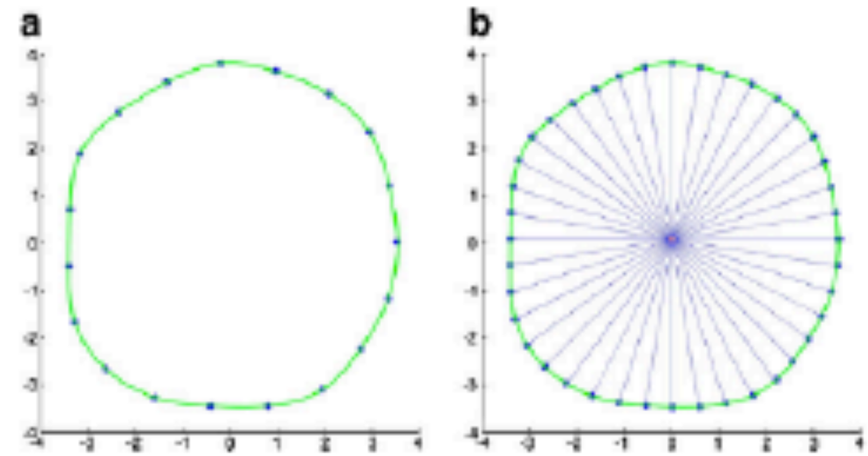


Fig. 3. (a) The original set of points on a slice. (b) The set of points after spline interpolation and resampling.

Contour Interpolation via EFD descriptors

- Input: 2 or more contour images
- Convert each contour to EFDs
- Assign a z-value to each contour (frame #, or distance between MRI slices)
- Choose z-value(s) at which to interpolate
- Use bicubic interpolation for EFDs at those values of z
- Convert interpolated EFDs back into x,y contour images