OBJECT DESCRIPTION – SHAPE FEATURE EXTRACTION

Fritz Albregtsen 18.09.2019
Today

From the textbook: DIP3E DIP4E

Boundary (Feature) Descriptors 11.2 (815-822) 11.3 (831-840)
Regional (Feature) Descriptors 11.3 (822-842) 11.4 (840-859)

Curriculum includes these lecture notes.

Today we cover the following:
1. Introduction
2. Topological features
3. Projections
4. Geometric features
5. Statistical shape features
6. Moment-based geometric features
7. Finding best feature subset
What is feature extraction?

• Devijver and Kittler (1982):

  "Extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class variability".

  – Within-class pattern variability: variance between objects belonging to the same class.

  – Between-class pattern variability: variance between objects from different classes.
Feature extraction

- We will discriminate between different object classes based on a set of features.
- The features are often chosen given the application.
- Normally, a large set of different features is investigated.
- Classifier design also involves feature selection - selecting the best subset out of a larger feature set.
- Given a training data set of a certain size, the dimensionality of the feature vector must be limited.

- Careful selection of an optimal set of features is the most important step in image classification!
Feature extraction methods

• There are a lot of different feature extraction methods, you will only learn some in this course.

• The focus of this lecture is on **features for describing the shape of an object**.

• Features can also be extracted in local windows around each pixel, e.g.
  – texture descriptors,
  – colour features,
  – or other methods.

• The features will later be used for object recognition and/or classification.
Example: Recognize printed numbers

- Goal: get the series of digits, e.g. 14159265358979323846......

Steps in the program:

1. Segment the image to find digit pixels.
2. Find angle of rotation and rotate back.
3. Create region objects – one object pr. digit or connected component.
4. **Compute features describing shape of objects**
5. Train a classifier on many objects of each digit.
6. Assign a class label to each new object, i.e., the class with the highest probability.

Focus of this lecture
Typical image analysis tasks

- Preprocessing/noise filtering
- Segmentation
- **Feature extraction**
  - Are the original image pixel values sufficient for classification, or do we need additional features?
  - What kind of features do we use in order to discriminate between the object classes involved?
- **Exploratory feature analysis and selection (next lecture)**
  - Which features separate the object classes best?
  - How many features are needed?
- **Classification (following three/four lectures)**
  - From a set of object examples with known class, decide on a method that separates objects of different types.
  - For new objects: assign each object/pixel to the class with the highest probability
- Testing and validation of classifier accuracy
Topologic features

- This is a group of invariant **integer** features
  - Invariant to position, rotation, scaling, warping

- Features based on the object skeleton
  - Number of terminations (one line from a point)
  - Number of breakpoints or corners (two lines from a point)
  - Number of branching points (three lines from a point)
  - Number of crossings (> three lines from a point)

- Region features:
  - Number of holes in the object (H)
  - Number of components (C)
  - Euler number, $E = C - H$
    - Number of connected components – number of holes
  - Symmetry

Region with two holes

Regions with three connected components
1D Projection histograms

- For each row in the region, count the number of object pixels.

![Image - binary region pixels](image1)

![Row histogram](image2)
Projections

• 1D horizontal projection of the region (project on the x-axis):
  \[ p_h(x) = \sum_y f(x, y) \]

• 1D vertical projection of the region (project on the y-axis):
  \[ p_v(y) = \sum_x f(x, y) \]

• \( f(x,y) \) is normally the binary segmented image
• Can be made scale independent by using a fixed number of bins and normalizing the histograms.

• Radial projection in reference to centroid -> “signature”, see previous lecture.
Use of projection histograms

• Divide the object into different regions and compute projection histograms for each region.
  – How can we use this to separate 6 and 9?

• Compute features from the histograms.
  – E.g. mean and variance of the histograms.

• The histograms can also be used as features directly.

• Projections are also useful for preprocessing (e.g., finding and correcting angle of rotation).
Use of projection histograms

- Check if a page with text is rotated

- Detecting lines, connected objects, single symbols ...

F06 18.09.2019
IN5520
Object area

• Generally, the area is defined as:
  \[ A = \int_{x}^{y} \int_{y}^{x} I(x, y) \, dx \, dy \]
  \[ I(x,y) = 1 \text{ if the pixel is within the object, and 0 otherwise.} \]
  \[ \Delta A = \text{area of one pixel. If } \Delta A = 1, \text{ area is simply measured in pixels.} \]

• In digital images:
  \[ A = \sum_{X} \sum_{Y} I(x, y) \, \Delta A \]

• Area changes if we change the scale of the image
  – change is not perfectly linear, because of the discretization of the image.

• Area \( \approx \) invariant to rotation (except small discretization errors).
Geometric features from contours

- Boundary length/perimeter
- Area
- Curvature
- Diameter/major/minor axis
- Eccentricity
- Bending energy
- Basis expansion (Fourier – last week)
Perimeter length from chain code

- Distance measure differs when using 8- or 4-neighborhood
- Using 4-neighborhood, measured length \( \geq \) actual length.
- In 8-neighborhood, fair approximation from chain code by:
  \[
P_F = n_E + n_O \sqrt{2}
\]
  where \( n_E \) and \( n_O \) are the number of even / odd chain elements.
- This overestimates real perimeters systematically.
- Freeman (1970) computed the area and perimeter of the chain by
  \[
  A_F = \sum_{i=1}^{N} c_{ix} \left( y_{i-1} + \frac{c_{iy}}{2} \right), \quad P_F = n_E + n_O \sqrt{2}
  \]
  where \( N \) is the length of the chain, \( c_{ix} \) and \( c_{iy} \) are the \( x \) and \( y \) components of the \( i \)th chain element \( c_i \) (\( c_{ix}, c_{iy} = \{1, 0, -1\} \) indicate the change of the \( x \)- and \( y \)-coordinates), \( y_{i-1} \) is the \( y \)-coordinate of the start point of the chain element \( c_i \).
Perimeter from chain code

- Vossepoel and Smeulders (1982) improved perimeter length estimate by a corner count $n_C$, defined as the number of occurrences of unequal consecutive chain elements:

$$P_{VS} = 0.980n_E + 1.406n_O - 0.091n_C$$

- Kulpa (1977) gave the perimeter as

$$P_K = \frac{\pi}{8} \left( 1 + \sqrt{2} \right) \left( n_E + \sqrt{2}n_O \right)$$
Pattern matching - bit quads

- Let \( n(Q) \) = number of matches between image pixels and pattern \( Q \).
- Then area and perimeter of 4-connected object is given by:
  \[
  A = n\{1\}, \quad P = 2n\{0 \quad 1\} + 2n\{0 \quad 1\}
  \]

“Bit Quads” can handle 8-connected images:

- Gray (1971) gave area and the perimeter as
  \[
  A_G = \frac{1}{4} \left[ n(Q_1) + 2n(Q_2) + 3n(Q_3) + 4n(Q_4) + 2n(Q_D) \right], \quad P_G = n(Q_1) + n(Q_2) + n(Q_3) + 2n(Q_D)
  \]

- More accurate formulas by Duda:
  \[
  A_D = \frac{1}{4} \left[ n(Q_1) + 2n(Q_2) + \frac{7}{2} n(Q_3) + 4n(Q_4) + 3n(Q_D) \right], \quad P_D = n(Q_2) + \frac{1}{\sqrt{2}} \left[ n(Q_2) + n(Q_3) + 2n(Q_D) \right]
  \]
A comparison of methods

- We have tested the methods on circles, $R = \{5, \ldots, 70\}$.

- Area estimator:
  - Duda is slightly better than Gray.

- Perimeter estimator:
  - Kulpa is more accurate than Freeman.

- Circularity:
  - Kulpa’s perimeter and Gray’s area gave the best result.

- Errors and variability largest when $R$ is small.
  - Test this yourself on this image:

- Best area and perimeter not computed simultaneously.

- Gray’s area can be computed using discrete Green’s theorem, suggesting that the two estimators can be computed simultaneously during contour following.
Object area from contour

- The surface integral over $S$ (having contour $C$) is given by Green’s theorem:

$$s := 0.0;$$
$$n := n + 1;$$
$$pkt[n].x := pkt[1].x;$$
$$pkt[n].y := pkt[1].y;$$
for $i:=2$ step 1 until $n$ do
begin
  $$dy := pkt[i].y - pkt[i-1].y$$
  $$s := s + (pkt[i].x + pkt[i-1].x)/2 * dy;$$
end;
area := if $(s > 0)$ then $s$ else $-s$;

- The region can also be represented by $n$ polygon vertices (see previous lecture)

$$\hat{A} = \frac{1}{2} \sum_{k=0}^{N-1} \left( x_k y_{k+1} - x_{k+1} y_k \right)$$

where the sign of the sum reflects the polygon orientation.
Compactness and circularity

• Compactness (very simple measure)
  – \( \gamma = \frac{P^2}{4\pi A} \), where \( P \) = Perimeter, \( A \) = Area,
  – For a circular disc, \( \gamma \) is minimum and equals 1.
  – Compactness attains high value for complex object shapes,
    but also for very elongated simple objects,
    like rectangles and ellipses where \( a/b \) ratio is high.

\[\Rightarrow \text{Compactness is not correlated with complexity!}\]

• G&W defines
  – Compactness = \( \frac{P^2}{A} \)
  – Circularity ratio = \( \frac{4\pi A}{P^2} \)

\( Y=3.4 \)
\( \text{Circ}=0.28 \)

\( Y=10.1 \)
\( \text{Circ}=0.09 \)
Circularity and irregularity

- **Circularity** may be defined by \( C = \frac{4\pi A}{P^2} \).
- \( C = 1 \) for a perfect continuous circle; betw. 0 and 1 for other shapes.
- In digital domain, \( C \) takes its smallest value for a
  - digital octagon in 8-connectivity perimeter calculation
  - digital diamond in 4-connectivity perimeter calculation
- **Dispersion** may be given as the major chord length to area
- **Irregularity** of an object of area \( A \) can be defined as:
  \[
  D = \frac{\pi \max\left((x_i - \bar{x})^2 + (y_i - \bar{y})^2\right)}{A}
  \]
  - where the numerator is the area of the centered enclosing circle.
- Alternatively, ratio of maximum and minimum centered circles:
  \[
  I = \frac{\max\left(\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}\right)}{\min\left(\sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}\right)}
  \]
Curvature

- In the continuous case, curvature is the rate of change of slope.
  \[ |\kappa(s)|^2 = \left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2 \]
- In the discrete case, difficult because boundary is locally ragged.
- Use difference between slopes of adjacent boundary segments
to describe curvature at point of segment intersection.

Curvature can be calculated from chain code.

*How to get from chain code to curvature, a simple example:*
0, 0, 2, 0, 1, 0, 7, 6, 0, 0 \rightarrow 0, 2, -2, 1, -1, -1, -1, 2, 0

*then square:*
0, 4, 4, 1, 1, 1, 1, 4, 0
Discrete computation of curvature

- Trace the boundary and insert vertices, at a given distance (e.g. 3 pixels apart), or by polygonization (previous lecture).

- Compute local curvature $c_i$ as the difference between the directions of two edge segments joining a vertex:
  \[ c_i = \vec{d}_i - \vec{d}_{i-1} \]

- Curvature feature: sum all local curvature measures along the border.

- More complex regions get higher curvature.
Contour based features

• Diameter = Major axis (a)
  Longest distance of a line segment connecting two points on the perimeter

• Minor axis (b)
  Computed along a direction perpendicular to the major axis. Largest length possible between two border points in the given direction.

• So called “Eccentricity” of the contour (a/b)

Eccentricity: 4.7
Eccentricity: 1.8
Bounding box and CH features

- Regular bounding box
  - Width/height of bounding box
  - Centre of mass position in box

- If the object’s orientation is known, a bounding box can also be oriented along this direction.

- Extent = Area/(Area of bounding box)
  - But which type of bounding box?
- Solidity = Area/(Area of Convex Hull) (also termed “convexity”)
Moments

- Borrows ideas from physics and statistics.
- For a given continuous intensity distribution \( g(x, y) \)
  we define moments \( m_{pq} \) by

\[
m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q g(x, y) \, dx \, dy
\]

- For sampled (and bounded) intensity distributions \( f(x, y) \)
  over a region \( R \)

\[
m_{pq} = \sum_{x} \sum_{y} x^p y^q f(x, y)
\]

- A moment \( m_{pq} \) is said to be of order \( p + q \).
Moments from binary images - area

- For binary images, where
  \[ f(x, y) = 1 \Rightarrow \text{object pixel} \]
  \[ f(x, y) = 0 \Rightarrow \text{background pixel} \]

- Area

\[ m_{00} = \sum_{x} \sum_{y} f(x, y) \]
Centroid/center of mass from moments

- For binary images, where
  \[ f(x, y) = 1 \Rightarrow \text{object pixel} \]
  \[ f(x, y) = 0 \Rightarrow \text{background pixel} \]

- Center of mass /”tyngdepunkt”

\[
m_{10} = \sum_x \sum_y x f(x, y) = \bar{x} m_{00} \quad \Rightarrow \quad \bar{x} = \frac{m_{10}}{m_{00}}
\]

\[
m_{01} = \sum_x \sum_y y f(x, y) = \bar{y} m_{00} \quad \Rightarrow \quad \bar{y} = \frac{m_{01}}{m_{00}}
\]

gives the position of the object
Grayscale moments

- In gray scale images, we may regard $f(x,y)$ as a discrete 2-D probability distribution over $(x,y)$.

- For probability distributions, we should have

\[ m_{00} = \sum_x \sum_y f(x,y) = 1 \]

  - And if this is not the case we can normalize by

\[ F(x,y) = f(x,y)/m_{00} \]
Example use of centroid of grayscale image

Ultrasound image of muscle fibers
Want to find the near-horizontal lines using Radon-transform
(generalized Hough to grayscale images)

Find peaks in the Radon domain
How do we robustly find the location of a «peak» in the marked areas?
Central moments

- These are position invariant moments, defined by

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

- where

\[ \bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{01}}{m_{00}} \]

- The total mass, and the center of mass coordinates are given by

\[ \mu_{00} = \sum_x \sum_y f(x, y), \quad \mu_{10} = \mu_{01} = 0 \]

- This corresponds to computing ordinary moments after having translated the object so that center of mass is in origo.

- Central moments are independent of position, but are not scaling or rotation invariant.

- **Q:** What is \( \mu_{00} \) for a binary object?
A simple example: center of mass

\[ m_{00} = \sum_x \sum_y f(x, y) = 6 \]

\[ m_{10} = \sum_x \sum_y x f(x, y) = \bar{x} m_{00} \quad \Rightarrow \quad \bar{x} = \frac{m_{10}}{m_{00}} = 2 \]

\[ m_{01} = \sum_x \sum_y y f(x, y) = \bar{y} m_{00} \quad \Rightarrow \quad \bar{y} = \frac{m_{01}}{m_{00}} = 2 \]
Central moments $\mu_{pq}$ from ordinary moments $m_{pq}$

- Moments $\mu_{pq}$ ($p + q \leq 3$) are given by $m_{pq}$ by:

$$
\begin{align*}
\mu_{00} &= m_{00}, \quad \mu_{10} = 0, \quad \mu_{01} = 0 \\
\mu_{20} &= m_{20} - \bar{x}m_{10} \\
\mu_{02} &= m_{02} - \bar{y}m_{01} \\
\mu_{11} &= m_{11} - \bar{y}m_{10} \\
\mu_{30} &= m_{30} - 3\bar{x}m_{20} + 2\bar{x}^2m_{10} \\
\mu_{12} &= m_{12} - 2\bar{y}m_{11} - \bar{x}m_{02} + 2\bar{y}^2m_{10} \\
\mu_{21} &= m_{21} - 2\bar{x}m_{11} - \bar{y}m_{20} + 2\bar{x}^2m_{01} \\
\mu_{03} &= m_{03} - 3\bar{x}m_{02} + 2\bar{y}^2m_{01}
\end{align*}
$$
Generalization to 3D moments from $m_{pq}$

- The 3D $\mu_{pqr}$ are expressed by $m_{pqr}$:

$$\mu_{pqr} = \sum_{s=0}^{p} \sum_{t=0}^{q} \sum_{u=0}^{r} -1^{[D-d]} \binom{p}{s} \binom{q}{t} \binom{r}{u} \Delta x^{p-s} \Delta y^{q-t} \Delta z^{r-u} m_{stu}$$

- where

  \[ D = (p + q + r); \ d = (s + t + u) \]

- and the binomial coefficients are given by

$$\binom{v}{w} = \frac{v!}{w! \ (v-w)!}, \ w < v$$
Moments of inertia or Variance

- The two second order central moments measure the spread of points around the y- and x-axis through the centre of mass

\[
\mu_{20} = \sum_x \sum_y (x - \bar{x})^2 f(x, y)
\]
\[
\mu_{02} = \sum_x \sum_y (y - \bar{y})^2 f(x, y)
\]

- From physics: moment of inertia about an axis: how much energy is required to rotate the object about this axis:
  - Statisticians like to call this variance.

- The cross moment of inertia is given by

\[
\mu_{11} = \sum_x \sum_y (x - \bar{x})(y - \bar{y}) f(x, y)
\]
  - Statisticians call this covariance or correlation.

- Orientation of the object can be derived from these moments.
  - This implies that they are not invariant to rotation.
A simple example

\[
\mu_{20} = \sum_x \sum_y (x - \bar{x})^2 f(x, y) = 5
\]

\[
\mu_{02} = \sum_x \sum_y (y - \bar{y})^2 f(x, y) = 5
\]

\[
\mu_{11} = \sum_x \sum_y (x - \bar{x})(y - \bar{y}) f(x, y) = 5
\]

Note that image coordinates are swapped
Object orientation - I

• Orientation is defined as the angle, relative to the X-axis, of an axis through the centre of mass that gives the lowest moment of inertia.

• Orientation $\theta$ relative to X-axis found by minimizing:

$$I(\theta) = \sum_\alpha \sum_\beta \beta^2 f(\alpha, \beta)$$

where the rotated coordinates are given by

$$\alpha = x \cos \theta + y \sin \theta, \quad \beta = -x \sin \theta + y \cos \theta$$

• The second order central moment of the object around the $\alpha$-axis, expressed in terms of $x$, $y$, and the orientation angle $\theta$ of the object is:

$$I(\theta) = \sum_\alpha \sum_\beta \left[ y \cos \theta - x \sin \theta \right]^2 f(x, y)$$

• We take the derivative of this expression with respect to the angle $\theta$

• Set derivative equal to zero, and find a simple expression for $\theta$:
Object orientation - II

- Second order central moment around the $\alpha$-axis:

$$I(\theta) = \sum_x \sum_y [y \cos \theta - x \sin \theta]^2 f(x, y)$$

- Derivative w.r.t. $\Theta = 0 \Rightarrow$

$$\frac{\partial}{\partial \theta} I(\theta) = \sum_x \sum_y 2 f(x, y) [y \cos \theta - x \sin \theta] [-y \sin \theta - x \cos \theta] = 0$$

$$\sum_x \sum_y 2 f(x, y) [xy (\cos^2 \theta - \sin^2 \theta)] = \sum_x \sum_y 2 f(x, y) [x^2 - y^2] \sin \theta \cos \theta$$

$$2\mu_{11} (\cos^2 \theta - \sin^2 \theta) = 2(\mu_{20} - \mu_{02}) \sin \theta \cos \theta$$

$$\frac{2\mu_{11}}{(\mu_{20} - \mu_{02})} = \frac{2 \sin \theta \cos \theta}{(\cos^2 \theta - \sin^2 \theta)} = \frac{2 \tan \theta}{1 - \tan^2 \theta} = \tan(2\theta)$$

- So the object orientation is given by:

$$\theta = \frac{1}{2} \tan^{-1} \left[ \frac{2\mu_{11}}{(\mu_{20} - \mu_{02})} \right], \quad \text{where} \quad \theta \in \left[0, \frac{\pi}{2}\right] \text{if } \mu_{11} > 0, \quad \theta \in \left[\frac{\pi}{2}, \pi\right] \text{if } \mu_{11} < 0$$
A simple example

\[
\mu_{20} = \sum_x \sum_y (x - \bar{x})^2 f(x, y) = 5
\]

\[
\mu_{02} = \sum_x \sum_y (y - \bar{y})^2 f(x, y) = 5
\]

\[
\mu_{11} = \sum_x \sum_y (x - \bar{x})(y - \bar{y}) f(x, y) = -5
\]

\[
\theta = \frac{1}{2} \tan^{-1} \left[ \frac{2 \mu_{11}}{(\mu_{20} - \mu_{02})} \right], \quad \text{where } \theta \in [0, \pi/2] \text{ if } \mu_{11} > 0, \quad \theta \in [\pi/2, \pi] \text{ if } \mu_{11} < 0
\]

Note that image coordinates are swapped

Image coordinate

\[
\begin{bmatrix}
4,1 \\
3,2 \\
2,3 \\
1,4
\end{bmatrix}
\]

\[= -45\text{degrees}\]
Bounding box - again

- **Image-oriented bounding box:**
  - The smallest rectangle around the object, having sides parallel to the edges of the image.
  - Found by searching for min and max x and y within the object \((x_{min}, y_{min}, x_{max}, y_{max})\)

- **Object-oriented bounding box:**
  - Smallest rectangle around the object, having one side parallel to the orientation of the object \((\theta)\).
  - The transformation
    \[
    \alpha = x \cos \theta + y \sin \theta, \quad \beta = y \cos \theta - x \sin \theta
    \]
  - is applied to all pixels in the object (or its boundary).
  - Then search for \(\alpha_{min}, \beta_{min}, \alpha_{max}, \beta_{max}\)
The best fitting ellipse

- Object ellipse is defined as the ellipse whose least and greatest moments of inertia equal those of the object.
- Semi-major and semi-minor axes are given by
  \[
  \left( \hat{a}, \hat{b} \right) = \sqrt{\frac{2}{\mu_{00}}} \left[ \mu_{20} + \mu_{02} \pm \sqrt{(\mu_{20} + \mu_{02})^2 + 4 \mu_{11}^2} \right]
  \]
- Numerical eccentricity is given by
  \[
  \hat{e} = \sqrt{\frac{\hat{a}^2 - \hat{b}^2}{\hat{a}^2}}
  \]
- Orientation invariant object features.
- Gray scale or binary object.
Radius of gyration, K

- The radius of a circle where we could concentrate all the mass of an object without altering the moment of inertia about its center of mass.

- For arbitrary object having a mass $\mu_{00}$ and a moment of inertia around the Z-axis, we may write

$$ I = \mu_{00} \hat{K}^2 \Rightarrow \hat{K} = \sqrt{\frac{I_Z}{\mu_{00}}} = \sqrt{\frac{I_X + I_Y}{\mu_{00}}} = \sqrt{\frac{\mu_{20} + \mu_{02}}{\mu_{00}}} $$

- This feature is obviously invariant to rotation.
Radius of gyration, $K$

- For homogeneous objects, only determined by geometry.
- Thus, the squared radius of gyration may be tabulated for simple object shapes:

  - **Rectangle:** $K^2 = \frac{b^2}{3}$
    
    ![Rectangle Diagram](image)
    
    \[ K^2 = \frac{a^2+b^2}{3} \]

  - **Circular disk:** $K^2 = \frac{R^2}{4}$
    
    ![Circular Disk Diagram](image)
    
    \[ K^2 = \frac{R^2}{2} \]

  - **Ellipse:** $K^2 = \frac{b^2}{4}$
    
    ![Ellipse Diagram](image)
    
    \[ K^2 = \frac{(a^2+b^2)}{4} \]
What if we want scale-invariance?

- Changing the scale of $f(x,y)$ by $(\alpha,\beta)$ gives a new image:
  $$f'(x,y) = f(x/\alpha, y/\beta)$$

- The transformed central moments
  $$\mu'_{pq} = \alpha^{1+p} \beta^{1+q} \mu_{pq}$$

- If $\alpha=\beta$, scale-invariant central moments are given by the normalization:
  $$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^\gamma}, \quad \gamma = \frac{p+q}{2} + 1, \quad p + q \geq 2$$
Symmetry

- To detect symmetry about center of mass, use central moments.

- For invariance of scale, use scale-normalised central moments 
  \( (\eta_{11}, \eta_{20}, \eta_{02}, \eta_{21}, \eta_{12}, \eta_{30}, \eta_{03}) \).

- Objects symmetric about either x or y axis will produce \( \eta_{11} = 0 \).

- Objects symmetric about y axis will give \( \eta_{12} = 0 \) and \( \eta_{30} = 0 \).

- Objects symmetric about x axis will give \( \eta_{21} = 0 \) and \( \eta_{03} = 0 \).

- X axis symmetry: \( \eta_{pq} = 0 \) for all \( p = 0, 2, 4, \ldots \); \( q = 1, 3, 5, \ldots \)

<table>
<thead>
<tr>
<th></th>
<th>( \eta_{11} )</th>
<th>( \eta_{20} )</th>
<th>( \eta_{02} )</th>
<th>( \eta_{21} )</th>
<th>( \eta_{12} )</th>
<th>( \eta_{30} )</th>
<th>( \eta_{03} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘M’</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>‘C’</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>0</td>
</tr>
<tr>
<td>‘O’</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Rotation invariant moments

Method 1:

Find principal axes of object, rotate and compute moments.

This can break down if object has no unique principal axes.
Rotation invariant moments

Method 2 : Hu moments

The method of absolute moment invariants:

This is a set of normalized central moment combinations, which can be used for scale, position, and rotation invariant pattern identification.

• For second order (p+q=2), there are two invariants/Hu moments:

\[
\varphi_1 = \eta_{20} + \eta_{02} \quad \varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2
\]
Third order Hu moments

- For third order moments, \((p+q=3)\), the invariants are:

\[
\varphi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\varphi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
\varphi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
\varphi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
\varphi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
\quad - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\]

\(\varphi_7\) is skew invariant, and may help distinguish between mirror images.

- These moments are not independent, and do not comprise a complete set.
Hu’s moments; a bit simplified notation

For second order moments \((p+q=2)\), two invariants are used:

\[
\begin{align*}
\varphi_1 &= \eta_{20} + \eta_{02} \\
\varphi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2
\end{align*}
\]

For third order moments, \((p+q=3)\), we can use

\[
\begin{align*}
a &= (\eta_{30} - 3\eta_{12}), & b &= (3\eta_{21} - \eta_{03}), \\
c &= (\eta_{30} + \eta_{12}), & d &= (\eta_{21} + \eta_{03})
\end{align*}
\]

and simplify the five last invariants of the set:

\[
\begin{align*}
\varphi_3 &= a^2 + b^2 \\
\varphi_4 &= c^2 + d^2 \\
\varphi_5 &= ac[c^2 - 3d^2] + bd[3c^2 - d^2] \\
\varphi_6 &= (\eta_{20} - \eta_{02})[c^2 - d^2] + 4\eta_{11}cd \\
\varphi_7 &= bc[c^2 - 3d^2] - ad[3c^2 - d^2]
\end{align*}
\]
Hu moments of simple objects

- In the continuous case, the two first Hu moments of a binary rectangular object of size 2a by 2b, are given by

\[ \phi_1 = \frac{1}{12} \left( \frac{a+b}{b} + a \right), \quad \phi_2 = \left( \frac{1}{12} \right)^2 \left( \frac{a-b}{b} \right)^2 \]

while the remaining five Hu moments are all zero.

- Similarly, the two first Hu moments of a binary elliptic object with semi-axes a and b, are given by

\[ \phi_1 = \frac{1}{4\pi} \left( \frac{a+b}{b} + a \right), \quad \phi_2 = \left( \frac{1}{4\pi} \right)^2 \left( \frac{a-b}{b} \right)^2 \]

while the remaining five Hu moments are all zero.
\( \Phi_1 \) and \( \varphi_2 \) versus \( a/b \)

- Only \((\varphi_1, \varphi_2)\) are useful for these simple objects.

- Notice that even in the continuous case it may be hard to distinguish between an ellipse and its bounding rectangle using these two moments.

- Relative difference in \( \varphi_1 \) of ellipse and its object oriented bounding rectangle is constant, 4.5%.

- Relative difference in \( \varphi_2 \) of ellipse and its object oriented bounding rectangle is constant, 8.8%.

- Relative differences given above are also true when comparing an ellipse with a same-area rectangle having the same \( a/b \) ratio, regardless of the size and eccentricity of the ellipse.
Moments as shape features

- The central moments are seldom used directly as shape descriptors.

- Major and minor axis, radius of gyration, and eccentricity are useful shape descriptors.

- Object orientation is normally not used directly, but to estimate rotation.

- The set of 7 Hu moments can be used as shape features. (Start with the first four, as the last half are often zero for simple objects).
Moments of inertia for simple shapes

- Rectangular object (2a×2b):
  \[ I_{20} = \frac{4a^3b}{3} , \quad I_{02} = \frac{4ab^3}{3} \]

- Square (a×a):
  \[ I_{20} = I_{02} = \frac{a^4}{12} \]

- Elliptical object, semi-axes (a,b):
  \[ I_{20} = \frac{\pi a^3b}{4} , \quad I_{02} = \frac{\pi ab^3}{4} \]

- Circular object, radius R:
  \[ I_{20} = I_{02} = \frac{\pi R^4}{4} \]
Moments of an ellipse

- Assume that the ellipse has semimajor and semiminor axes \((a, b)\), \(a > b\).

An ellipse where major axis is along \(x\)-axis is given by

\[
\frac{(x/a)^2}{1} + \frac{(y/b)^2}{1} = 1 \implies y = \pm \frac{b}{a} \sqrt{a^2 - x^2}
\]

The largest second order central moment (here called \(I_{20}\)) is given by

\[
I_{20} = 2 \int_{-a}^{a} x^2 y \, dx = 2 \frac{b}{a} \int_{-a}^{a} x^2 \sqrt{a^2 - x^2} \, dx
\]

\[
I_{20} = 2 \frac{b}{a} \left[ \frac{x}{8} \left(2x^2 - a^2\right) \sqrt{a^2 - x^2} + \frac{a^4}{8} \sin^{-1}\left(\frac{x}{a}\right) \right]_{-a}^{a}
\]

\[
I_{20} = 2 \frac{b}{a} \left[ \frac{a^4}{8} \left(\frac{\pi}{2} + \frac{\pi}{2}\right) \right] = \frac{\pi}{4} a^3 b
\]

Similarly, the smallest moment of inertia is

\[
I_{\text{min}} = \frac{\pi}{4} a b^3
\]
Grayscale contrast invariants

- Abo-Zaid *et al.* have defined a normalization that cancels both scaling and contrast.
- The normalization is given by
  \[ \eta'_{pq} = \frac{\mu_{pq}}{\mu_{00}} \left( \frac{\mu_{00}}{\mu_{20} + \mu_{02}} \right)^{\frac{(p+q)}{2}} \]
- This normalization also reduces the dynamic range of the moment features, so that we may use higher order moments without having to resort to logarithmic representation.
- Abo-Zaid’s normalization cancels the effect of changes in contrast, but not the effect of changes in intensity:
  \[ f'(x, y) = f(x, y) + b \]
- In practice, we often experience a combination:
  \[ f'(x, y) = cf(x, y) + b \]
From features to discrimination between objects

- The following slides introduces simple tools like scatter plots to visualize how good a feature (or combination of 2-3 features) is in separating objects of different types/classes.

- To evaluate features, we use training data consisting of objects with KNOWN CLASS.
Scatter plots

- A 2D scatter plot is a plot of feature values for two different features. Each object’s feature values are plotted in the position given by the features values, and with a class label telling its object class.

- Matlab: gscatter(feature1, feature2, labelvector)

- Classification is done based on more than two features, but this is difficult to visualize.

- Features with good class separation show clusters for each class, and different clusters should ideally be separated.
The “curse-of-dimensionality”

• Also called “peaking phenomenon”.
• For a finite training sample size, the correct classification rate initially increases when adding new features, attains a maximum and then begins to decrease.
• The implication is that:
• For a high measurement complexity, we will need large amounts of training data in order to attain the best classification performance.
• => 5-10 samples per feature per class.

Illustration from G.F. Hughes (1968).
Finding best feature subset

• The goal: to find the subset of observed features which
  – best characterizes the differences between groups
  – is similar within the groups
  – **Maximize the ratio of between-class and within-class variance.**

• If we want to perform an exhaustive search through \( D \) features for the optimal subset of the \( d \leq m \) “best features”, the number of combinations to test is

\[
n = \sum_{d=1}^{D} \frac{D!}{(D-d)! \cdot d!}
\]

• Impractical even for a moderate number of features!

\( d \leq 5, \ D = 100 \ => \ n = 79.374.995 \)
A simulation study design

- Monte Carlo study, averaging 100 simulations per setting
- 2 classes, normally distributed, common covariance
- 10 to 500 feature candidates
- Only 5 features are different between the classes
  
  For these 5, squared difference of class means \( \frac{\delta^2}{\sqrt{5}} \); \( \delta^2 = 0, 1, 4 \)

  the rest of the continuous distributions are EQUAL!

- Stepwise forward-backward feature selection
- 20 - 1000 training samples
- 20 - 1000 test samples
Probabilistic distance measures

• If the class-conditional probability distributions are Gaussian:

\[ p(\xi \mid \omega_i) = \left[(2\pi)^d |\Sigma_i|\right]^{-1/2} \exp \left\{ -\frac{1}{2} (\xi - \mu_i)^T \Sigma_i^{-1} (\xi - \mu_i) \right\} \]

where \( \mu_i \) and \( \Sigma_i \) are the mean vector and the covariance matrix of the \( i \)-th class distribution; the Mahalanobis distance is

\[ \delta^2 = (\mu_2 - \mu_1)^T \Sigma^{-1} (\mu_2 - \mu_1), \text{ if } \Sigma_1 = \Sigma_2 = \Sigma \]

• The Bhattacharyya distance may also be useful:

\[ J_B = \frac{1}{4} (\mu_2 - \mu_1)^T [\Sigma_1 - \Sigma_2]^{-1} (\mu_2 - \mu_1) + \frac{1}{2} \ln \left[ \frac{1}{2} (\Sigma_1 + \Sigma_2) \right] \]

\[ \frac{1}{\sqrt{|\Sigma_1||\Sigma_2|}} \]
Samples from distributions

Distribution of 2 independent sets of 20 samples from standardized normal distributions, $\delta^2 = 0$.

Distribution of 2 independent sets of 200 samples from standardized normal distributions, $\delta^2 = 0$.

- For small sample sets and small class distances, observations may indicate a separation of classes, while no real difference exists !!!
Simulation results - Feature selection II

- The number of correctly selected features increases with
  - increasing # of training samples
  - decreasing # of candidates
  - (increasing class distance)

- For small sample sizes the number of candidates features is of great importance:
  - For $D = 50$ and $\delta^2 = 1$, half of the 5 selected features will be noise if $n_{Tr} = 100$.
  - For $D = 50$, $\delta^2 = 1$, $n_{Tr} = 50$, 60% of the selected features will be noise!
  - So, Be Careful !!!

*The average number of correctly selected features, when the class Mahalanobis distance $\delta^2 = 1$. 
Learning goals – object description

• Invariant topological features
• Projections and signatures – use and limitations
• Geometric features
  – Area, perimeter and circularity/compactness
  – Bounding boxes

• Moments, binary and grayscale
  • Ordinary moments and central moments
  – Moments of objects, object orientation, and best fitting ellipse
  • Focus on first- and second-order moments.
  – Invariance may be important

• Inspection of feature scatter plots
• Select your feature set with great care!