Recognition

Computer Vision I

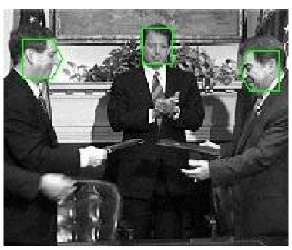
CSE 252A

Lecture 18

Announcements

– Read Chapters 15 & 16 of Forsyth & Ponce
– Homework 4 is due Dec 18, 11:59 PM
– Please complete evaluations
  - Course
  - TA

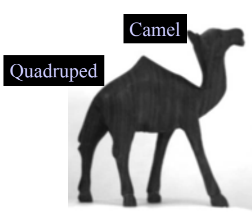
Recognition



Given a database of objects and an image determine what, if any of the objects are present in the image.

Recognition

Given a database of objects and an image determine what, if any of the objects are present in the image.

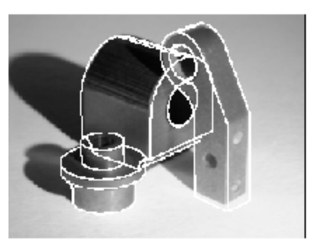


Problem:

Recognizing instances

Recognizing categories

Recognition

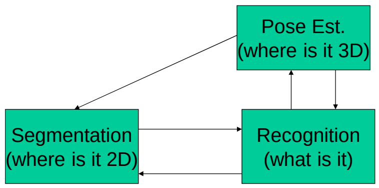


Given a database of objects and an image determine what, if any of the objects are present in the image.

Object Recognition: The Problem

Given: A database D of "known" objects and an image I:

1. Determine which (if any) objects in D appear in I
2. Determine the pose (rotation and translation) of the object



WHAT AND WHERE!!!

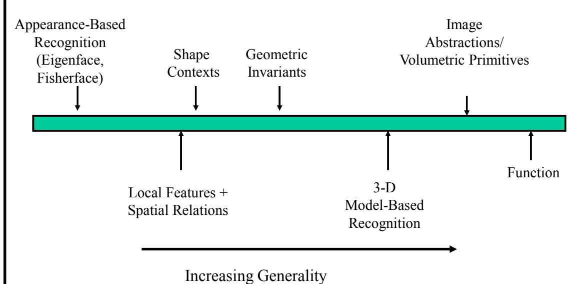
Recognition Challenges

- Within-class variability
  - Different objects within the class have different shapes or different material characteristics
  - Deformable
  - Articulated
  - Compositional
- Pose variability:
  - 2-D Image transformation (translation, rotation, scale)
  - 3-D Pose Variability (perspective, orthographic projection)
- Lighting
  - Direction (multiple sources & type)
  - Color
  - Shadows
- Occlusion – partial
- Clutter in background -> false positives

Object Recognition System Design Considerations

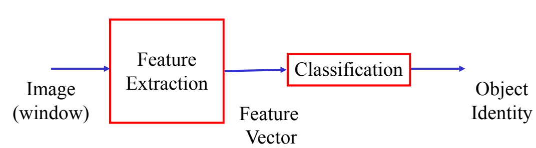
- How general is the problem?
  - 2D vs. 3D
  - range of viewing conditions
  - available context
  - segmentation cues
- What sort of data is best suited to the problem?
  - Whole images
  - Local 2D features (color, texture)
  - 3D (range) data
- What information do we have in the database?
  - Collection of images?
  - 3-D models?
  - Learned representation?
  - Learned classifiers?
- How many objects are involved?
  - small: brute force search
  - large: ??

A Rough Recognition Spectrum



Appearance-based Recognition

Sketch of a Pattern Recognition Architecture

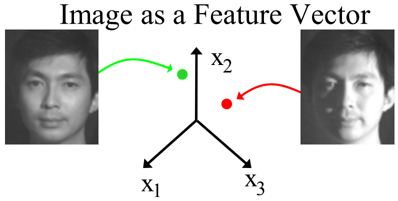


Example: Face Detection

- Scan window over image.
- Classify window as either:
  - Face
  - Non-face

- What are the features?
- What is the classifier?

Simplest feature: Image as a Feature Vector



- Consider an n-pixel image to be a point in an n-dimensional space, $\mathbf{x} \in \mathbf{R}^n$.
- Each pixel value is a coordinate of **x**.

More features

- Filtered image
- Filter with multiple filters (bank of filters
- Histogram of colors
- Histogram of Gradients (HOG)
- Haar wavelets
- Scale Invariant Feature Transform (SIFT)
- Speeded Up Robust Feature (SURF)

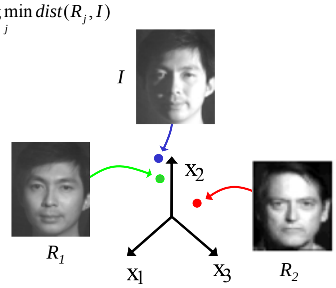
Pattern Classification Summary

- Supervised vs. Unsupervised: Do we have labels?
- Supervised
  - Nearest Neighbor
  - Bayesian
    - Plug in classifier
    - Distribution-based
    - Projection Methods (Fisher's, LDA)
  - Neural Network
  - Support Vector Machine
  - Kernel methods
- Unsupervised
  - Clustering
  - Reinforcement learning

Nearest Neighbor Classifier

$\{ R_j \}$ are set of training images.

$$ID = \arg\min_j dist(R_j, I)$$

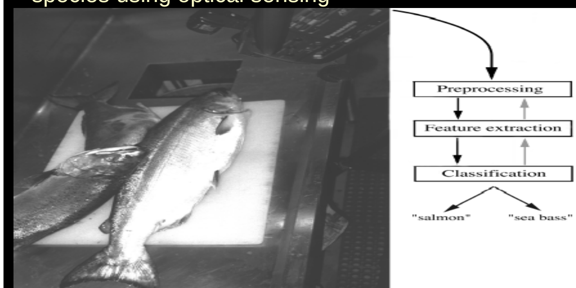


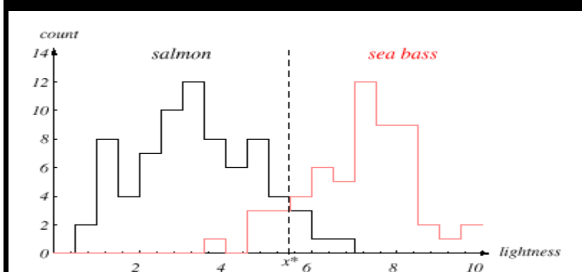
Comments

- Sometimes called "Template Matching"
- Variations on distance function (e.g. L_1, robust distances)
- Multiple templates per class- perhaps many training images per class.
- Expensive to compute k distances, especially when each image is big (N dimensional).
- May not generalize well to unseen examples of class.
- Some solutions:
  - Bayesian classification
  - Dimensionality reduction

An Example

- "Sorting incoming Fish on a conveyor according to species using optical sensing"

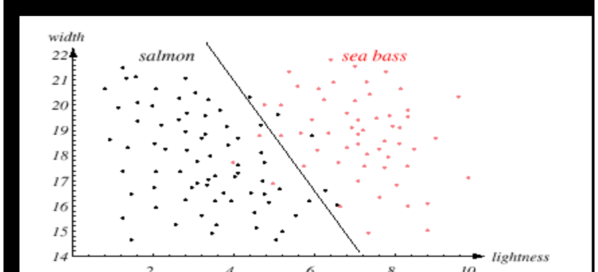


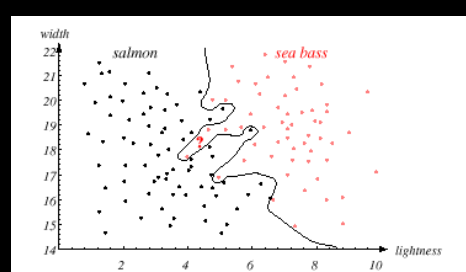


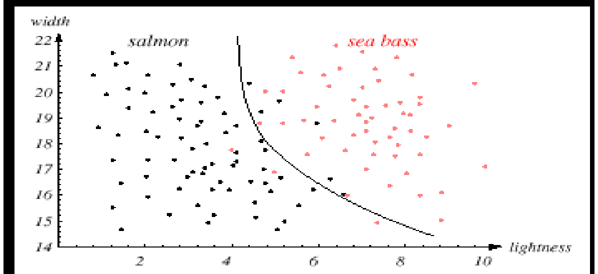
- Adopt the lightness and add the width of the fish

Fish $\rightarrow$ $x^T = [x_1, x_2]$

Lightness Width







Bayesian Decision Theory
Continuous Features

Introduction

- The sea bass/salmon example
  - State of nature is a random variable, ω_i
    - ω_1 – *the fish is a salmon*
    - ω_2 – *the fish is a sea bass*
  - *Prior Probabilities*
    - $P(\omega_1)$, $P(\omega_2)$
    - $P(\omega) > 0$
    - $P(\omega_1) + P(\omega_2) = 1$ (exclusivity and exhaustivity)
  - *Example prior: bass & salmon are equally likely*
    - $P(\omega_1) = P(\omega_2) = ½$ (uniform priors)

- Decision rule with only the prior information
  - Decide ω_1 if $P(\omega_1) > P(\omega_2)$ otherwise decide ω_2
- Use of the class–conditional information
- $P(x \mid \omega_1)$ and $P(x \mid \omega_2)$ describe the difference in lightness between populations of sea-bass and salmon

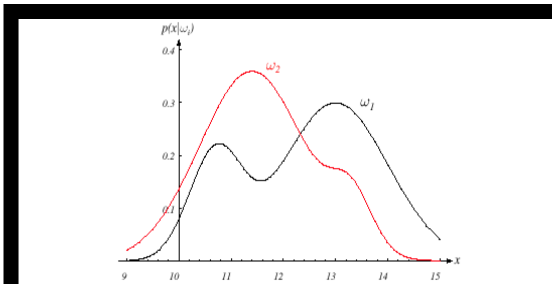


FIGURE 2.1. Hypothetical class-conditional probability density functions show the probability density of measuring a particular feature value x given the pattern is in category ω_i. If x represents the lightness of a fish, the two curves might describe the difference in lightness of populations of two types of fish. Density functions are normalized, and thus the area under each curve is 1.0. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

- Posterior, likelihood, evidence
  - $$P(\omega_j \mid x) = \frac{P(x \mid \omega_j)P(\omega_j)}{P(x)}$$ **(BAYES RULE)**
  - In words, this can be said as:
    Posterior = (Likelihood * Prior) / Evidence
  - Where in case of two categories

$$P(x) = \sum_{j=1}^{j=2} P(x \mid \omega_j)P(\omega_j)$$

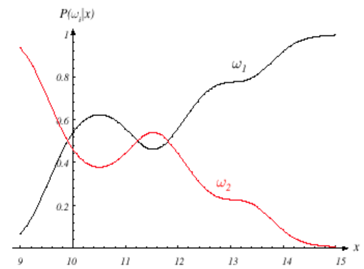
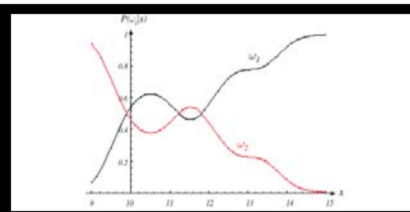


FIGURE 2.2. Posterior probabilities for the particular priors $P(\omega_1) = 2/3$ and $P(\omega_2) = 1/3$ for the class-conditional probability densities shown in Fig. 2.1. Thus in this case, given that a pattern is measured to have feature value $x = 14$, the probability it is in category ω_2 is roughly 0.08, and that it is in ω_1 is 0.92. At every x, the posteriors sum to 1.0. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



- Intuitive decision rule given the posterior probabilities:
  Given x:
  if $P(\omega_1 | x) > P(\omega_2 | x)$ ➡ True state of nature = ω_1
  if $P(\omega_1 | x) < P(\omega_2 | x)$ ➡ True state of nature = ω_2

  Why do this?: Whenever we observe a particular x, the probability of error is :
  $P(error | x) = P(\omega_1 | x)$ *if we decide* ω_2
  $P(error | x) = P(\omega_2 | x)$ *if we decide* ω_1

Plug-in classifiers

- Assume that class conditional distributions $P(x|\omega_i)$ have some parametric form - now estimate the parameters from the data.
- Common:
  - assume a normal distribution with shared covariance, different means; use usual estimates
  - Normal distribution but with different covariances;

Support Vector Machines

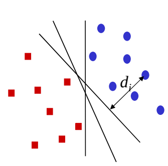
- Bayes classifiers and generative approaches in general try to model of the posterior, $p(\omega|\mathbf{x})$
- Instead, try to obtain the decision boundary directly
  - potentially easier, because we need to encode only the geometry of the boundary, not any irrelevant wiggles in the posterior.
  - Not all points affect the decision boundary

Support Vector Machines

- Set S of points $x_i \in R^n$, each x_i belongs to one of two classes $y_i \in \{-1,1\}$
- The goals is to find a hyperplane that divides S in these two classes

S is separable if $\exists \mathrm{w} \in R^n, b \in R$

$$y_i(\mathrm{w}.\mathrm{x}_i + b) \geq 1$$



Separating hyperplanes

$$\mathrm{w}.\mathrm{x} + b = 0$$

$$d_i = \frac{\mathrm{w}.\mathrm{x}_i + b}{w} \qquad w = \|\mathrm{w}\|$$

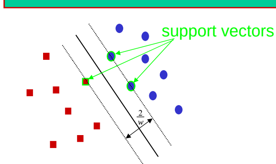
Closest point

$$y_i d_i = \frac{1}{w}$$

Support Vector Machines

- Optimal separating hyperplane maximizes $\frac{1}{w}$

Problem 1:
Minimize $\frac{1}{2}\mathrm{w}.\mathrm{w}$
Subject to $y_i(\mathrm{w}.\mathrm{x}_i + b) \geq 1, \quad i = 1,2,\ldots,N$



Optimal separating hyperplane (OSH)

Solve using Lagrange multipliers

- Lagrangian

$$L(\mathrm{w},b,\alpha)=\tfrac{1}{2}\,\mathrm{w}.\mathrm{w}-\sum_{i=1}^{N}\alpha_i\{y_i(\mathrm{w}.\mathrm{x}_i+b)-1\}$$

  – at solution

$$\frac{\partial L}{\partial b}=\sum_{i=1}^{N}y_i\alpha_i=0$$

$$\frac{\partial L}{\partial \mathrm{w}}=\mathrm{w}-\sum_{i=1}^{N}\alpha_i y_i\,\mathrm{x}_i=0$$

  – therefore

$$L(\boldsymbol{\alpha})=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j\,\mathrm{x}_i.\mathrm{x}_j$$

$$\boldsymbol{\alpha}\geq 0$$

Decision function

- Once w and b have been computed the classification decision for input x is given by

$$\boxed{f(\mathrm{x})=\mathrm{sign}(\mathrm{w}.\mathrm{x}+b)}$$

- Note that the globally optimal solution can always be obtained (convex problem)

Non-linear SVMs

- Non-linear separation surfaces can be obtained by non-linearly mapping the data to a high dimensional space and then applying the linear SVM technique
- Note that data only appears through vector product
- Need for vector product in high-dimension can be avoided by using Mercer kernels:

$$K(\mathrm{x}_i,\mathrm{x}_i)=\Phi(\mathrm{x}_i)\Phi(\mathrm{x}_i)$$

e.g. $K(\mathrm{x},\mathrm{y})=(\mathrm{x}.\mathrm{y})^p$ (Polynomial kernel)

$K(\mathrm{x},\mathrm{y})=(\mathrm{x}.\mathrm{y})^2=x_1^2y_1^2+x_1x_2y_1y_2+x_2^2y_2^2$

$\Phi(\mathrm{x})=(x_1^2,x_1x_2,x_2^2)$

$K(\mathrm{x},\mathrm{y})=\exp\left(\frac{-\|\mathrm{x}-\mathrm{y}\|^2}{\sigma^2}\right)$ (Radial Basis Function)

$K(\mathrm{x},\mathrm{y})=\tanh(\kappa\,\mathrm{x}.\mathrm{y}-\delta)$ (Sigmoïdal function)

Example: Finding skin
Non-parametric Representation of CCD

- Skin has a very small range of (intensity independent) colors, and little texture
  - Compute an intensity-independent color measure, check if color is in this range, check if there is little texture (median filter)
  - See this as a classifier - we can set up the tests by hand, or learn them.
  - get class conditional densities (histograms), priors from data (counting)
- Classifier is
  - if $p(\mathrm{skin}|\boldsymbol{x})>\theta$, classify as skin
  - if $p(\mathrm{skin}|\boldsymbol{x})<\theta$, classify as not skin
  - if $p(\mathrm{skin}|\boldsymbol{x})=\theta$, choose classes uniformly and at random

Figure from "Statistical color models with application to skin detection," M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE

Receiver
Operating
Curve

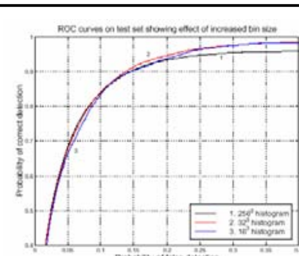


Figure from "Statistical color models with application to skin detection," M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE