Recognition
(Part 2)

Introduction to Computer Vision

CSE 152

Lecture 17

Announcements

- Homework 3 is due May 29, 11:59 PM
- Homework 4 is due next week
- Final exam will be a take home exam
- Reading:
  - Section 14.2.1 Eigenfaces
  - Section 14.4.1 Bag of words

Linear Subspaces & Linear Projection

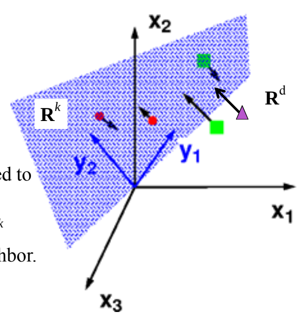
- A d-pixel image $\mathbf{x} \in \mathbf{R}^d$ can be projected to a low-dimensional feature space $\mathbf{y} \in \mathbf{R}^k$ by

$$\mathbf{y} = W\mathbf{x}$$

where W is an k by d matrix.

- Each training image is projected to the subspace
- Recognition is performed in $\mathbf{R}^k$ using, for example, nearest neighbor.
- How do we choose a good W?

Example: Projecting from $\mathbf{R}^3$ to $\mathbf{R}^2$



PCA & Fisher's Linear Discriminant

- Between-class scatter

$$S_B = \sum_{i=1}^{c} |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$$

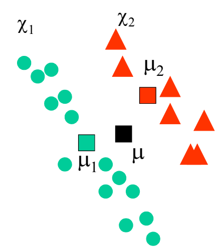
- Within-class scatter

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in \chi_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

- Total scatter

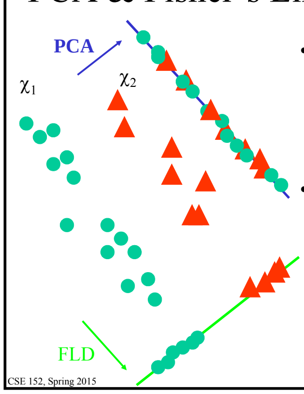
$$S_T = \sum_{i=1}^{c} \sum_{x_k \in \chi_i} (x_k - \mu)(x_k - \mu)^T = S_B + S_W$$

- Where
  - c is the number of classes
  - μ_i is the mean of class χ_i
  - $|\chi_i|$ is number of samples of χ_i.



If the data points x_i are projected by $y_i = Wx_i$ and the scatter of x_i is S, then the scatter of the projected points y_i is WSW^T

PCA & Fisher's Linear Discriminant



- PCA

$$W_{PCA} = \arg\max_W |W^T S_T W|$$

Maximizes projected total scatter

- Fisher's Linear Discriminant

$$W_{fld} = \arg\max_W \frac{|W^T S_B W|}{|W^T S_W W|}$$

Maximizes ratio of projected between-class to projected within-class scatter

Eigenfaces: Principal Component Analysis (PCA)

Assume we have a set of n feature vectors $\boldsymbol{x}_i$ $(i = 1, \ldots, n)$ in $\mathbb{R}^d$. Write

$$\boldsymbol{\mu} = \frac{1}{n} \sum_i \boldsymbol{x}_i$$

$$\Sigma = \frac{1}{n-1} \sum_i (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$$

The unit eigenvectors of Σ — which we write as $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d$, where the order is given by the size of the eigenvalue and $\boldsymbol{v}_1$ has the largest eigenvalue — give a set of features with the following properties:

- They are independent.
- Projection onto the basis $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ gives the k-dimensional set of linear features that preserves the most variance.

Algorithm 22.5: *Principal components analysis identifies a collection of linear features that are independent, and capture as much variance as possible from a dataset.*

Eigen decomposition of covariance matrix.

<u>Alternative</u>: singular value decomposition of (mean-deviation form of) data matrix.

Performing PCA with SVD

- Singular values of A are the square roots of eigenvalues of both AA^T and A^TA
- Columns of U are corresponding Eigenvectors of AA^T
- And $\sum_{i=1}^{n} a_i a_i^T = [a_1 \ a_2 \ \cdots \ a_n][a_1 \ a_2 \ \cdots \ a_n]^T = AA^T$
- Covariance matrix is:

$$\Sigma = \frac{1}{n-1}\sum_{i=1}^{n}(\vec{x}_i - \vec{\mu})(\vec{x}_i - \vec{\mu})^T$$

- So, ignoring 1/(n-1), subtract mean image μ from each input image, create a d x n data matrix, and perform thin SVD on the data matrix. $D=[x_1-\mu \mid x_2-\mu \mid \ldots \ x_n-\mu]$

Computing the Fisher Projection Matrix

$$W_{opt} = \arg\max_W \frac{|W^T S_B W|}{|W^T S_W W|} = [\mathbf{w}_1 \ \mathbf{w}_2 \ \ldots \ \mathbf{w}_m] \qquad (4)$$

where $\{\mathbf{w}_i \mid i = 1, 2, \ldots, m\}$ is the set of generalized eigenvectors of S_B and S_W corresponding to the m largest generalized eigenvalues $\{\lambda_i \mid i = 1, 2, \ldots, m\}$, i.e.,

$$S_B \mathbf{w}_i = \lambda_i S_W \mathbf{w}_i, \quad i = 1, 2, \ldots, m$$

- The w_i are orthonormal
- There are at most c-1 non-zero generalized Eigenvalues, so $m \le c-1$
- Can be computed with *eig* in Matlab

Fisherfaces

$$W = W_{fld} W_{PCA}$$

$$W_{PCA} = \arg\max_W |W^T S_T W|$$

$$W_{fld} = \arg\max_W \frac{|W^T W_{PCA}^T S_B W_{PCA} W|}{|W^T W_{PCA}^T S_W W_{PCA} W|}$$

- Since S_W is rank N-c, project training set to subspace spanned by first *N-c* principal components of the training set.
- Apply FLD to *N-c* dimensional subspace yielding *c-1* dimensional feature space.

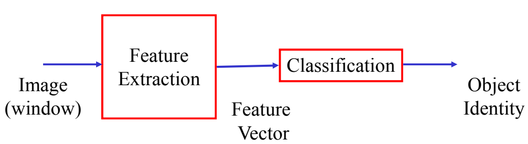
- Fisher's Linear Discriminant projects away the within-class variation (lighting, expressions) found in training set.
- Fisher's Linear Discriminant preserves the separability of the classes.

Appearance-Based Vision for Instances Level Recognition:

A Pattern Classification Viewpoint

1. Feature Space + Nearest Neighbor
2. Dimensionality Reduction
3. Bayesian Classification
4. Appearance Manifolds

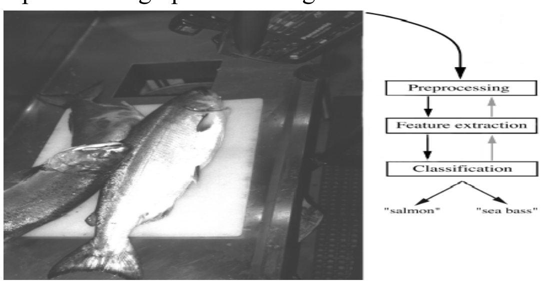
Sketch of a Pattern Recognition Architecture

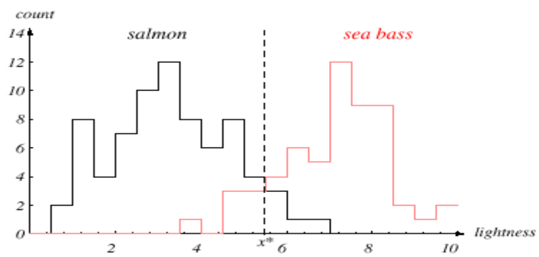


Bayesian Classification

Example: Sorting

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

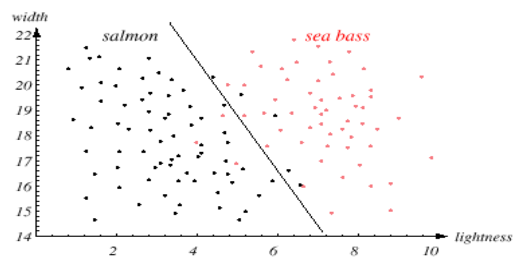


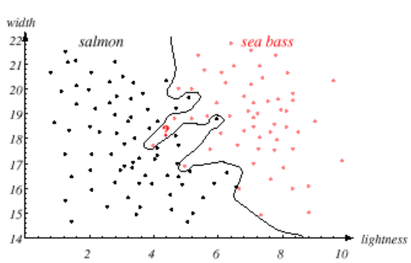


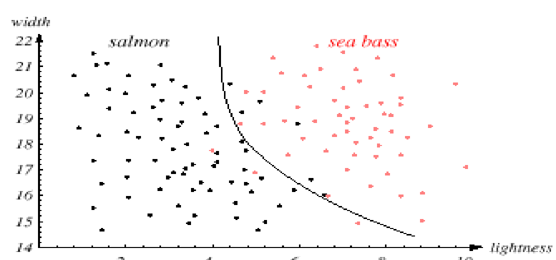
- Adopt the lightness and add the width of the fish

Fish → $x^T = [x_1, x_2]$

Lightness **Width**







Basic ideas in classifiers

- Loss
  - some errors may be more expensive than others
    - e.g. a fatal disease that is easily cured by a cheap medicine with no side-effects -> false positives in diagnosis are better than false negatives
  - We discuss two class classification: L(1->2) is the loss caused by calling 1 a 2
- Total risk of using classifier s

$$R(s) = Pr\{1 \to 2 | \text{using } s\} L(1 \to 2) + Pr\{2 \to 1 | \text{using } s\} L(2 \to 1)$$

Basic ideas in classifiers

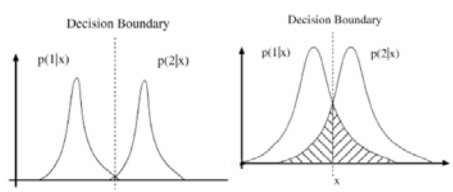
- Generally, we should classify as 1 if the expected loss of classifying as 1 is better than for 2
- gives

1 if $\quad p(1|\boldsymbol{x})L(1 \to 2) > p(2|\boldsymbol{x})L(2 \to 1)$

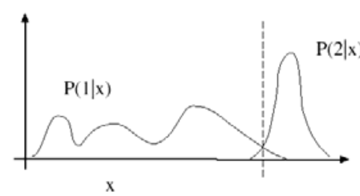
2 if $\quad p(1|\boldsymbol{x})L(1 \to 2) < p(2|\boldsymbol{x})L(2 \to 1)$

- Crucial notion: Decision boundary
  - points where the loss is the same for either case

Some loss may be inevitable: the minimum risk (shaded area) is called the Bayes risk



Finding a decision boundary is not the same as modelling a conditional density.



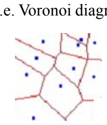
- Classifier boils down to: choose class that minimizes:

$$\delta(\underline{x}, \underline{\mu}_k)^2 - 2\log \pi_k$$

where

Mahalanobis distance — $$\delta(\underline{x}, \underline{\mu}_k) = \left[(\underline{x} - \underline{\mu}_k)^T \underline{\underline{\Sigma}}^{-1} (\underline{x} - \underline{\mu}_k)\right]^{1/2}$$

because covariance is common, this simplifies to sign of a linear expression (i.e. Voronoi diagram in 2D for Σ=I)



Plug-in classifiers

- Assume that class conditional distributions P(x|ωi) 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
- Issue: parameter estimates that are “good” may not give optimal classifiers.

Example: Finding skin

- 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)
- Classific
  - if $p(\text{skin}|\boldsymbol{x}) > \theta$, classify as skin
  - if $p(\text{skin}|\boldsymbol{x}) < \theta$, classify as not skin
  - if $p(\text{skin}|\boldsymbol{x}) = \theta$, 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

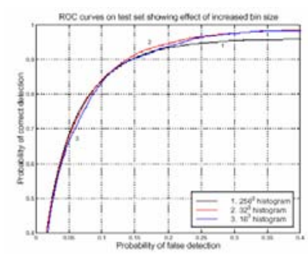
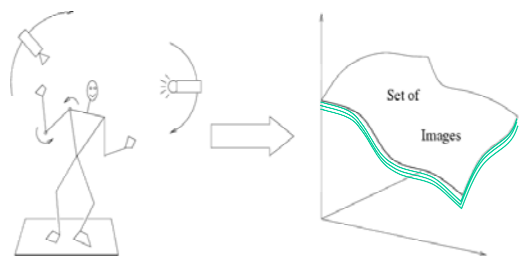


Figure from "Statistical color models with application to skin detection," M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE



Variability: Camera position
Illumination
Internal parameters

→ Within-class variations

Appearance manifold approach

(Nayar et al. '96)

- for every object
  1. sample the set of viewing conditions
  2. Crop & scale images to standard size
  3. Use as feature vector
- apply a PCA over all the images
- keep the dominant PCs
- Set of views for one object is represented as a manifold in the projected space
- Recognition: What is nearest manifold for a given test image?

Limitations of these approaches

- Object must be segmented from background (How would one do this in non-trivial situations?)
- Occlusion?
- The variability (dimension) in images is large, so is sampling feasible?
- How can one generalize to classes of objects?

Appearance-Based Vision: Lessons

Strengths

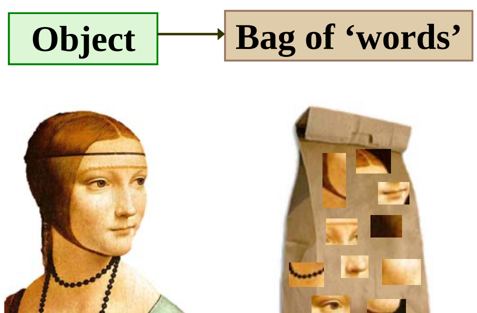
- Posing the recognition metric in the image space rather than a derived representation is more powerful than expected.
- Modeling objects from many images is not unreasonable given hardware developments.
- The data (images) may provide a better representations than abstractions for many tasks.

Appearance-Based Vision: Lessons

Weaknesses

- Segmentation or object detection is still an issue.
- To train the method, objects have to be observed under a wide range of conditions (e.g. pose, lighting, shape deformation).
- Limited power to extrapolate or generalize (abstract) to novel conditions.

Bag-of-features models

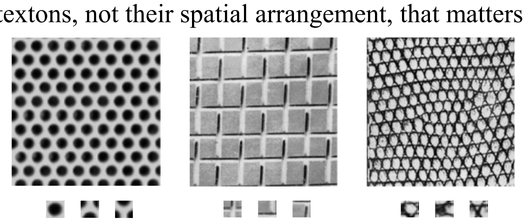


 Slides from Svetlana Lazebnik who borrowed from others

Bag-of-features models

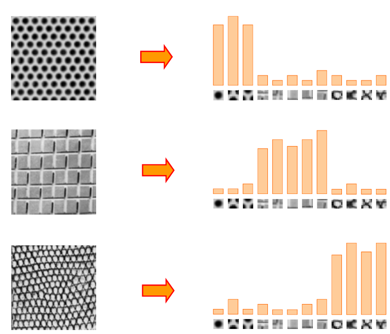
Origin 1: Texture recognition

- Texture is characterized by the repetition of basic elements or *textons*
- For stochastic textures, it is the identity of the textons, not their spatial arrangement, that matters



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Origin 1: Texture recognition



Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Which US President?
Franklin D. Roosevelt, John F. Kennedy, George W. Bush

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Which US President?
Franklin D. Roosevelt, John F. Kennedy, George W. Bush

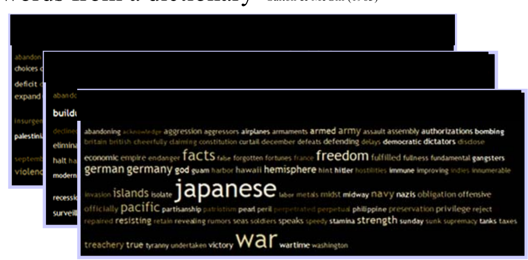
Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Which US President?
Franklin D. Roosevelt, John F. Kennedy, George W. Bush

Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)



Which US President?
Franklin D. Roosevelt, John F. Kennedy, George W. Bush

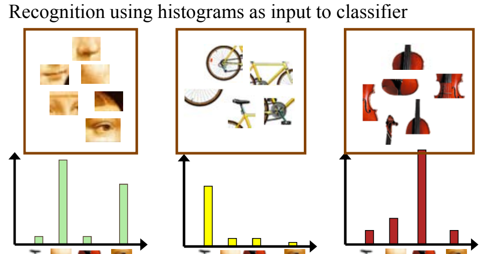
Origin 2: Bag-of-words models

- Orderless document representation: frequencies of words from a dictionary Salton & McGill (1983)

Which US President?
Franklin D. Roosevelt, John F. Kennedy, George W. Bush

Bag-of-features steps

1. Extract features
2. Learn "visual vocabulary"
3. Quantize features using visual vocabulary
4. Represent images by frequencies (histogram) of "visual words"
5. Recognition using histograms as input to classifier



Feature extraction

- Regular grid or interest regions