

Recognition III

Introduction to Computer Vision
CSE 152
Lecture 19

CSE152, Spr 11

Intro Computer Vision

Announcements

- HW 4 due Friday
- Final Exam: Tuesday, 6/7 at 8:00-11:00
-

CSE152, Spr 11

Intro Computer Vision

CSE Peer Mentoring Program *seeking volunteers*

- Was your first quarter at UCSD hard?
- Would you like to help others through this time?
- We are seeking volunteers for a new peer mentoring program
 - Each mentor will work with a few new majors
 - just a couple hours per week
 - Mentors will be advised by CSE graduate students
 - Mentorship will look great on your resume
- Visit this URL to fill out a short form, and we'll contact you over the summer:
<http://goo.gl/xLAAj>
- Questions? Contact Bill Griswold: wgg@cs.ucsd.edu

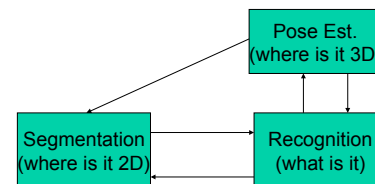
CSE152, Spr 11

Intro Computer Vision

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

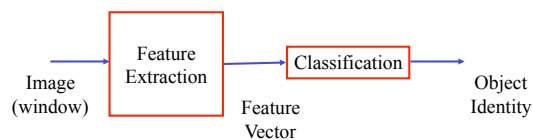


WHAT AND WHERE!!!

CSE152, Spr 11

Intro Computer Vision

Sketch of a Pattern Recognition Architecture



CSE152, Spr 11

Intro Computer Vision

Example: Face Detection

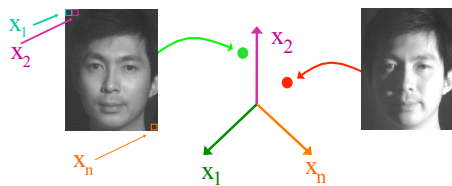
- Scan window over image.
- Search over position & scale.
- Classify window as either:
 - Face
 - Non-face



CSE152, Spr 11

Intro Computer Vision

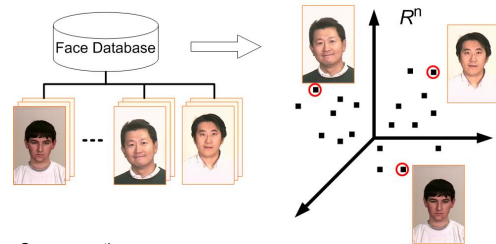
The Space of Images



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

Face Space:

- A set of face images construct a *face space* in $\mathbb{R}^n$
- Appearance-based methods analyze the distributions of individual faces in face space



Some questions:

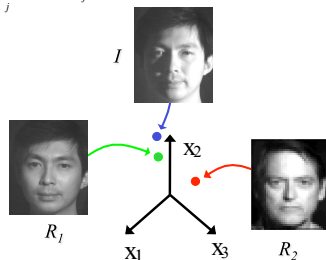
- How are images of an individual, under all conditions, distributed in this space?
- How are the images of all individuals distributed in this space?

CSE152

Computer Vision I

Nearest Neighbor Classifier

$\{R_j\}$ are set of training images.
 $ID = \arg \min_j \text{dist}(R_j, I)$



CSE152, Spr 11

Intro Computer Vision

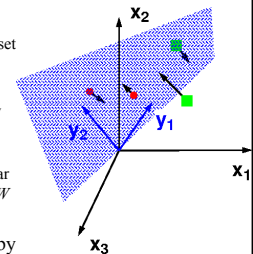
An idea:

Represent the set of images as a linear subspace

What is a linear subspace?

Let V be a vector space and let W be a subset of V . Then W is a subspace iff:

- The zero vector, $\mathbf{0}$, is in W .
- If $\mathbf{u}$ and $\mathbf{v}$ are elements of W , then any linear combination of $\mathbf{u}$ and $\mathbf{v}$ is an element of W ; $a\mathbf{u} + b\mathbf{v} \in W$
- If $\mathbf{u}$ is an element of W and c is a scalar from K , then the scalar product $c\mathbf{u} \in W$



A d -dimensional subspace is spanned by d linearly independent vectors

It is spanned by a d -dimensional orthogonal basis.

Example: A 2-D linear subspace of $\mathbb{R}^3$ spanned by y_1 and y_2 .

CSE152

Computer Vision I

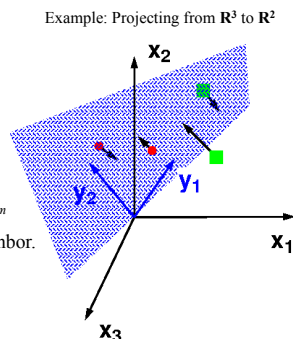
Linear Subspaces & Linear Projection

- An n -pixel image $\mathbf{x} \in \mathbb{R}^n$ can be projected to a low-dimensional feature space $\mathbf{y} \in \mathbb{R}^m$ by

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

where W is an m by n matrix.

- Recognition is performed in $\mathbb{R}^m$ using, for example, nearest neighbor.
- How do we choose a good W ?



CSE152, Spr 11

Intro Computer Vision

Linear Subspaces & Recognition

- Approximate all training images as a single linear subspace (Eigenfaces).
- Represent lighting variation w/o shadowing for a single individual as a 3-D linear subspace. A collection of n individuals is modeled as n 3-D linear subspaces.
- Represent lighting variation w/ shadowing for a single individual as a 9-D linear subspace. A collection of n individuals is modeled as n 9-D linear subspaces.
- Project all training images to a single subspace that enhances discriminability (Fisherfaces).

CSE152, Spr 11

Intro Computer Vision

Eigenfaces: Principal Component Analysis (PCA)

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

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

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

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

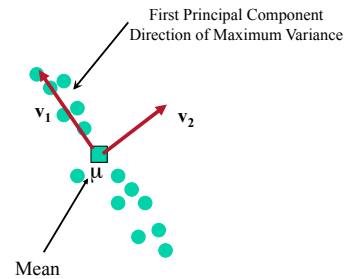
- They are independent.
- Projection onto the basis $\{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ gives the k -dimensional set of linear features that preserves the most variance.

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.

CSE152, Spr 11

Intro Computer Vision

PCA Example



CSE152, Spr 11

Intro Computer Vision

Eigenfaces

- Modeling
 1. Given a collection of n labeled training images,
 2. Compute mean image and covariance matrix.
 3. Compute k Eigenvectors (note that these are images) of covariance matrix corresponding to k largest Eigenvalues. (Or perform using SVD!!)
 4. Project the training images to the k -dimensional Eigenspace.
- Recognition
 1. Given a test image, project vectorized image to Eigenspace.
 2. Perform classification to the projected training images.

CSE152, Spr 11

Intro Computer Vision

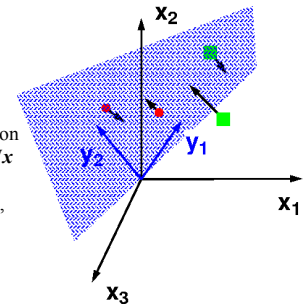
Distance to Subspace

- An n -pixel image $\mathbf{x} \in \mathbb{R}^n$ can be projected to a low-dimensional feature space $\mathbf{y} \in \mathbb{R}^m$ by

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

- From $\mathbf{y} \in \mathbb{R}^m$, the reconstruction of the point in $\mathbb{R}^n$ is $\mathbf{W}^T\mathbf{y} = \mathbf{W}^T\mathbf{W}\mathbf{x}$

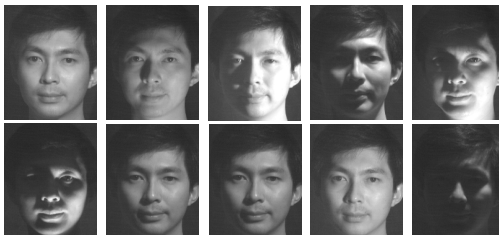
- The error of the reconstruction, or the distance from $\mathbf{x}$ to the subspace spanned by $\mathbf{W}$ is: $\|\mathbf{x} - \mathbf{W}^T\mathbf{W}\mathbf{x}\|$



CSE152, Spr 11

Intro Computer Vision

Illumination Variability



- How does the set of images of an object in fixed pose vary as lighting changes?
- How can we recognize people across all lighting conditions without having to see the person every way?

CSE152, Spr 11

Intro Computer Vision

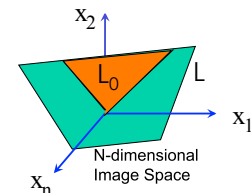
3-D Linear subspace

The set of images of a Lambertian surface with no shadowing is a subset of 3-D linear subspace.

[Moses 93], [Nayar, Murase 96], [Shashua 97]

$$\mathbf{L} = \{\mathbf{x} \mid \mathbf{x} = \mathbf{B}\mathbf{s}, \forall \mathbf{s} \in \mathbb{R}^3\}$$

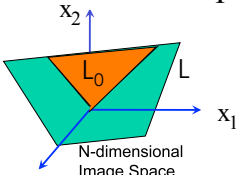
where $\mathbf{B}$ is a n by 3



CSE152, Spr 11

Intro Computer Vision

Distance to Linear Subspace



- From a collection of images of person i under variable lighting, compute basis $\mathbf{B}_i$ images of a 3-D linear subspace using PCA or SVD.
- Recognize individual by finding subspace with shortest distance from a test image to the subspace.

$$\mathbf{d}_i = \|\mathbf{x} - \mathbf{B}_i \mathbf{B}_i^T \mathbf{x}\|$$

Intro Computer Vision

CSE152, Spr 11

Intro Computer Vision

What about shadows?

- The set of images of a Lambertian surface with no shadowing is a subset of 3-D linear subspace.
[Moses 93], [Nayar, Murase 96], [Shashua 97]
- The set of images of a convex Lambertian surfaces is a convex cone in the spaces of images. [Belhumeur, Kriegman 1998]
- The set of images of a convex Lambertian surfaces is well-approximated by a 9-D linear subspace.
[Basri, Jacobs 2001], [Ramamoorthi, Hanrahan 2001]
- The 9-D linear subspace approximating the set of images of a convex Lambertian surfaces is approximately spanned by nine images taken under 9 well-chosen light sources. [Lee, Ho, Kriegman 2005]

CSE152, Spr 11 Intro Computer Vision

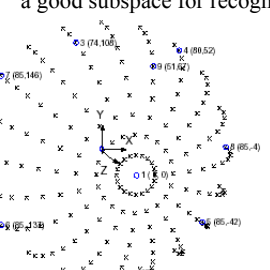
CSE152, Spr 11

Intro Computer Vision

Acquiring Subspace for Recognition

Lee, Ho, Kriegman, PAMI 2005

- Can we find a way to gather real images that span a good subspace for recognition



CS1

10/11

CSI

ion

An important footnote:

We don't really implement PCA by constructing a covariance matrix!

Why?

1. How big is Σ ?
 - n by n where n is the number of pixels in an image!!
2. You only need the first k Eigenvectors

CSE152, Spr 11

Intro Computer Vision

Singular Value Decomposition

- Any m by n matrix A may be factored such that

$$A = U \Sigma V^T$$

$$[m \times n] = [m \times m][m \times n][n \times n]$$
- U : m by m , orthogonal matrix
 - Columns of U are the eigenvectors of AA^T
- V : n by n , orthogonal matrix,
 - columns are the eigenvectors of $A^T A$
- Σ : m by n , diagonal with non-negative entries $(\sigma_1, \sigma_2, \dots, \sigma_s)$ with $s = \min(m, n)$ are called the singular values
 - Singular values are the square roots of eigenvalues of both AA^T and $A^T A$ & Columns of U are corresponding Eigenvectors!!**
 - Result of SVD algorithm:* $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_s$

CSE152, Spr 11 Intro Computer Vision

CSE152, Spr 11

Intro Computer Vision

SVD Properties

- In Matlab $[u \ s \ v] = \text{svd}(A)$, and you can verify that: $A = u * s * v'$
- $r = \text{Rank}(A) = \#$ of non-zero singular values.
- U, V give an orthonormal bases for the subspaces of A :
 - 1st r columns of U : Column space of A
 - Last $m - r$ columns of U : Left nullspace of A
 - 1st r columns of V : Row space of A
 - Last $n - r$ columns of V : Nullspace of A
- *For some d where $d \leq r$, the first d column of U provide the best d -dimensional basis for columns of A in least squares sense.*

CSE152, Spr 11
Intro Computer Vision

CSE152, Spr 11

Intro Computer Vision

Thin SVD

- Any m by n matrix A may be factored such that

$$A = U \Sigma V^T$$

$$[m \times n] = [m \times m][m \times n][n \times n]$$

- If $m > n$, then one can view Σ as:

$$\begin{bmatrix} \Sigma \\ 0 \end{bmatrix}$$

- Where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_s)$ with $s = \min(m, n)$, and lower matrix is $(n-m \text{ by } m)$ of zeros.

- Alternatively, you can write:

$$A = U \Sigma' V^T$$

This is what you should use!!

- In Matlab, thin SVD is: $[U \Sigma V] = \text{svds}(A)$

CSE152, Spr 11

Intro Computer Vision

Performing PCA with SVD

- Singular values of A are the square roots of eigenvalues of both AA^T and $A^T A$ & Columns of U are corresponding Eigenvectors

- And $\sum_{i=1}^n a_i a_i^T = \begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix} \begin{bmatrix} a_1 & a_2 & \dots & a_n \end{bmatrix}^T = AA^T$

- Covariance matrix is:

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

- So, ignoring $1/n$ subtract mean image μ from each input image, create data matrix, and perform thin SVD on the data matrix.

CSE152, Spr 11

Intro Computer Vision

Alternative projections

CSE152, Spr 11

Intro Computer Vision

Fisherfaces: Class specific linear projection

P. Belhumeur, J. Hespanha, D. Kriegman, *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection*, PAMI, July 1997, pp. 711-720.

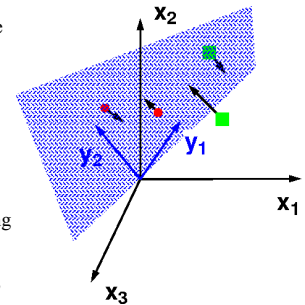
- An n -pixel image $x \in \mathbb{R}^n$ can be projected to a low-dimensional feature space $y \in \mathbb{R}^m$ by

$$y = Wx$$

where W is an n by m matrix.

- Recognition is performed using nearest neighbor in $\mathbb{R}^m$.

- How do we choose a good W ?



CSE152, Spr 11

Intro Computer Vision

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

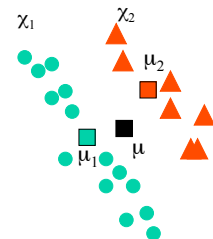
$$S_W = \sum_{i=1}^c \sum_{x_i \in \chi_i} (x_i - \mu_i)(x_i - \mu_i)^T$$

- Total scatter

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

- 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 .

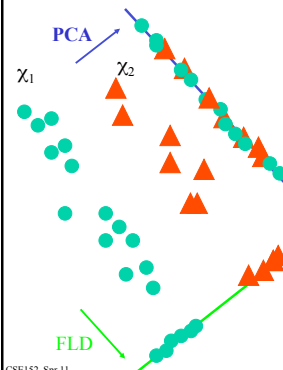


- If the data points are projected by $y = Wx$ and scatter of points is S , then the scatter of the projected points is $W^T S W$

CSE152, Spr 11

Intro Computer Vision

PCA & Fisher's Linear Discriminant



- PCA (Eigenfaces)

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

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

CSE152, Spr 11

Intro Computer Vision

Computing the Fisher Projection Matrix

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

$$= [w_1 \ w_2 \ \dots \ w_m] \quad (4)$$

where $\{w_i | i = 1, 2, \dots, m\}$ is the set of generalized eigenvectors of S_B and S_W corresponding to the m largest generalized eigenvalues $\{\lambda_i | i = 1, 2, \dots, m\}$, i.e.,

$$S_B w_i = \lambda_i S_W w_i, \quad i = 1, 2, \dots, m$$

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

CSE152, Spr 11

Intro Computer Vision

Fisherfaces

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

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

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

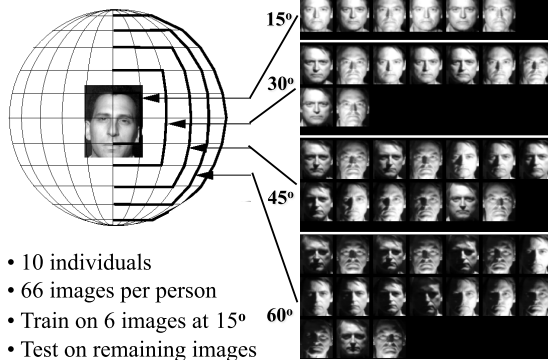
- Since S_W is rank $N-c$, project training set to subspace spanned by first $N-c$ principal components of the training set.
- Apply FLD to $N-c$ dimensional subspace yielding $c-1$ dimensional feature space.

- 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.

CSE152, Spr 11

Intro Computer Vision

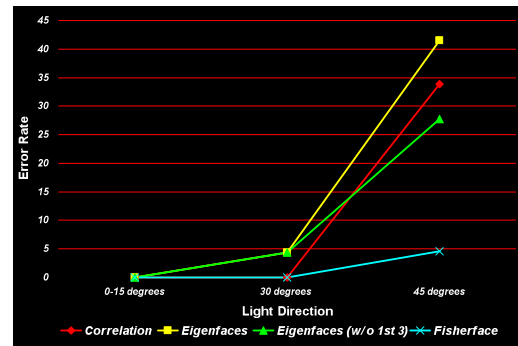
Harvard Face Database



CSE152, Spr 11

Intro Computer Vision

Recognition Results: Lighting Extrapolation



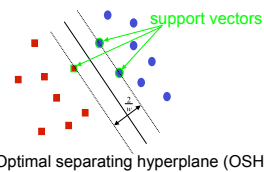
CSE152, Spr 11

Intro Computer Vision

Support Vector Machines

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

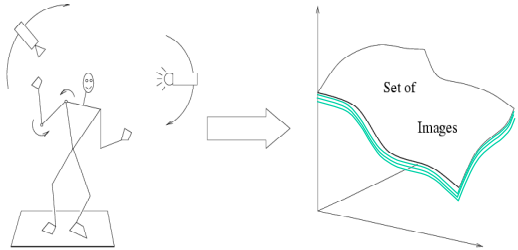
Problem 1:
 Minimize $\frac{1}{2} W \cdot W$
 Subject to $y_i (w \cdot x_i + b) \geq 1, \quad i = 1, 2, \dots, N$



CSE152, Spr 11

[From Marc Pollefeys]

Intro Computer Vision



Variability: Camera position
 Illumination
 Internal parameters



Within-class variations

CSE152, Spr 11

Intro Computer Vision

Appearance manifold approach

- for every object (Nayar et al. '96)
 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?



CSE152, Spr 11

Intro Computer Vision

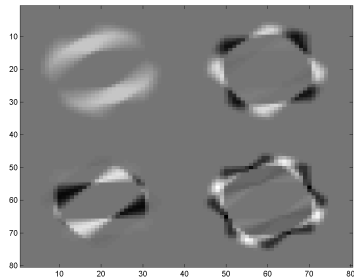
An example: input images



CSE152, Spr 11

Intro Computer Vision

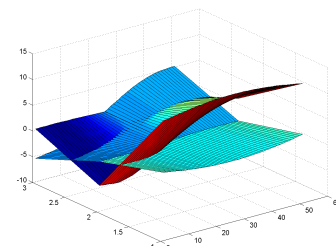
An example: basis images



CSE152, Spr 11

Intro Computer Vision

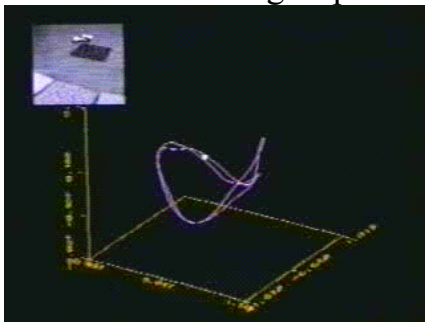
An example: surfaces of first 3 coefficients



CSE152, Spr 11

Intro Computer Vision

Parameterized Eigenspace



CSE152, Spr 11

Intro Computer Vision

Employ spatial relations

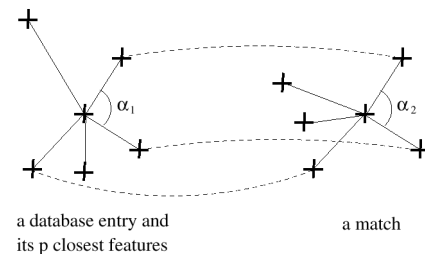


Figure from "Local grayvalue invariants for image retrieval," by C. Schmid and R. Mohr, IEEE Trans. Pattern Analysis and Machine Intelligence, 1997 copyright 1997, IEEE

CSE152, Spr 11

Intro Computer Vision

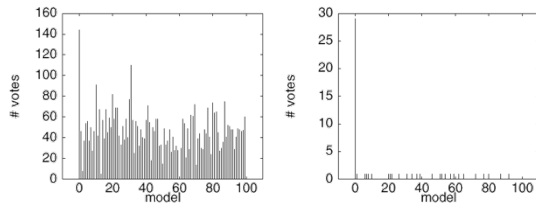
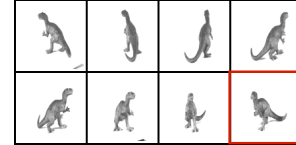


Figure from "Local grayvalue invariants for image retrieval," by C. Schmid and R. Mohr, IEEE Trans. Pattern Analysis and Machine Intelligence, 1997 copyright 1997, IEEE

Example

Training examples



Test image



Finding faces using relations

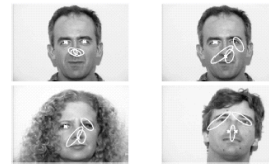
- Strategy:
 - Face is eyes, nose, mouth, etc. with appropriate relations between them
 - build a specialised detector for each of these (template matching) and look for groups with the right internal structure
 - Once we've found enough of a face, there is little uncertainty about where the other bits could be

Finding faces using relations

- Strategy: compare

$P(\text{one face at } F | X_e = x_1, X_{re} = x_2, X_m = x_3, X_n = x_4, \text{all other responses})$
with

$P(\text{no face} | X_e = x_1, X_{re} = x_2, X_m = x_3, X_n = x_4, \text{all other responses})$



Notice that once some facial features have been found, the position of the rest is quite strongly constrained.

Figure from, "Finding faces in cluttered scenes using random labelled graph matching," by Leung, T.; Burl, M and Perona, P., Proc. Int. Conf. on Computer Vision, 1995 copyright 1995, IEEE

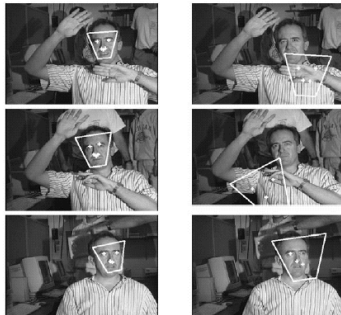


Figure from, "Finding faces in cluttered scenes using random labelled graph matching," by Leung, T.; Burl, M and Perona, P., Proc. Int. Conf. on Computer Vision, 1995 copyright 1995, IEEE