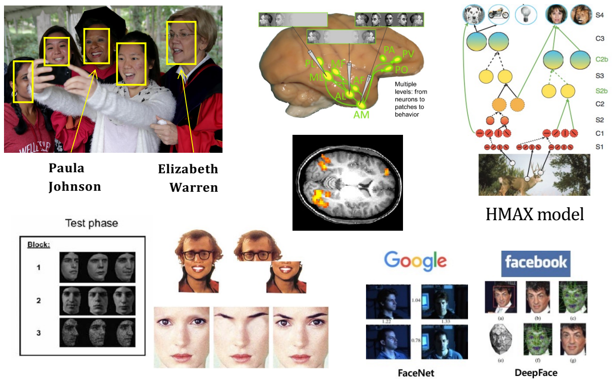
The recognition of faces



1

Why is face analysis important?

Remember/recognize people we've seen before

Categorization – e.g. gender, race, age, kinship

Social communication – emotions/mood, intentions, trustworthiness, competence or intelligence, attractiveness

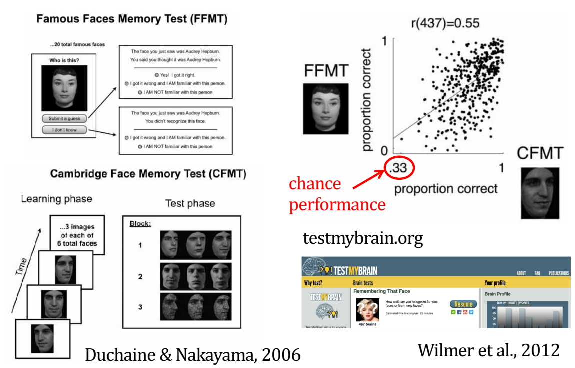
Scene understanding, e.g. direction of gaze suggests focus of attention

2

Why is face recognition hard?

3

Face recognition performance in humans



Duchaine & Nakayama, 2006

Wilmer et al., 2012

4

How good are the best machines?

Public databases of face images serve as benchmarks:

Labeled Faces in the Wild (LFW, http://vis-www.cs.umass.edu/lfw)
> 13,000 images of celebrities, 5,749 different identities

YouTube Faces Database (YTF, http://www.cs.tau.ac.il/~wolf/ytfaces)
3,425 videos, 1,595 different identities

Private face image datasets:

(Facebook) Social Face Classification dataset
4.4 million face photos, 4,030 different identities
(Google) 100-200 million face images, ~ 8 million different identities

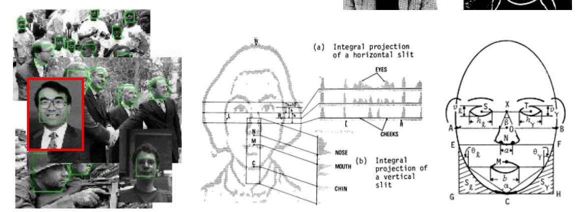
	LFW	YTF
Facebook DeepFace	97.4%	91.4%
Google FaceNet	99.6%	95.1%
Human performance	97.5%	89.7%

5

It all began with Takeo Kanade (1973)…

PhD thesis, *Picture Processing System by Computer Complex and Recognition of Human Faces*

- Special purpose algorithms to locate eyes, nose, mouth, boundaries of face
- ~ 40 geometric features, e.g. ratios of distances and angles between features

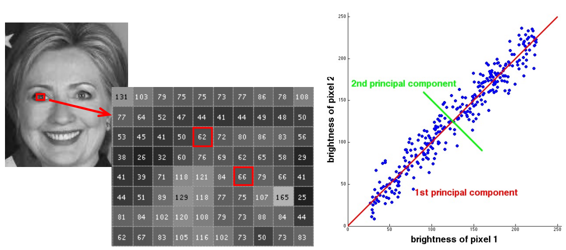


6

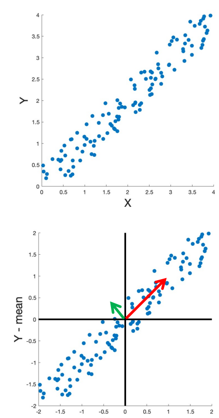
Eigenfaces for recognition (Turk & Pentland)
Principal Components Analysis (PCA)

Goal: reduce the dimensionality of the data while retaining as much information as possible in the original dataset

PCA allows us to compute a linear transformation that maps data from a high dimensional space to a lower dimensional subspace



7



$$\mathrm{var}(X) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(X_i - \overline{X})}{(n-1)}$$

$$\mathrm{cov}(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{(n-1)}$$

$$\begin{bmatrix} \mathrm{var}(X) & \mathrm{cov}(Y,X) \\ \mathrm{cov}(X,Y) & \mathrm{var}(Y) \end{bmatrix}$$

covariance matrix

$\boxed{(x,y) = w_1 {*} E_1 + w_2 {*} E_2}$

Eigenvector of Matrix A

$$\mathbf{Ax} = \lambda \mathbf{x}$$

Eigenvalue of Matrix A

$$\underset{\mathbf{A}}{\begin{bmatrix} 1 & -3 & 3 \\ 3 & -5 & 3 \\ 6 & -6 & 4 \end{bmatrix}} \underset{\mathbf{x}}{\begin{bmatrix} 1/2 \\ 1/2 \\ 1 \end{bmatrix}} \underset{\lambda}{=\ 4} \underset{\mathbf{x}}{\begin{bmatrix} 1/2 \\ 1/2 \\ 1 \end{bmatrix}}$$

eigenvectors of the covariance matrix are the principal components of the data

- orthogonal
- form a basis set
- eigenvector associated with largest eigenvalue is 1st principal component

8

Typical sample training set…

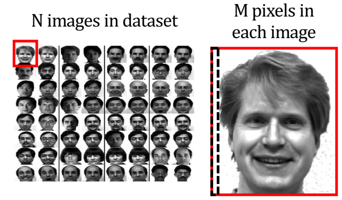
one or more images per person

aligned & cropped to common pose, size

simple background

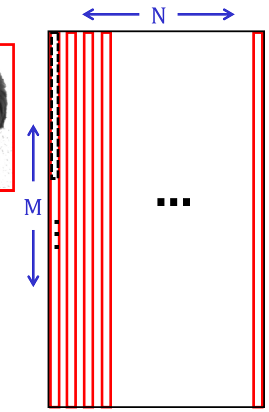
Sample images from the Yale face database

9

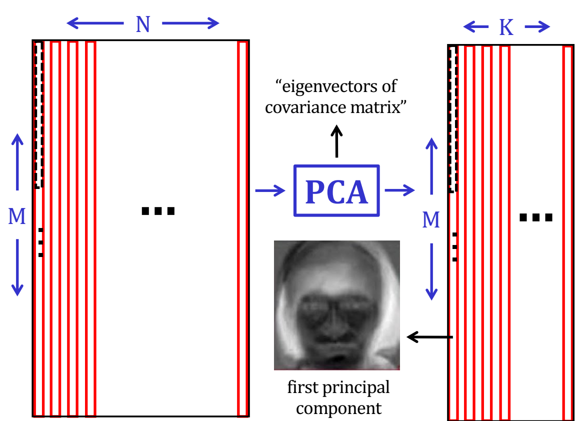


Prepare image data for PCA:

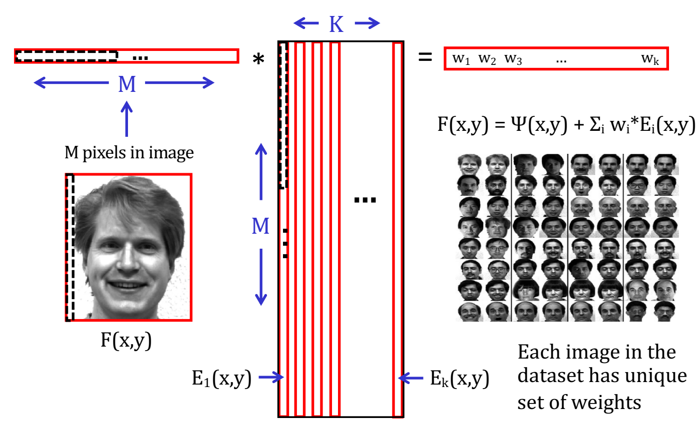
- For each image in dataset, place columns end-to-end to create one long column vector
- Place column vectors for each image side-by-side in an MxN matrix
- Subtract the mean vector (average face) from each column



10



11



12

Eigenfaces for recognition (Turk & Pentland)

$\Psi(x,y)$

Perform ***PCA*** on a large set of training images, to create a set of *eigenfaces*, $E_i(x,y)$, that span the dataset

First components capture most of the variation across the dataset, later components capture subtle variations

$\Psi(x,y)$: average face (across all faces)

http://vismod.media.mit.edu/vismod/demos/facerec/basic.html

Each face image F(x,y) can be expressed as a weighted combination of the eigenfaces $E_i(x,y)$:

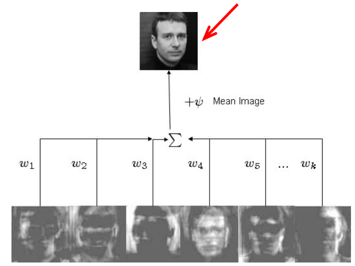
$$F(x,y) = \Psi(x,y) + \Sigma_i\, w_i * E_i(x,y)$$

13

Representing individual faces

Each face image F(x,y) can be expressed as a weighted combination of the eigenfaces $E_i(x,y)$:

$$F(x,y) = \Psi(x,y) + \Sigma_i\, w_i * E_i(x,y)$$



Recognition process:

(1) Compute weights w_i for novel face image

(2) Find image m in face database with most similar weights, e.g.

$$\min \sum_{i=1}^{k} (w_i - w_i^m)^2$$

14