Imperial College London

Face Recognition by Eigenfaces

Subspace, PCA

Tae-Kyun Kim
Senior Lecturer
http://www.iis.ee.ic.ac.uk/ComputerVision/

Prerequisite:

Linear algebra (EE310)

- Eigenvector/Eigenvalue
- Basis vectors/Subspace
- Orthogonal/orthonormal vectors

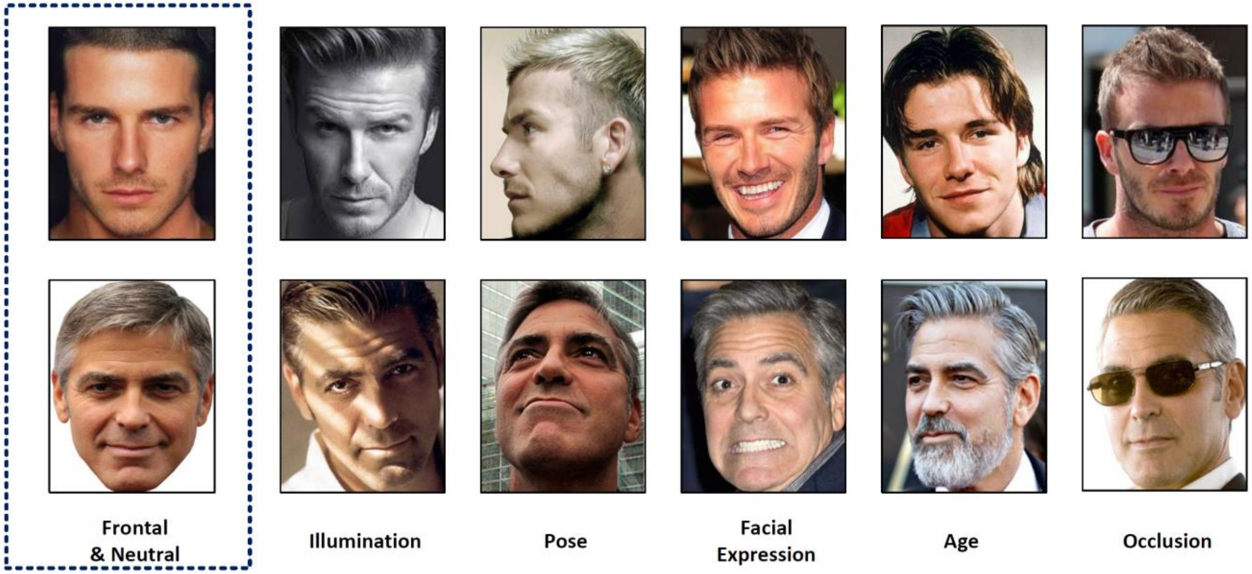
Further reading:

M.Turk and A.Pentland, Eigenfaces for recognition, 1991.

http://www.face-rec.org/algorithms/PCA/jcn.pdf

Motivation

- The face is a primary object of interest in visual recognition, conveying information on person identity, emotion, gaze, aging, etc.
- The human ability to recognize faces is remarkable. We can recognize thousands of faces learned in our lifetime and identify familiar faces at a glance.
- The human skill is quite robust against large changes in the visual stimulus caused by lighting conditions, poses, expression, aging, and occlusions due to glasses or facial hair.



Images in the same row share the same identity. For each ID, five types of variations are illustrated.

Motivation

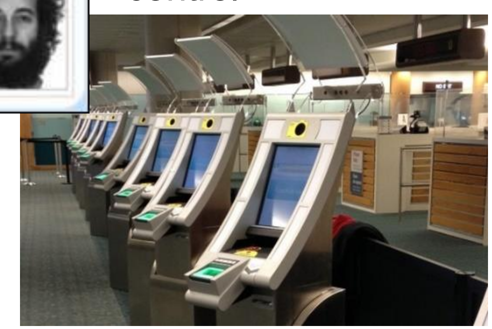
- Face recognition is an important topic for both human vision and computer vision.
- Computational models for face recognition contribute to a wide range of practical applications.
- Computers that recognize faces has been applied to, including: image and film processing, criminal identification, security systems, and human-computer interaction.

Automatic face image tagging at digital photo albums

Feature length film character summarisation

Face image retrieval in MPEG7

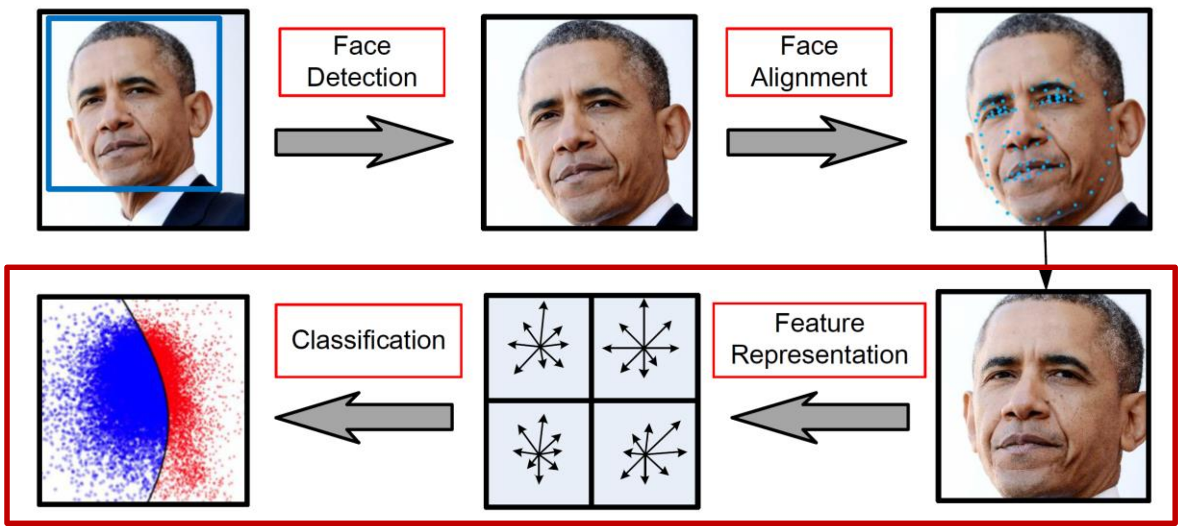
Automatic access control



(see http://www.iis.ee.ic.ac.uk/ComputerVision/Research.html)

Introduction

- Our goal is to develop a real-time computer system that can locate and track a subject's face (face detection), and then recognize the person ID by comparing characteristics of the face to those of known individuals (face recognition).
- Even the ability to detect faces or facial landmarks (face alignment), as opposed to recognizing them, is very useful.
- This lecture is about designing a computational model of face recognition that is fast, reasonably simple, and accurate. It consists of feature representation and classification.

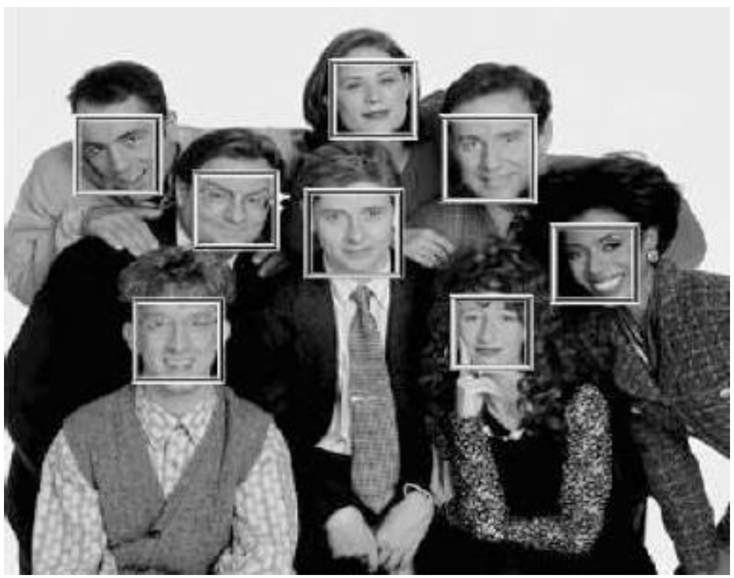


This lecture focuses on feature representation/classification.

Face Detection vs Face Recognition

– Face detection is to estimate locations and sizes (bounding boxes) of a known object class (i.e. human face) in a given image.

– Face recognition (identification or verification) is to estimate person's identity given a face image.

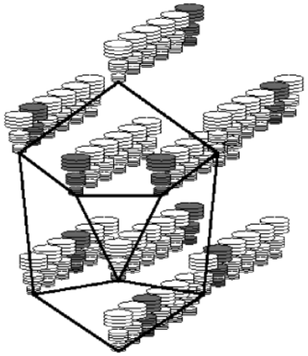


Viola et al. CVPR01

Kim et al. IVC05

Background and Related Work

- Much of the prior work in Automatic Face Recognition (AFR) has focused on detecting individual features such as eyes, nose, mouth, and head outline (facial landmarks), and defining a face model by the position, size, and relationships among these features (e.g. Kanade 1973, Wong, Law, & Tsaug, 1989).

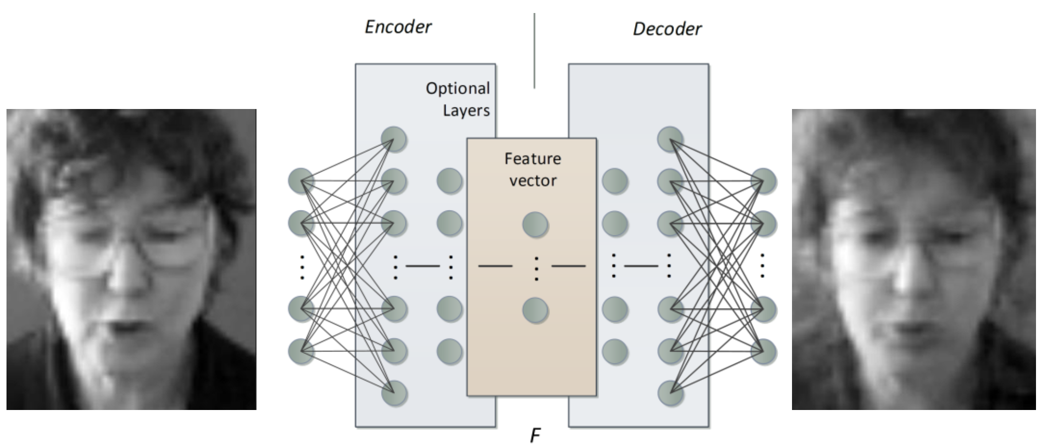


The (Gabor) bunch graph representation of faces used in elastic graph matching [Wiskott et al.1997].

- Such approaches have shown difficult to extend to multiple views, and have often been fragile.
- Research in human vision for face recognition has also shown that individual features and their geometrical relationships do not provide sufficient information to account for the performance of adult human face identification (Carey & Diamond, 1977).

Background and Related Work

- The perspective of information theory, coding and decoding face images, gives an insight into the information content of facial images.
- The significant local and global features found may or may not be directly related to our intuitive notion of facial features such as eyes, nose, and lips.
- In the perspective of information theory, we want to extract/encode the relevant information in a face image as efficiently as possible, and compare one face encoding with a database of models encoded in the same way.



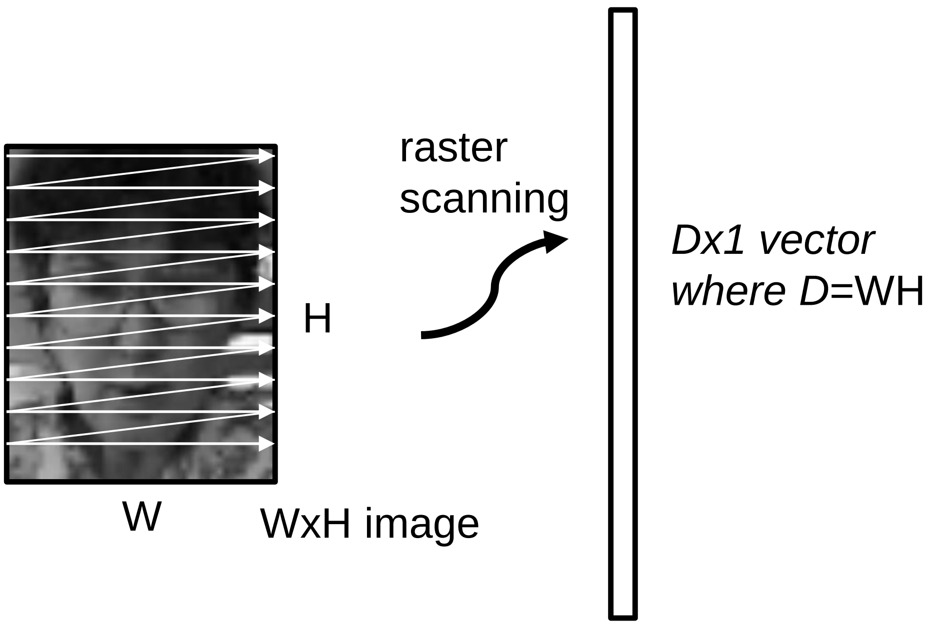
Sparse Autoencoder

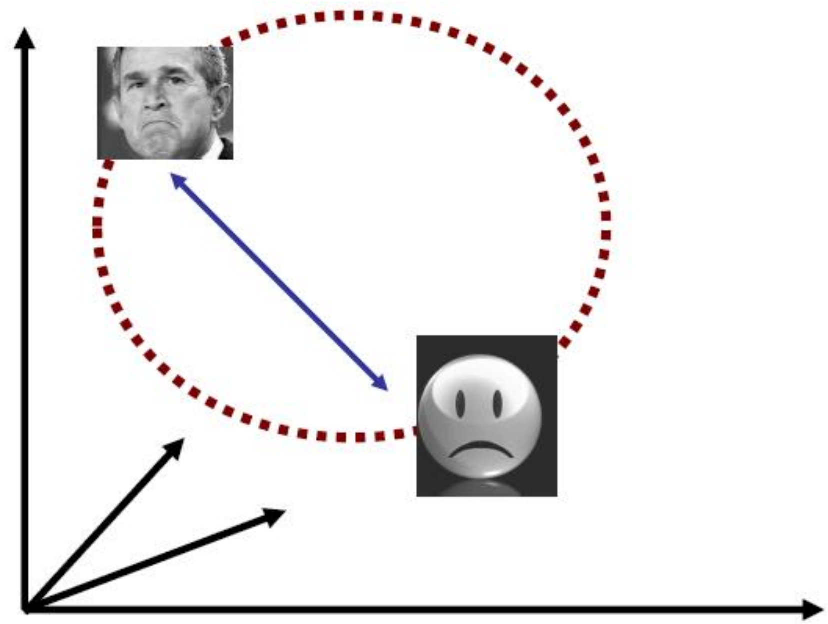
EIGENFACE approach
[Turk and Pentland, 1991]

- An approach to extracting the information contained in a face image is to somehow capture the direction of variation in a collection of face images, and exploit this to encode and compare individual face images.
- We project face images onto a feature space that spans the significant variations among face images.
- The significant features are known as eigenfaces: they are the eigenvectors (or principal components) of the set of faces. They do not necessarily correspond to features such as eyes, nose, and mouth.
- Recognition is performed by projecting a new image into the subspace spanned by the eigenfaces ("face space") and then classifying the face by comparing its position in face space with the positions of known individuals.
- An individual face image can be reconstructed by a weighted sum of the eigenface features.

EIGENFACE approach

– In mathematical terms, we treat an image **x** as a point (or vector) in a very high dimensional space.

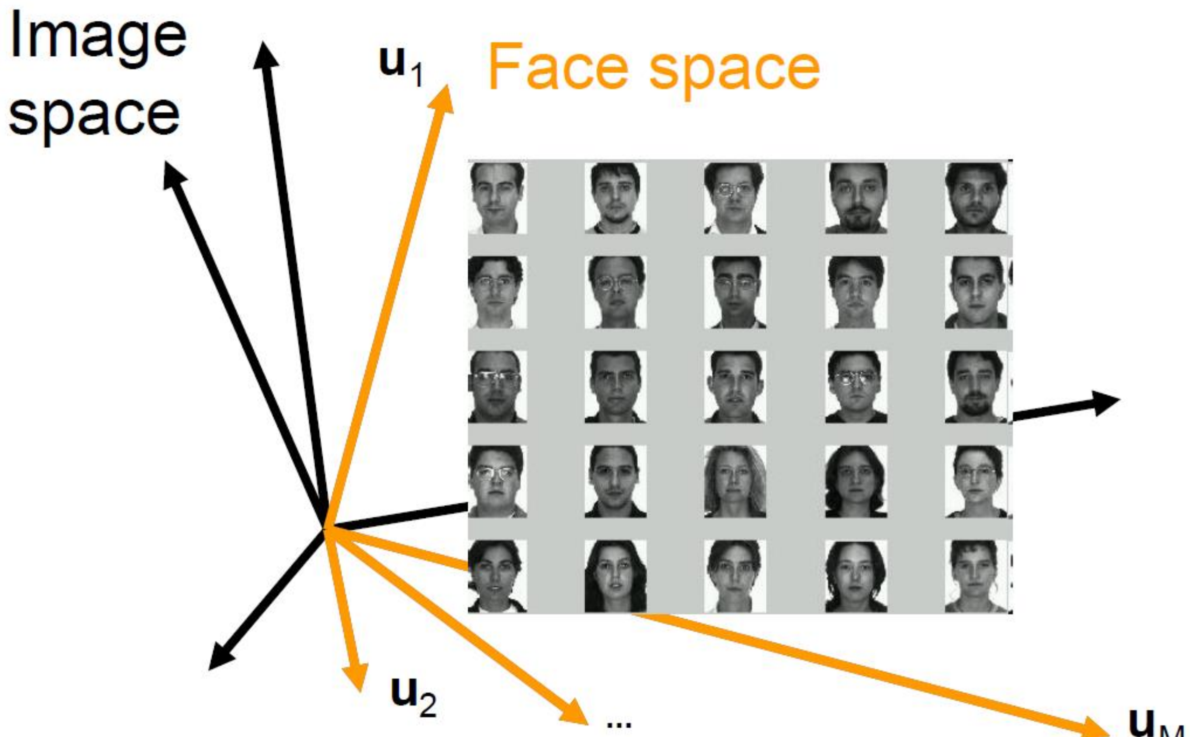




An image **x** is a point in a D-dimensional space.

EIGENFACE approach

- We compute a M-dim subspace such that the projection of the data points onto the subspace has the largest data variance (or the maximum scatter) among all M-dim subspaces.
- The subspace is spanned by the principal components $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M$ of the distribution of faces, or the eigenvectors of the covariance matrix of the set of face images.

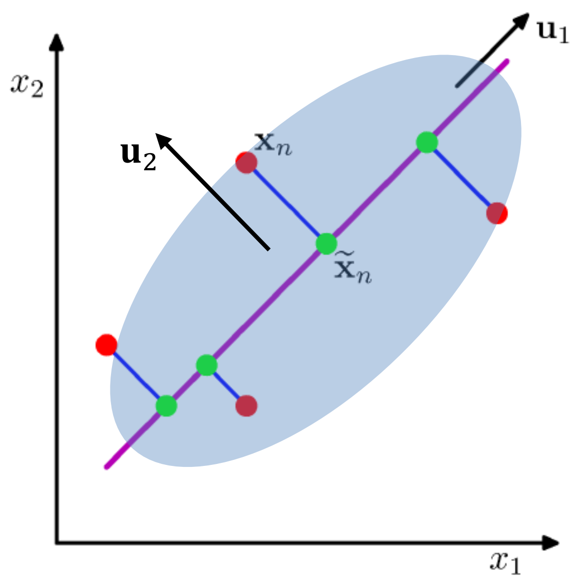


PCA for estimating the subspace

Given a data set $\{\mathbf{x}_n\}$, n = 1,...,N and $\mathbf{x}_n \in R^D$, our goal is to project the data onto a space of dimension $M << D$ while maximising the projected data variance.

For simplicity, M = 1. The direction of this space is defined by a vector $\mathbf{u}_1 \in R^D$. Each data point $\mathbf{x}_n$ is then projected onto $\tilde{\mathbf{x}}_n$.

The M dimensional subspace is spanned by the M eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M$ of the data covariance matrix $\mathbf{S}$ corresponding to the M largest eigenvalues $\lambda_1, \lambda_2 \ldots, \lambda_M$.



EIGENFACE approach

- These eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M$ can be thought of as a set of features that characterize the variation among face images.
- The eigenvectors are ordered by their eigenvalues, with each one accounting for a different kind (direction) and amount of the variation.
- The eigenvector can be visualised as an image. It looks like a ghostly face which we call an eigenface.
- Each eigenface deviates from uniform gray where some facial feature differs among the set of training faces; they are the map of the face image variations.

Example of 20 eigenfaces

Projection onto the Eigenfaces

– A face image **x** is converted to eigenface coordinates (projection coefficients, or features) by

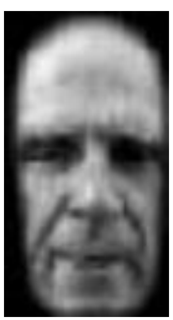
$$\mathbf{x} \rightarrow \left(\underbrace{(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{u}_1}_{a_1}, \underbrace{(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{u}_2}_{a_2}, \ldots, \underbrace{(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{u}_M}_{a_M} \right)$$

where $\bar{\mathbf{x}}$ is the average face.

– So to recognize a particular face, we compare these coordinates to those of known individuals.

– Each face can be represented (or reconstructed) in a linear combination of the eigenfaces.

$$\mathbf{x} \approx \bar{\mathbf{x}} + a_1\mathbf{u}_1 + a_2\mathbf{u}_2 + \ldots + a_M\mathbf{u}_M$$



$\mathbf{x}$

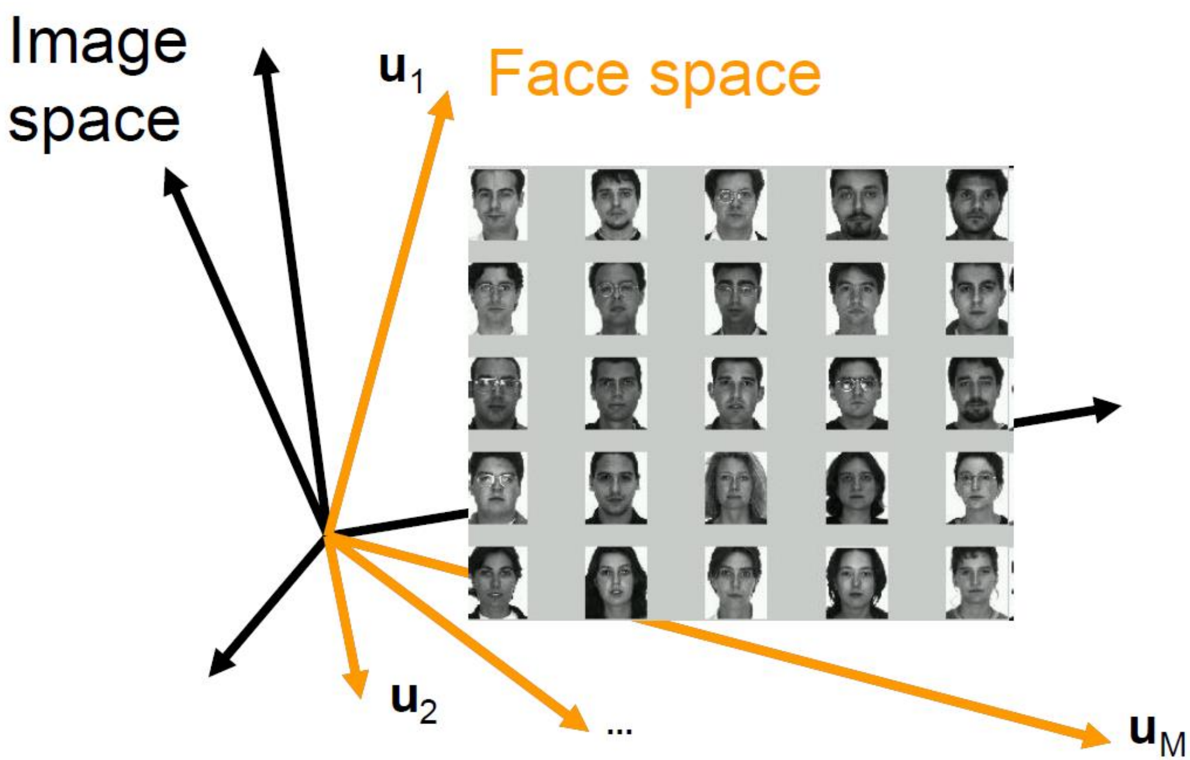
$a_1\mathbf{u}_1$ $a_2\mathbf{u}_2$ $a_3\mathbf{u}_3$ $a_4\mathbf{u}_4$ $a_5\mathbf{u}_5$ $a_6\mathbf{u}_6$ $a_7\mathbf{u}_7$ $a_8\mathbf{u}_8$

$\tilde{\mathbf{x}}$

EIGENFACE approach

– Each face can be approximated using only the "best" eigenfaces that have the largest eigenvalues, accounting for the most data variance in the set of face images.

– The best M eigenfaces $\mathbf{u}_1$, $\mathbf{u}_2$, … , $\mathbf{u}_M$ span the M-dimensional subspace called face space of all possible images.



Procedures

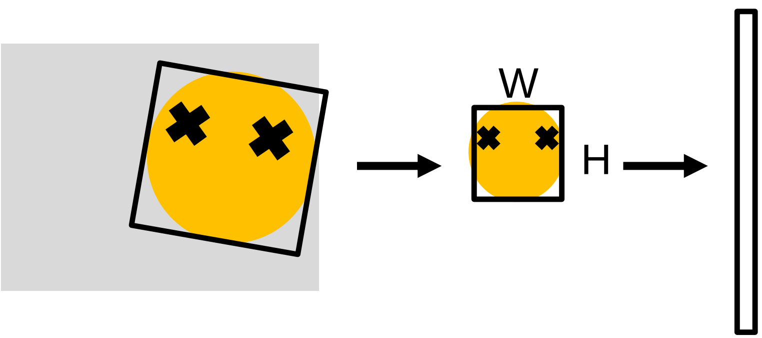
The approach to face recognition involves the following steps for training:

– Step 1: obtain face images $\mathbf{I}_1, \mathbf{I}_2, \ldots, \mathbf{I}_N$ (training faces), where N is the number of images.

(**important:** the face images must be *centered* and of the same *size*)

- Normalize for scale, orientation, translation (e.g. using eye locations)

– Step 2: represent every image $\mathbf{I}_n$ as a vector $\mathbf{x}_n$.



$\mathbf{x} \in R^{D\times 1}$

where
D=WH

Procedures

- Step 3: compute the average face vector $\overline{\mathbf{x}}$

$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n.$$

- Step 4: subtract the mean face:

$$\phi_n = \mathbf{x}_n - \overline{\mathbf{x}}$$

- Step 5: compute the covariance matrix S

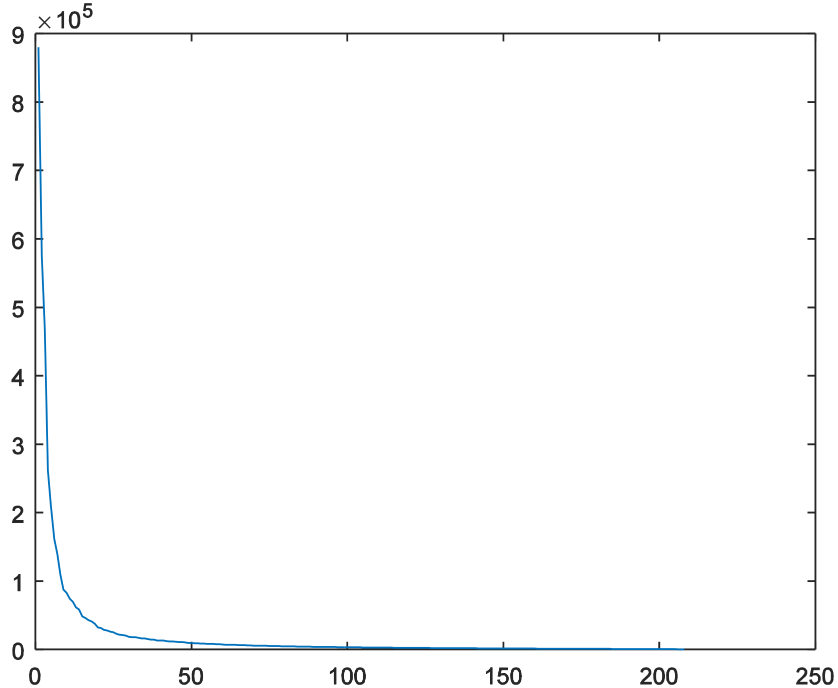
$$\mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \overline{\mathbf{x}})(\mathbf{x}_n - \overline{\mathbf{x}})^T = \frac{1}{N}\sum_{n=1}^{N}\phi_n\phi_n^T = \frac{1}{N}AA^T \in R^{D\times D}$$

$$A = [\phi_1, \phi_2, \ldots, \phi_N] \in R^{D\times N}$$

Procedures

– Step 6: compute the eigenvectors of $S=(1/N)AA^T$: $\mathbf{S}\mathbf{u}_i = \lambda_i \mathbf{u}_i$
  (i=1,…,D, if S is the full-rank matrix).

– Step 7: keep the best M eigenvectors, $\mathbf{u}_1$, $\mathbf{u}_2$, … , $\mathbf{u}_M$ (corresponding to the M largest eigenvalues).



Top 50 eigenfaces

Procedures

Representing faces onto eigenfaces

- Step 8: each normalized training face ϕ_n is represented by its projections:

$$\omega_n = [a_{n1}, a_{n2}, \ldots, a_{nM}]^T$$

where $a_{ni} = \phi_n^T \mathbf{u}_i$, i =1,…,M

- Step 9 (optional): each face in the training set can be reconstructed as a linear combination of the best M eigenvectors:

$$\widetilde{\mathbf{x}}_n = \overline{\mathbf{x}} + \sum_{i=1}^{M} a_{ni} \mathbf{u}_i$$

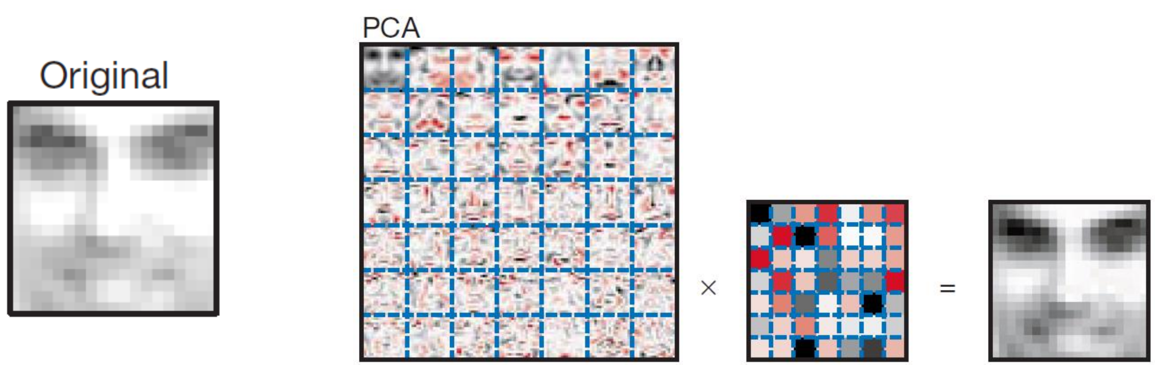


Illustration of face reconstruction

Procedures

Reconstruction of faces using eigenfaces

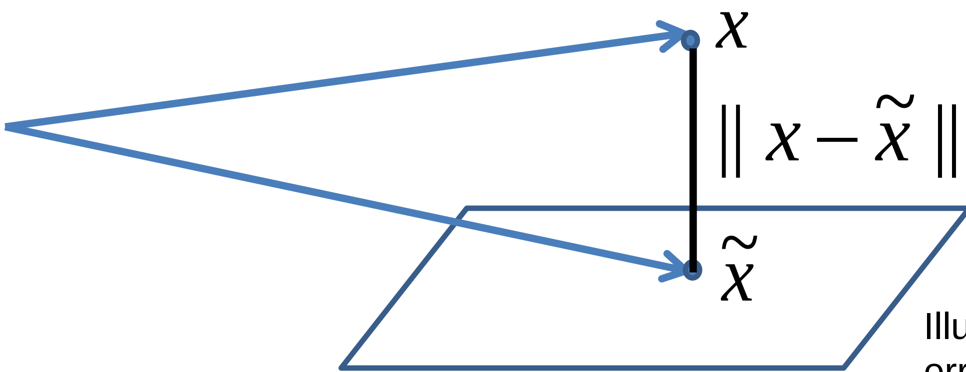


Illustration of the reconstruction error

Original face

Reconstructed face when M=10

M=50

M=100

M=200

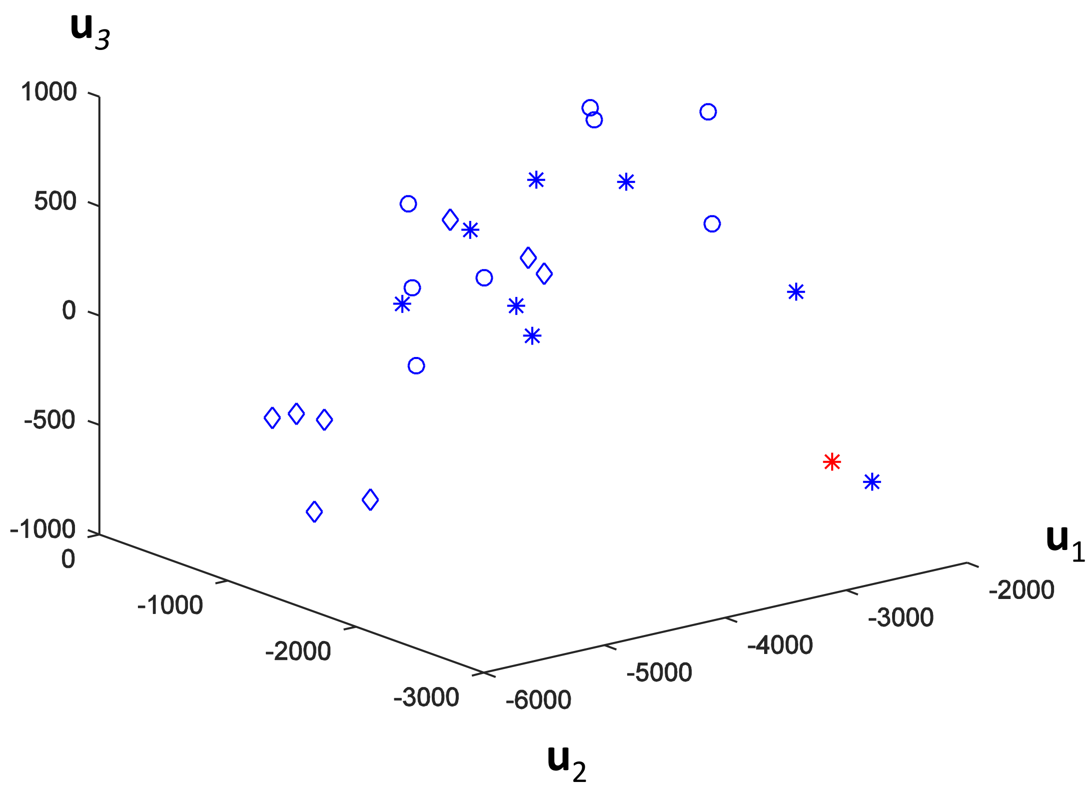
Procedures

(Optional) The training steps can also be performed from time to time whenever we have additional face images and free excess computational capacity. Add new images to the training data set and update the eigenfaces and projections.

Having the learnt model, the following (testing) steps are then applied to recognize a new (unknown) face image **x** (centered and of the same size like the training faces):

- Step 1: normalize **x**: $\phi = \mathbf{x} - \overline{\mathbf{x}}$
- Step 2: project on the eigenspace: $a_i = \phi^T \mathbf{u}_i$, i =1,…,M
- Step 3: represent the projection as: $\omega = [a_1, a_2, ..., a_M]^T$
- Step 4: find $e = \min_n ||\omega - \omega_n||, n = 1, ..., N$
- Step 5: then **x** is recognized as the identify of face *n* from the training set.

NN (Nearest Neighbour) classification



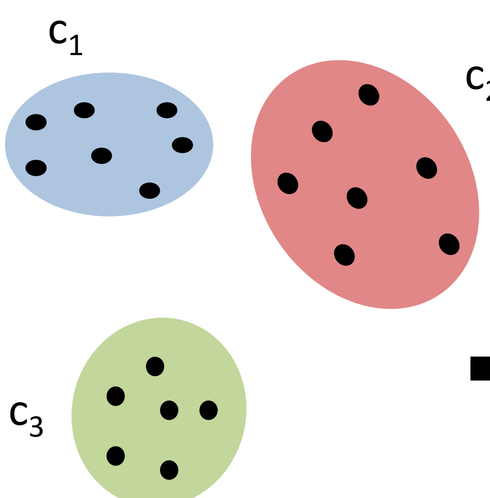
Face images in 3-dimensional eigen-subspace

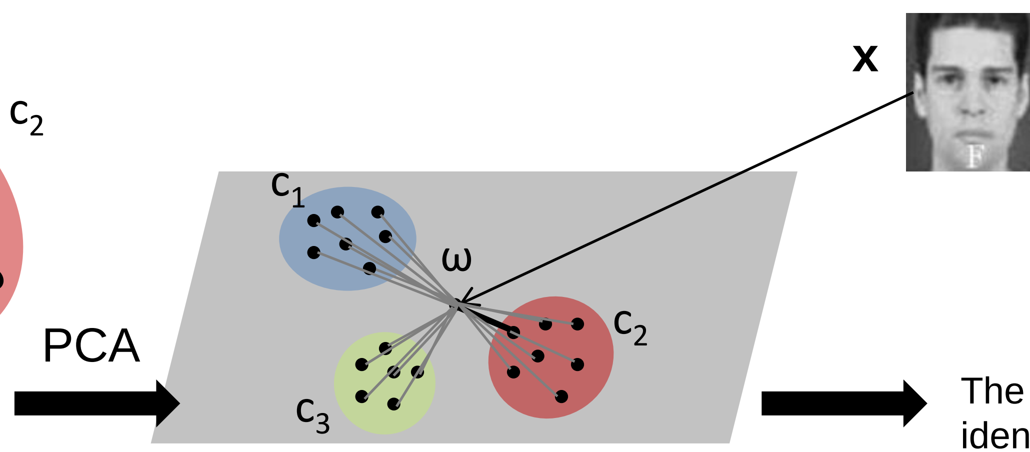
24 training images of 3 different face classes (star, diamond, circle, “in blue”) are projected. A query image projection is “in red”.

Procedures

NN (Nearest Neighbor) classification

- Given face images of different classes (i.e. identities) c_i , compute the principal (eigen) subspace over all data.
- A query (test) image **x** is projected on the eigen-subspace and its projection ω is compared with the projections of all training images.
- The class that has the minimum error is assigned.





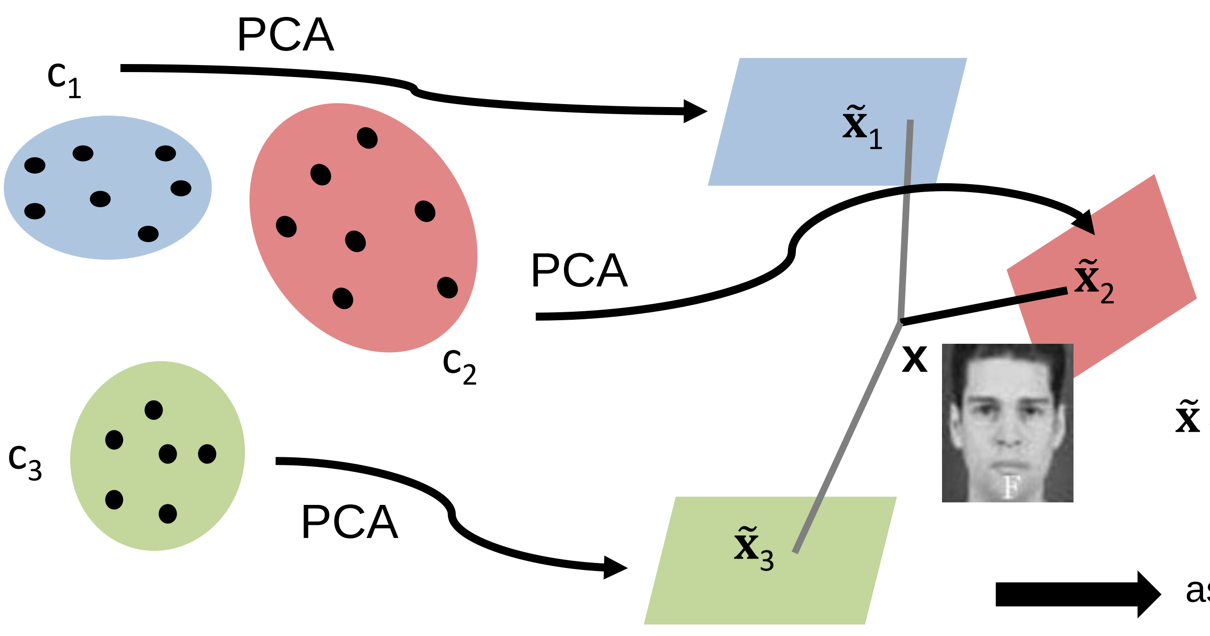
$$e = \min_n ||\omega - \omega_n||, n = 1, \ldots, N$$

The class (or identify) of face n is assigned to the new image.

Procedures

Alternative method

- Given face images of different classes (i.e. identities) c_i, compute the principal (eigen) subspace per class.
- A query (test) image **x** is projected on each eigen-subspace and its reconstruction error is measured.
- The class that has the minimum error is assigned.



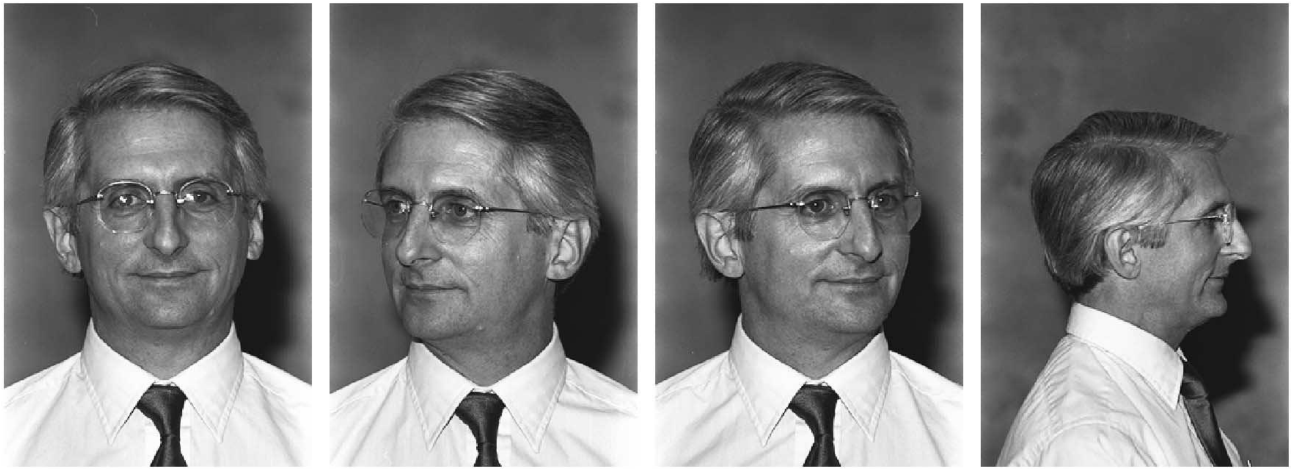
$\tilde{\mathbf{x}}_i$: reconstruction by i-th class subspace

assign $\arg_i \min ||\mathbf{x} - \tilde{\mathbf{x}}_i||$

Multiview face recognition

- Face recognition problem here is treated as an intrinsically two-dimensional (2D) recognition problem rather than requiring recovery of three-dimensional geometrical shape: faces are described by a set of 2D characteristic views.
- Recognition under widely varying conditions is achieved by training on a limited number of characteristic views (e.g., a frontal view, a 45° view, and a profile view).
- The advantages over other face recognition schemes are in: its speed and simplicity, learning capacity, and insensitivity to small or gradual changes in face images.

Images from the FERET dataset



Some references

NIPS 2018

Differentially Private Robust PCA, Raman Arora · Vladimir braverman · Jalaj Upadhyay

The Price of Fair PCA: One Extra dimension, Samira Samadi · Uthaipon Tantipongpipat · Jamie Morgenstern · Mohit Singh · Santosh Vempala

Sparse PCA from Sparse Linear Regression, Guy Bresler · Sung Min Park · Madalina Persu

PCA of high dimensional stochastic processes, Joseph Antognini · Jascha Sohl-Dickstein

Streaming~Kernel~PCA~with~~O(√n)O~(n)***~Random~Features,*** Md Enayat Ullah · Poorya Mianjy · Teodor Vanislavov Marinov · Raman Arora

ICML 2018

Stochastic PCA with ℓ2 and ℓ1 Regularization, Poorya Mianjy · Raman Arora

Streaming Principal Component Analysis in Noisy Setting, Teodor Vanislavov Marinov · Poorya Mianjy · Raman Arora

Nearly Optimal Robust Subspace Tracking, Praneeth Narayanamurthy · Namrata Vaswani

Subspace Embedding and Linear Regression with Orlicz Norm, Alexandr Andoni · Chengyu Lin · Ying Sheng · Peilin Zhong · Ruiqi Zhong

Gradient-Based Meta-Learning with Learned Layerwise Metric and Subspace, Yoonho Lee · Seungjin Choi

Matlab Demos
Face Recognition by PCA

- Face Images
- Eigenvectors and Eigenvalue plot
- Face image reconstruction
- Projection coefficients (visualisation of high-dimensional data)
- Face recognition (NN classification)