パターン認識
Pattern Recognition

佐藤真一
Shin'ichi Satoh

国立情報学研究所
National Institute of Informatics

June 13, 2023

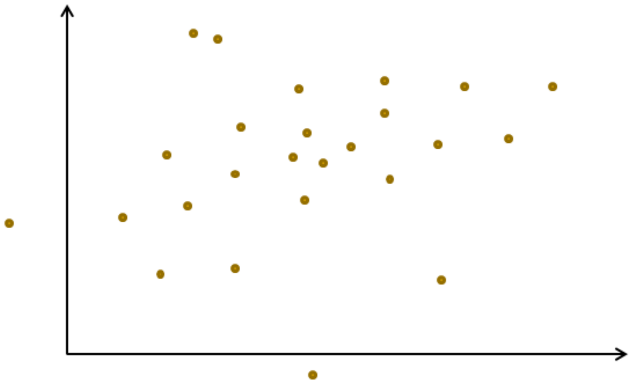
Final Report

- Find any PR paper in top journals or top conferences
- e.g., IEEE TPAMI, IJCV, CVPR, ICCV, NeurIPS, ICML, ACMMM...
- Describe the following:
  - bibliographic info of the paper
  - brief of the paper
  - what is the problem, why it's important, how it's solved, validation?
  - why you selected the paper, what is exciting?
  - feedback to the lecture, any comments
- 2-4 pages A4
- due: 07/31/2023
- send via ITC-LMS

Today's topics

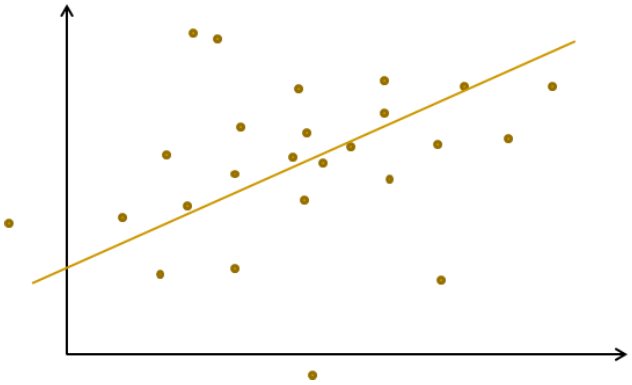
- Example applications of linear algebra in pattern recognition
- Face detection and recognition
  - Principal Component Analysis (PCA) and Eigenface method
  - Linear Discriminant Analysis (LDA) and Fisherface method

Eigenface: Introduction



- Assume that we have points scattered in the vector space.
- Principal Component Analysis (PCA) is a powerful tool to obtain linear manifold which "best" fits with the scatter
- How we compute? What can be possible applications?

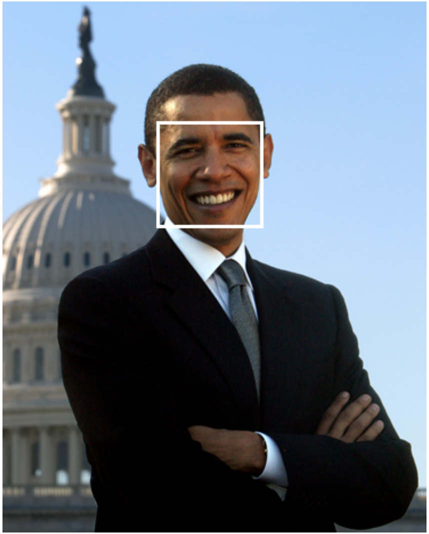
Eigenface: Introduction



- Assume that we have points scattered in the vector space.
- Principal Component Analysis (PCA) is a powerful tool to obtain linear manifold which "best" fits with the scatter
- How we compute? What can be possible applications?

Face detection and recognition

- Assume that we have a set of face images (with identity)
- Face detection: decide if given unknown image is face or not
- Face recognition: decide the identity of given face image



Images as intensity data

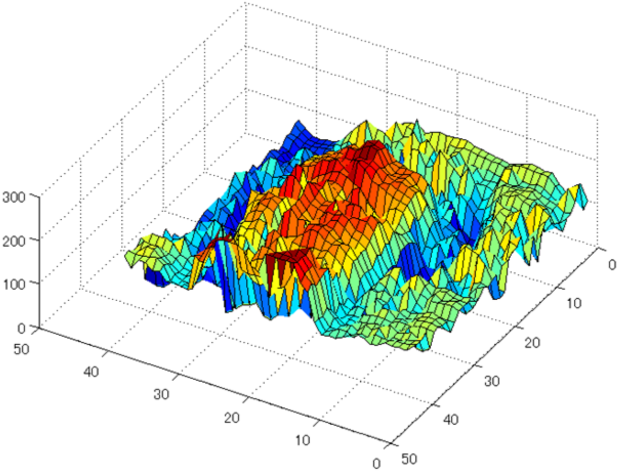


Image transformation


original


gray scale

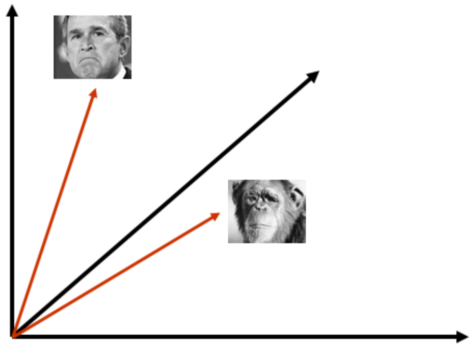

crop


discretize


quantize

$$\mathbf{x} = [x_1\ x_2\ x_3\ \cdots\ x_n]$$

The Space of Faces

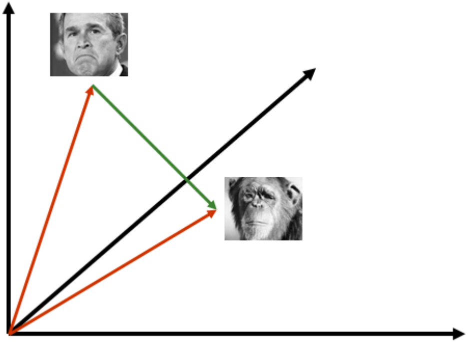


An image is a point in a high dimensional space

- An $N \times M$ image is a point in $\mathbb{R}^{NM}$
- We can define vectors in this space as we did in the 2D case

[Thanks to Chuck Dyer, Steve Seitz, Nishino]

The Space of Faces

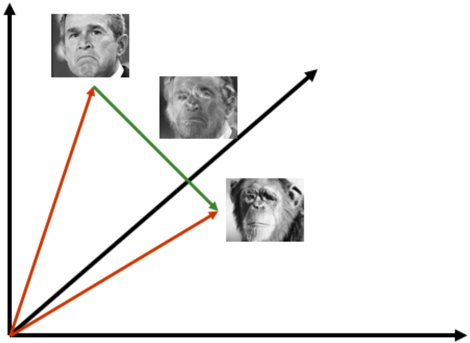


An image is a point in a high dimensional space

- An $N \times M$ image is a point in $\mathbb{R}^{NM}$
- We can define vectors in this space as we did in the 2D case

[Thanks to Chuck Dyer, Steve Seitz, Nishino]

The Space of Faces

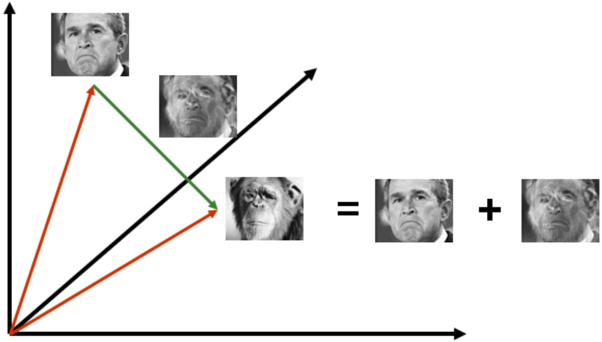


An image is a point in a high dimensional space

- An $N \times M$ image is a point in $\mathbb{R}^{NM}$
- We can define vectors in this space as we did in the 2D case

[Thanks to Chuck Dyer, Steve Seitz, Nishino]

The Space of Faces



An image is a point in a high dimensional space

- An $N \times M$ image is a point in $\mathbb{R}^{NM}$
- We can define vectors in this space as we did in the 2D case

[Thanks to Chuck Dyer, Steve Seitz, Nishino]

Multivariate normal distribution

- We typically assume normal distribution
- Let's assume $p(x|\text{non-face})$ yields uniform distribution, and $p(x|\text{face})$ yields normal distribution
- multivariate normal distribution

$$N_x(M, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} exp(-\frac{1}{2}(X - M)^T\Sigma^{-1}(X - M))$$

Normal distribution

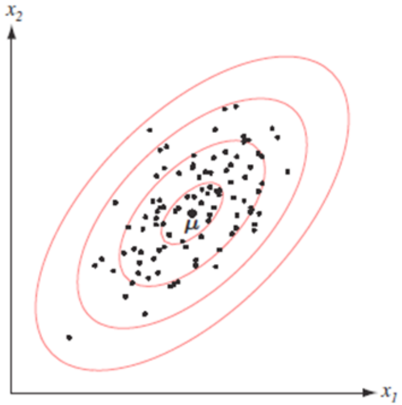


FIGURE 2.9. Samples drawn from a two-dimensional Gaussian lie in a cloud centered on the mean $\boldsymbol{\mu}$. The ellipses show lines of equal probability density of the Gaussian. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Eigenface method

- Key idea: the distribution of face images in the image space yields low dimensional linear manifold
- find principal components by principal component analysis (PCA) via eigenvalue decomposition
- project unknown images onto the manifold spanned by the obtained principal components
- face detection: decide face if it's close enough to the manifold
- face recognition: decide the identity of the closest face image within the mainfold

Eigenface method

- Assume we have samples of faces:

$$X = [x_1 \, x_2 \, x_3 \, \cdots \, x_n]$$

- We can then obtain covariance matrix:

$$\mu = E[\mathbf{x}] = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\Sigma = E[(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T] = \frac{1}{n}(X-\mu)(X-\mu)^T$$

Eigenface method

- We then apply eigenvalue decomposition to the covariance matrix

$$\Sigma\phi_i = \lambda_i\phi_i$$

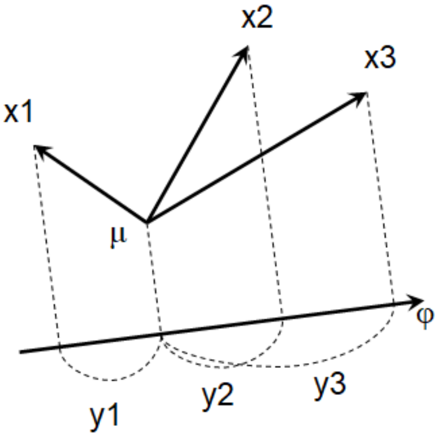
- where λ_i are eigenvalues and ϕ_i are eigenvectors
- Retain m eigenvectors (ϕ_i , $i = 1, \cdots, m$) corresponding to the m largest eigenvalues
- Then each image x can be converted into m-dimensional vector

$$\Phi = [\phi_1 \, \phi_2 \, \cdots \, \phi_m]$$
$$x' = \Phi^T(x - \mu)$$

Covariance matrix and its algebraic/geometric interpretation

What is the quadratic form $\phi^T \Sigma \phi$?

$$
\begin{aligned}
\phi^T \Sigma \phi =& \phi^T \left[\frac{1}{n} \sum_i (x_i - \mu)(x_i - \mu)^T \right] \phi \\
=& \frac{1}{n} \sum_i \left[\phi^T (x_i - \mu)(x_i - \mu)^T \phi \right] \\
=& \frac{1}{n} \sum_i \left[\phi^T (x_i - \mu) \right] \left[(x_i - \mu)^T \phi \right] \\
=& \frac{1}{n} \sum_i \left[\phi^T (x_i - \mu) \right] \left[\phi^T (x_i - \mu) \right]^T \\
=& \frac{1}{n} \sum_i \left[\phi^T (x_i - \mu) \right]^2 \\
=& \frac{1}{n} y_i^2
\end{aligned}
$$

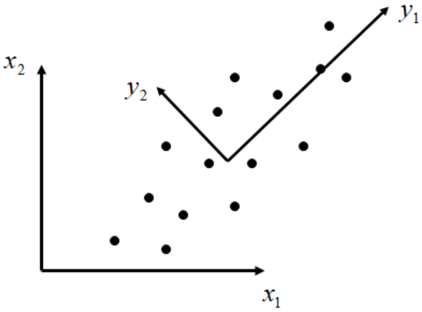


Covariance matrix and its algebraic/geometric interpretation

How to maximize $\phi^T \Sigma \phi$ wrt $\phi^T \phi = 1$?

$$
\begin{aligned}
J =& \phi^T \Sigma \phi - \lambda(\phi^T \phi - 1) \quad \text{Lagrange multipliers method} \\
\frac{\partial J}{\partial \phi} =& \cdots \\
=& 2\Sigma\phi - 2\lambda\phi = 0 \\
\Sigma\phi =& \lambda\phi
\end{aligned}
$$

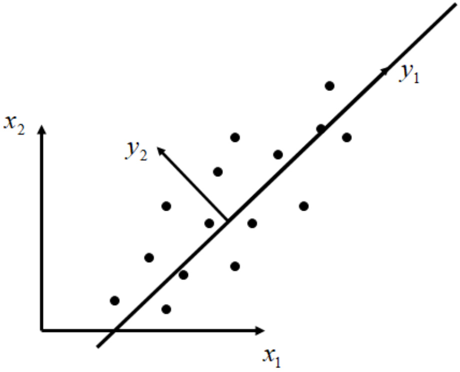
Face detection and recognition by Eigenface method



$$
\begin{aligned}
\Sigma &= E\mathbf{x}\mathbf{x}^T \\
\Sigma\phi_i &= \lambda_i\phi_i \\
&\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \\
\Phi &= [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n \\
y &= \Phi^T x \\
e &= x - \Phi y
\end{aligned}
$$

PCA is an orthonormal projection of a random vector **x** onto a lower-dimensional subspace Y that minimizes mean square error.

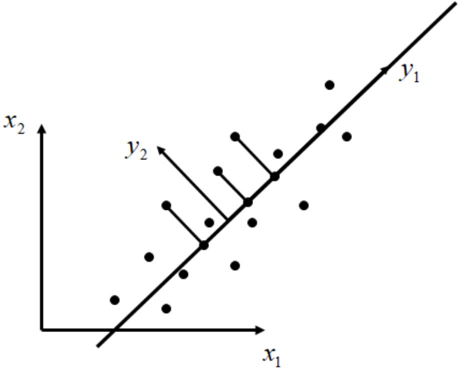
Face detection and recognition by Eigenface method



$$\Sigma = E\mathbf{x}\mathbf{x}^T$$
$$\Sigma\phi_i = \lambda_i\phi_i$$
$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$$
$$\Phi = [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n$$
$$y = \Phi^T x$$
$$e = x - \Phi y$$

PCA is an orthonormal projection of a random vector $\mathbf{x}$ onto a lower-dimensional subspace Y that minimizes mean square error.

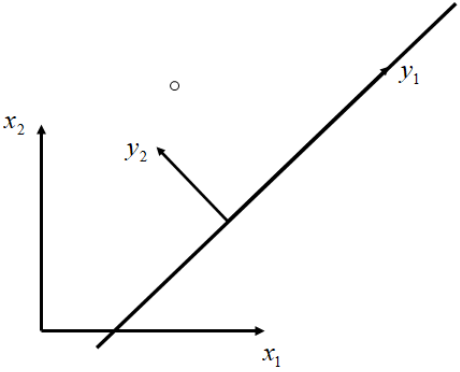
Face detection and recognition by Eigenface method



$$\Sigma = E\mathbf{x}\mathbf{x}^T$$
$$\Sigma\phi_i = \lambda_i\phi_i$$
$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$$
$$\Phi = [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n$$
$$y = \Phi^T x$$
$$e = x - \Phi y$$

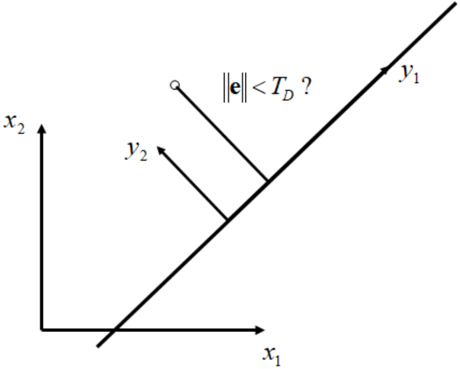
PCA is an orthonormal projection of a random vector **x** onto a lower-dimensional subspace Y that minimizes mean square error.

Face detection and recognition by Eigenface method



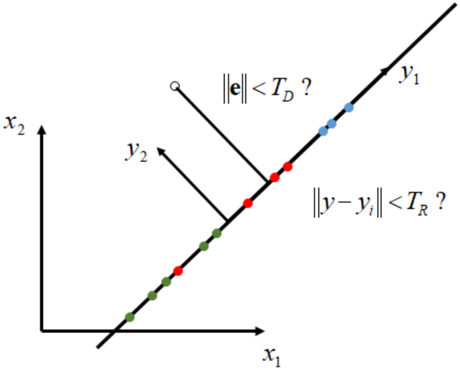
$$\begin{aligned} \Sigma &= E\mathbf{x}\mathbf{x}^T \\ \Sigma\phi_i &= \lambda_i\phi_i \\ & \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \\ \Phi &= [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n \\ y &= \Phi^T x \\ e &= x - \Phi y \end{aligned}$$

Face detection and recognition by Eigenface method



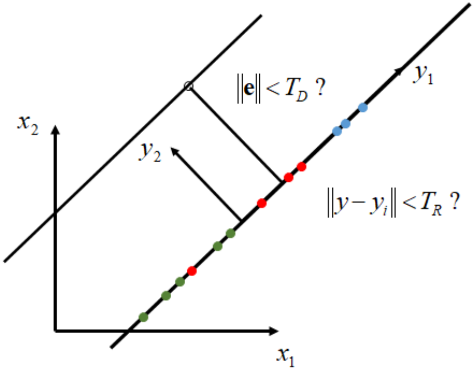
$$
\begin{aligned}
\Sigma &= E\mathbf{x}\mathbf{x}^T \\
\Sigma\phi_i &= \lambda_i\phi_i \\
&\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \\
\Phi &= [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n \\
y &= \Phi^T x \\
e &= x - \Phi y
\end{aligned}
$$

Face detection and recognition by Eigenface method



$$\Sigma = E\mathbf{x}\mathbf{x}^T$$
$$\Sigma\phi_i = \lambda_i\phi_i$$
$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$$
$$\Phi = [\phi_1 \ \phi_2 \ \cdots \ \phi_m] \quad m \ll n$$
$$y = \Phi^T x$$
$$e = x - \Phi y$$

Face detection and recognition by Eigenface method



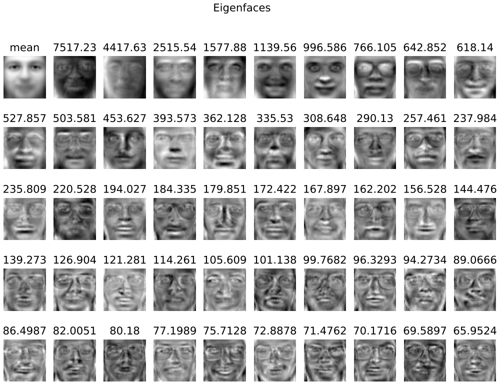
$$\Sigma = E\mathbf{x}\mathbf{x}^T$$
$$\Sigma\phi_i = \lambda_i\phi_i$$
$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$$
$$\Phi = [\phi_1\ \phi_2\ \cdots\ \phi_m] \quad m \ll n$$
$$y = \Phi^T x$$
$$e = x - \Phi y$$

- Detection of faces based on distance from face space
- Recognition of faces based on distance within face space

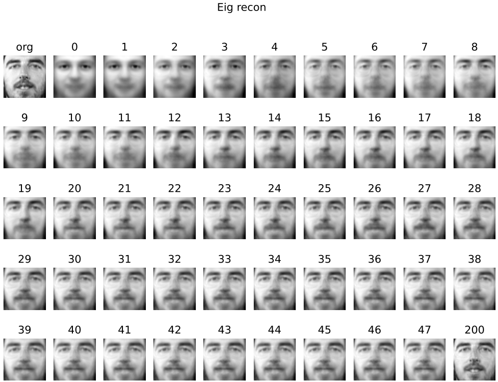
Eigenface

PCA extracts the eigenvectors of Σ

- Gives a set of vectors $\phi_1, \phi_2, \phi_3, \cdots$
- Each one of these vectors is a direction in face space
- what do these look like?



Projecting onto the Eigenfaces



The eigenfaces $\phi_1, \phi_2, \phi_3, \cdots, \phi_m$ span the space of faces

A face is converted to eigenface coordinates by

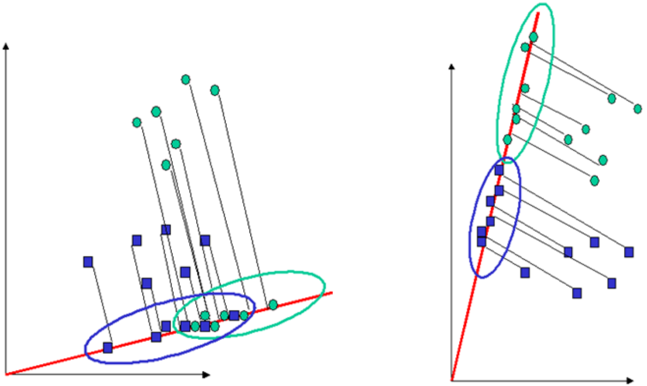
$$x \rightarrow [\underbrace{(x - \bar{x})^T \phi_1}_{y_1} \; \underbrace{(x - \bar{x})^T \phi_2}_{y_2} \cdots \underbrace{(x - \bar{x})^T \phi_m}_{y_m}]$$

$$x \approx \bar{x} + y_1 \phi_1 + y_2 \phi_2 + \cdots + y_m \phi_m$$

Fisherface method

- Eigenface finds principal components which maximize the variance of face distribution
- What happens if we use identity information?
- it would be reasonable idea to maximize the variance between different people, while minimize the variance within the same people
- find such components by Linear Discriminant Analysis (LDA)
- project unknown images onto the space spanned by the obtained components
- face recognition (only): decide the identity of the closest face image within the space

Fisherface method



Poor Projection Good Projection

Fisherface method

- Assume that we have N sample images:

$$\{x_1, x_2, \cdots, x_n\}$$

- Assume that we have C classes (identities):

$$\{\chi_1, \chi_2, \cdots, \chi_C\}$$

- Then average of each class is:

$$\mu_i = \frac{1}{|\chi_i|} \sum_{x \in \chi_i} x$$

- Total average:

$$\mu = \frac{1}{N} \sum_{k=1}^{N} x_k$$

Fisherface method

- Scatter of class i:

$$S_i = \sum_{x \in \chi_i} (x - \mu_i)(x - \mu_i)^T$$

- Within class scatter:

$$S_W = \sum_{i=1}^{C} S_i$$

- Between class scatter:

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

- Total scatter:

$$S_T = S_W + S_B$$

Fisherface method

- We want to obtain linear projection W which converts input x to low-dimensional vector y:

$$y = W^T x$$

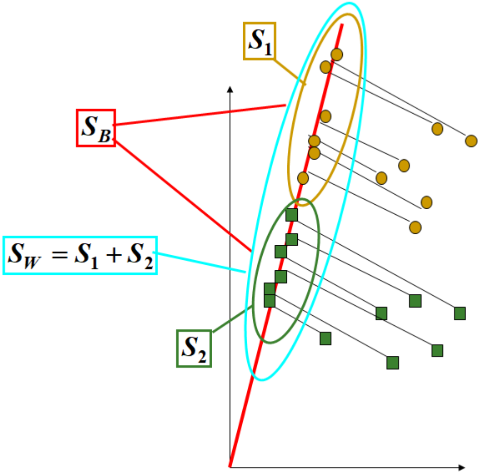
- Between class scatter in the space of y:

$$\tilde{S_B} = W^T S_B W$$

- Within class scatter in the space of y:

$$\tilde{S_W} = W^T S_W W$$

Fisherface method



Good separation

Fisherface method

- The wanted projection:

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

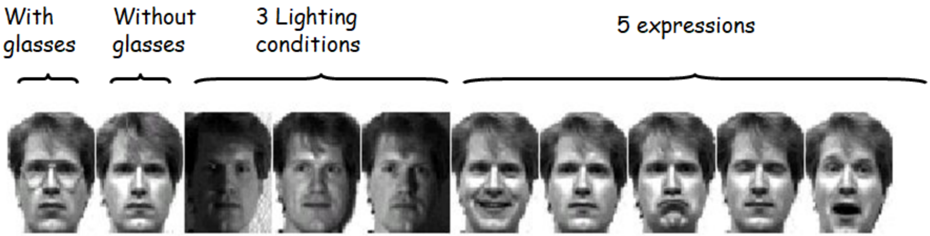
- This can be obtained by generalized Eigen value decomposition

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

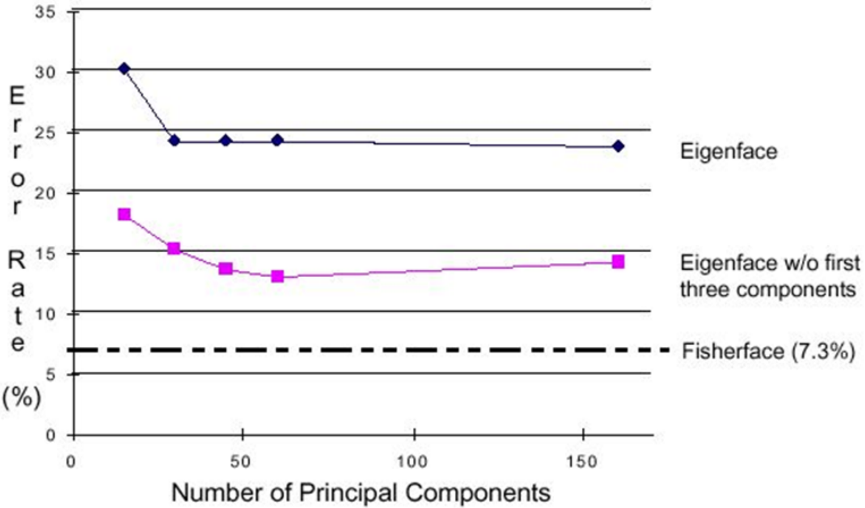
Fisherface method: experiments

Yale dataset having variation in Facial Expression, Eyewear, and Lighting

- input: 160 images of 16 people
- Train: 159, text: 1 (Leave-one-out)



Fisherface method: experiments



Demo

- The experiments of Eigenfaces and Fisherfaces using public face dataset
- Eigenfaces visualization
- Face reconstruction using Eigenfaces
- Head to head comparison of Eigenfaces vs Fisherfaces in face recognition
- Olivetti faces dataset
- 40 individuals 10 images (different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses))