Face Recognition using Eigenfaces and Fisherfaces

MOR OHANA

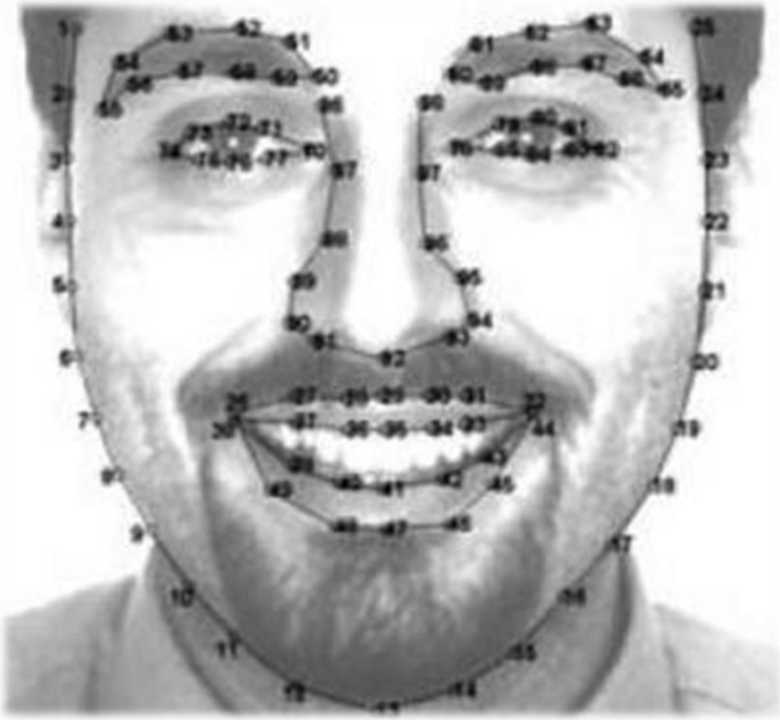
RECOGNITION AND CLASSIFICATION IN IMAGES AND VIDEO

MARCH 2014

Previous approaches

1. Use edge detection to find key points
2. Learn key point placement in faces
3. Upon receiving a new face, use the above to recognize face

- Hard to implement and not deterministic



Eigenface approach

- Project training images into “eigenspace”
- Deduce the top eigenvectors from the projection (called “eigenfaces” because they look like faces)
- Every face Image can be represented as a linear combination of the eigenfaces

PCA – Principal Component Analysis

- Allows us to lower the dimension of a data set while retaining as much of the original data as possible
- For example for X of dimension N:
  - $X = a_1 v_1 + a_2 v_2 \dots + a_N v_N$
  - $v_1, v_2 \dots v_N$ is the dimensional basis
- We can compute $\hat{X}$ of dimension K << N to be:
  - $\hat{X} = b_1 u_1 + b_2 u_2 + \cdots + b_K u_k$
- If $K = N$ then $X = \hat{X}$

PCA – Principal Component Analysis

- Dimensional reduction = Information Loss
- PCA minimizes the error rate of $\|X - \hat{X}\|$
- The best way to do this is to keep the largest eigenvectors of the covariance matrix of X
  - Also called the “Principal Components”

PCA Methodology

For vectors $x_1, x_2 \ldots x_M$ of order $N \times 1$

1. Compute the avg. $\bar{X} = \frac{1}{M}\sum_{i=1}^{M} x_i$
2. Subtract to center at zero $\phi_i = x_i - \bar{x_i}$
3. Form the matrix $A = [\phi_1 \phi_2 \ldots \phi_M]$ of order $N \times M$ then compute:

   $\mathrm{C} = \frac{1}{M}\sum_{i=1}^{M} \phi_i \phi_i^T = AA^T$

   (Covariance matrix of order $N \times N$)

PCA Methodology

4. Eigenvalues of C: $\lambda_1 > \lambda_2 > \cdots > \lambda_N$
5. Eigenvectors of C: $u_1, u_2 \ldots u_N$ form a basis, so as we've seen:

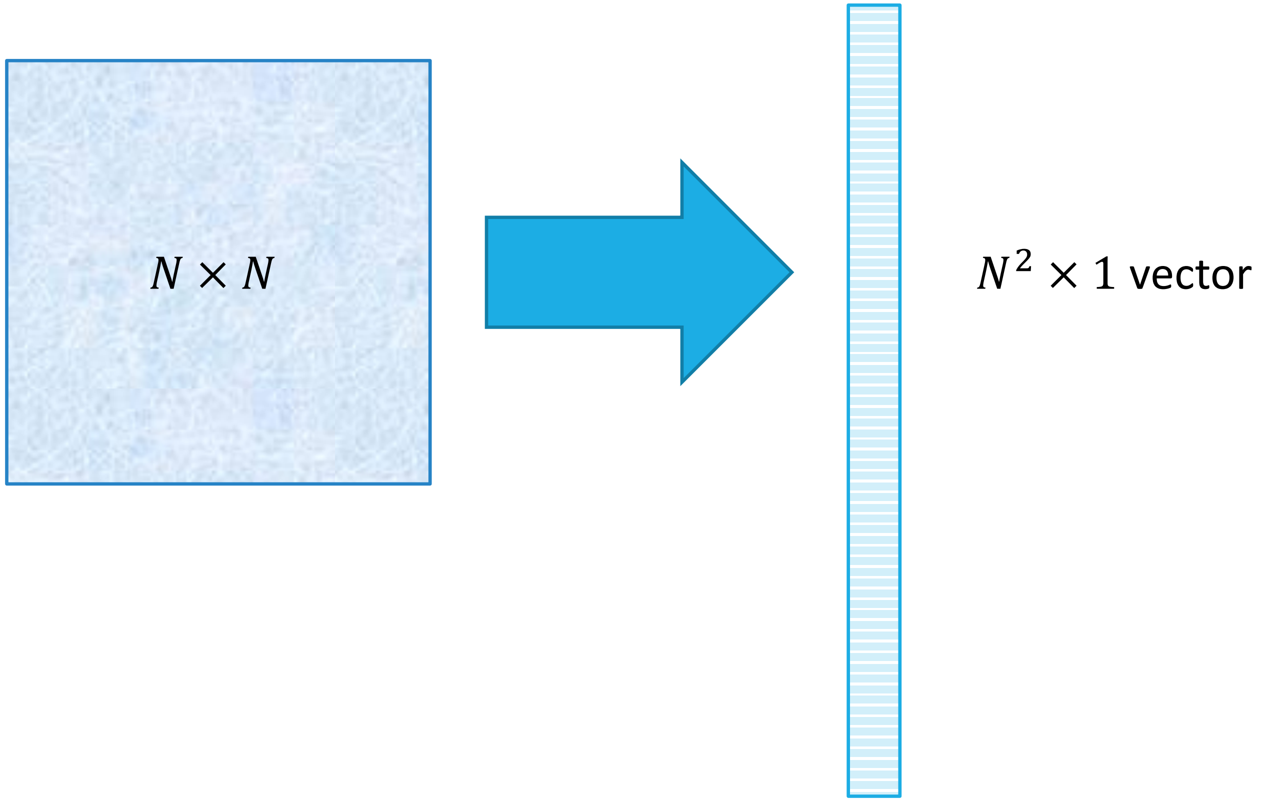
$$\mathrm{x} = b_1 u_1 + b_2 u_2 + \cdots + b_K u_k$$

6. Reduce the dimension from N to K by keeping the K largest eigenvalues: $x \Rightarrow \hat{x}$

So $\hat{x} - \bar{x} = [b_1\ b_2\ \ldots b_K]$

Application to eigenfaces

1. For face Images $I_1, I_2 \dots I_M$ of the same size
2. Convert every image I_i to vector Γ_i



Application to eigenfaces

3. Compute the average face: $\Psi = \frac{1}{M}\sum_{i=1}^{M} \Gamma_i$
4. Subtract : $\Phi_i = \Gamma_i - \Psi$
5. Compute the covariance matrix C:

$$C = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \Phi_i^T = AA^T \ (N^2 \times N^2)$$

$$A = [\Phi_1 \ \Phi_2 \ \ldots \Phi_M] \ (N^2 \times M)$$

The catch

- Now we must compute the eigenvectors u_i of AA^T ⇨ $AA^T u_i = \lambda_i u_i$
  - But that is not practical, as it is very large! $(N^2 \times N^2)$
- However if the number of points in the image space is less than the dimension of the space $(M < N^2)$, there will be only $M - 1$ meaningful eigenvectors
- We can solve for an $M \times M$ matrix $A^T A$ instead of the

  $N^2 \times N^2 \ \ AA^T$

The Trick

- Compute the eigenvectors v_i of $A^T A$
  - $A^T A v_i = \mu_i v_i$
  - Multiply both sides by A and we get:
  - $A A^T A v_i = \mu_i A v_i \Rightarrow$
  - $u_i = A v_i$ and $\lambda_\mathrm{i} = \mu_i$
- Or in other words $A^T A$ and $A A^T$ have the same eigenvalues, and their eigenvectors correspond to $u_i = A v_i$
- With this, calculations are reduced from an order of N^2 to an order of $M \ll N^2$

Visual examples

Training set

Average face Ψ

HTTP://CNX.ORG/CONTENT/M33183/LATEST/?COLLECTION=COL11153/LATEST

Visual examples

Averaging process

Visual examples

Top K eigenvectors

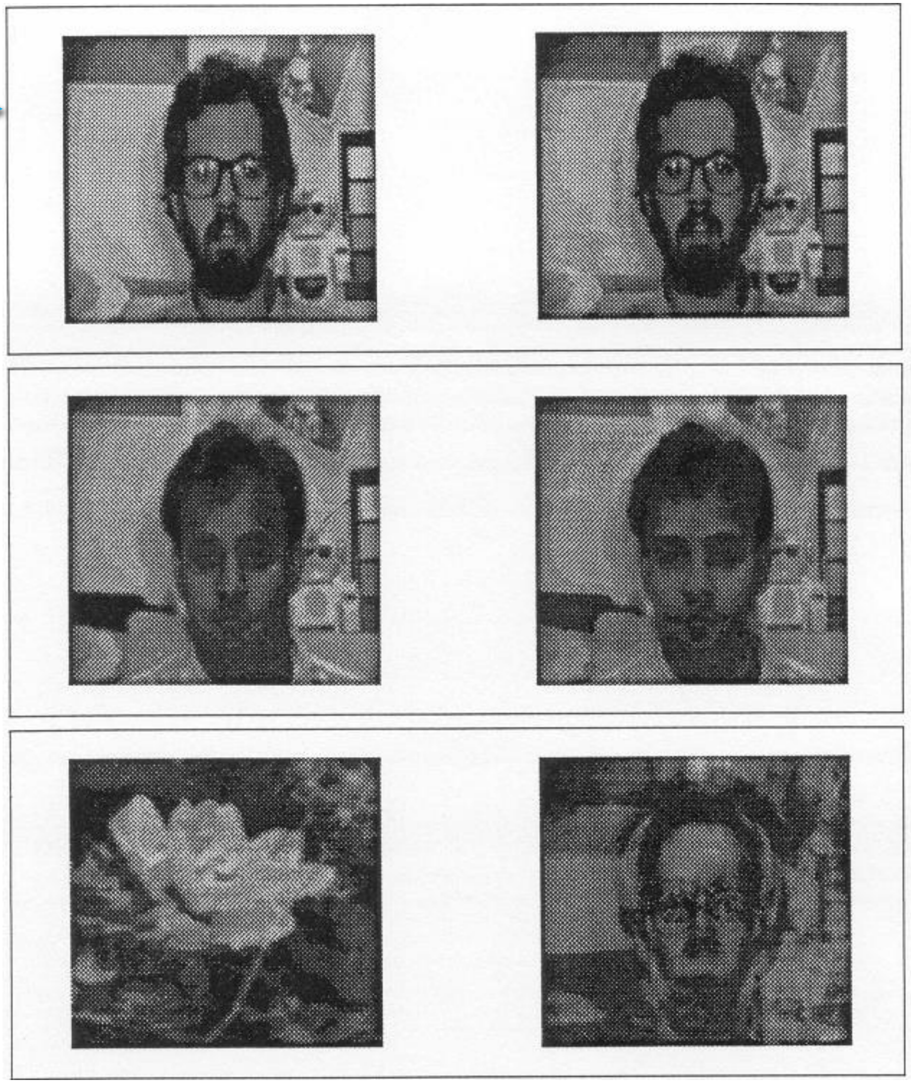
Visual examples

Reconstruct with eigenvectors

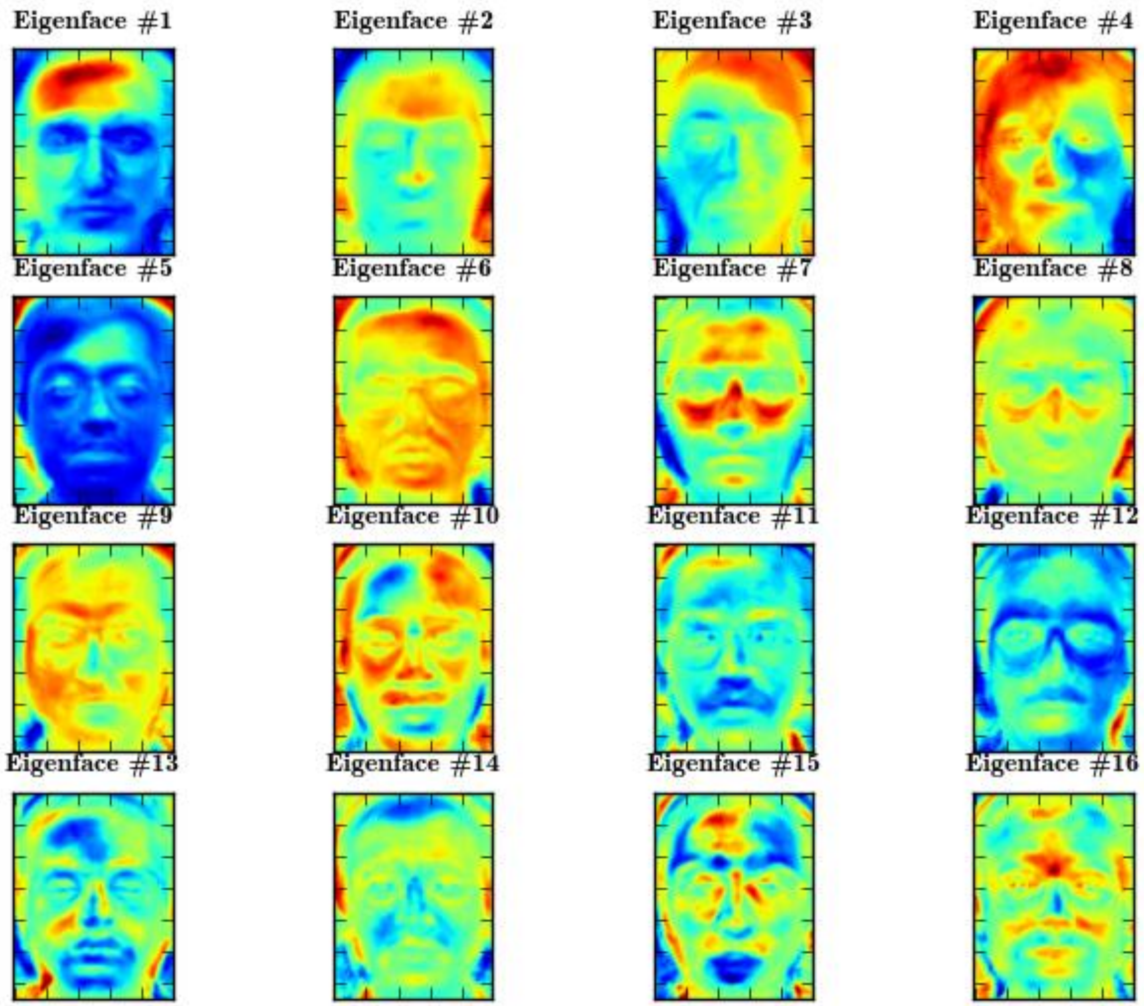
HTTP://JEREMYKUN.COM/2011/07/27/EIGENFACES/

Visual examples

Projections



Visual examples



Eigenfaces
in color!

What more can it do?

- Eigenfaces does more then just recognition
- Detection
  - Basically a template matching problem
  - Problematic at high dimension space
  - Map to lower dimensions first
- Reconstruction
  - Works well!

Problems and Limitations

- Background dependent (changes cause problems).
- Light changes

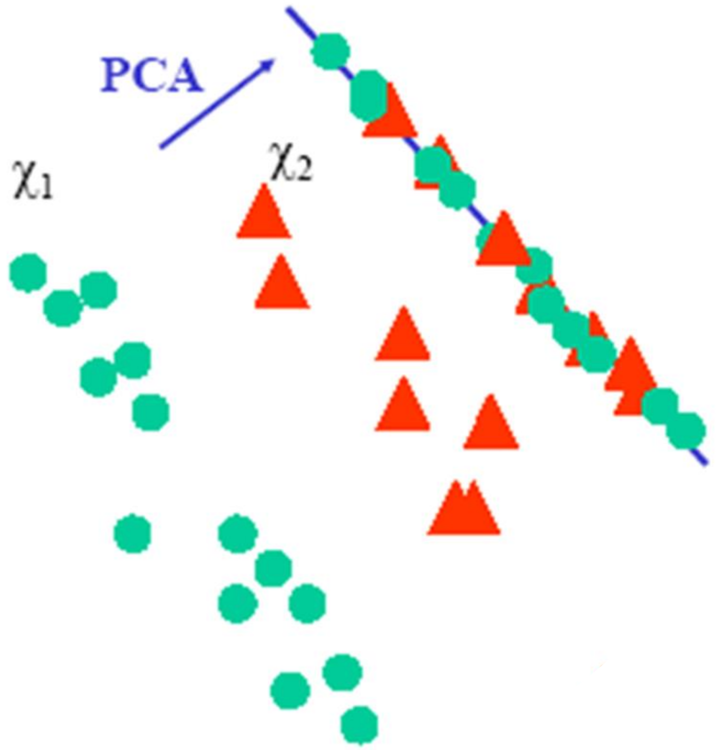
Problems and Limitations

- Face size and orientation
- Changes in expression
- Faces must be centered!
  - Misalignment problems:

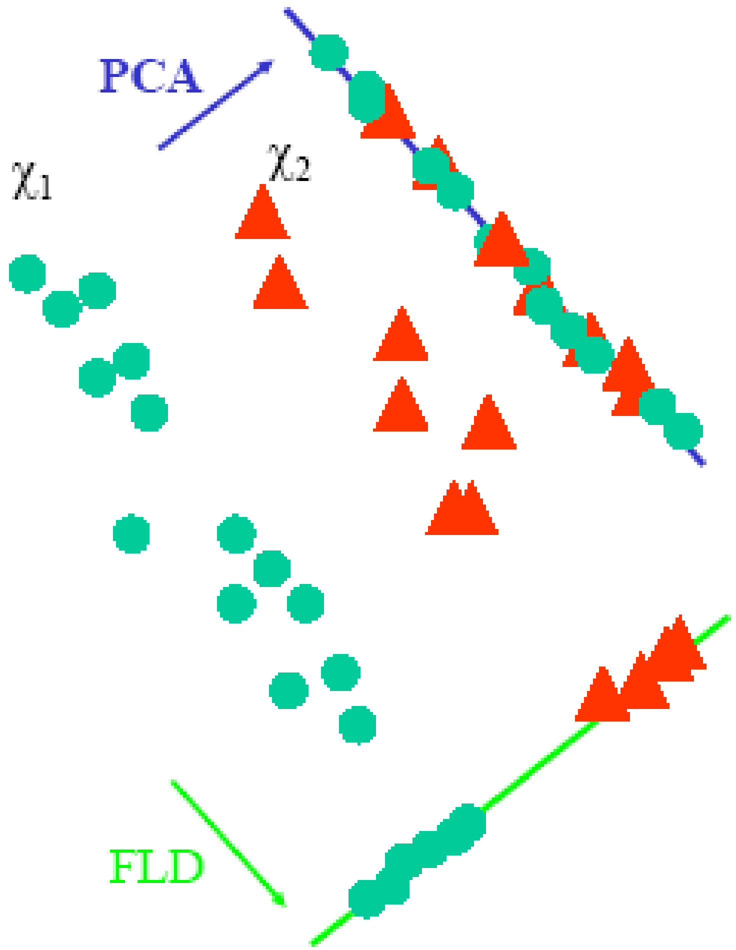
Problems and Limitations

- PCA is not an optimal dimensional reduction for classification purposes:



LDA – Linear Discriminant Analysis

- An enhancement to PCA
- Also called FLD
- Constructs a subspace that:
  - Minimizes the scatter between data points of the same class
  - Maximizes the scatter between data points of different classes

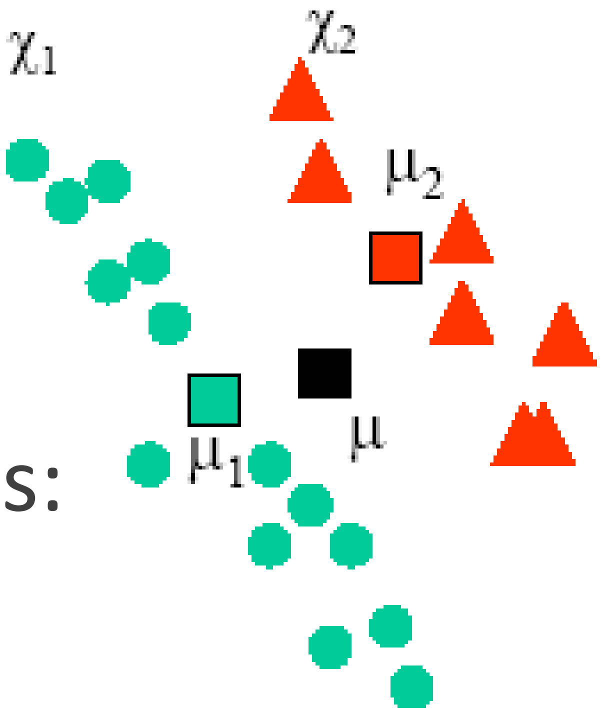


FDA Methodology

- For c face classes $X_1, X_2 \ldots X_c$, each face class has k images $x_1, x_2 \ldots x_k$
- Compute the mean for each class:
  - $\mu_i = \frac{1}{k}\sum_{j=1}^{k} x_j$
- The mean of all classes can be calculated as:
  - $\mu = \frac{1}{c}\sum_{i=1}^{c} \mu_i$

FDA Methodology

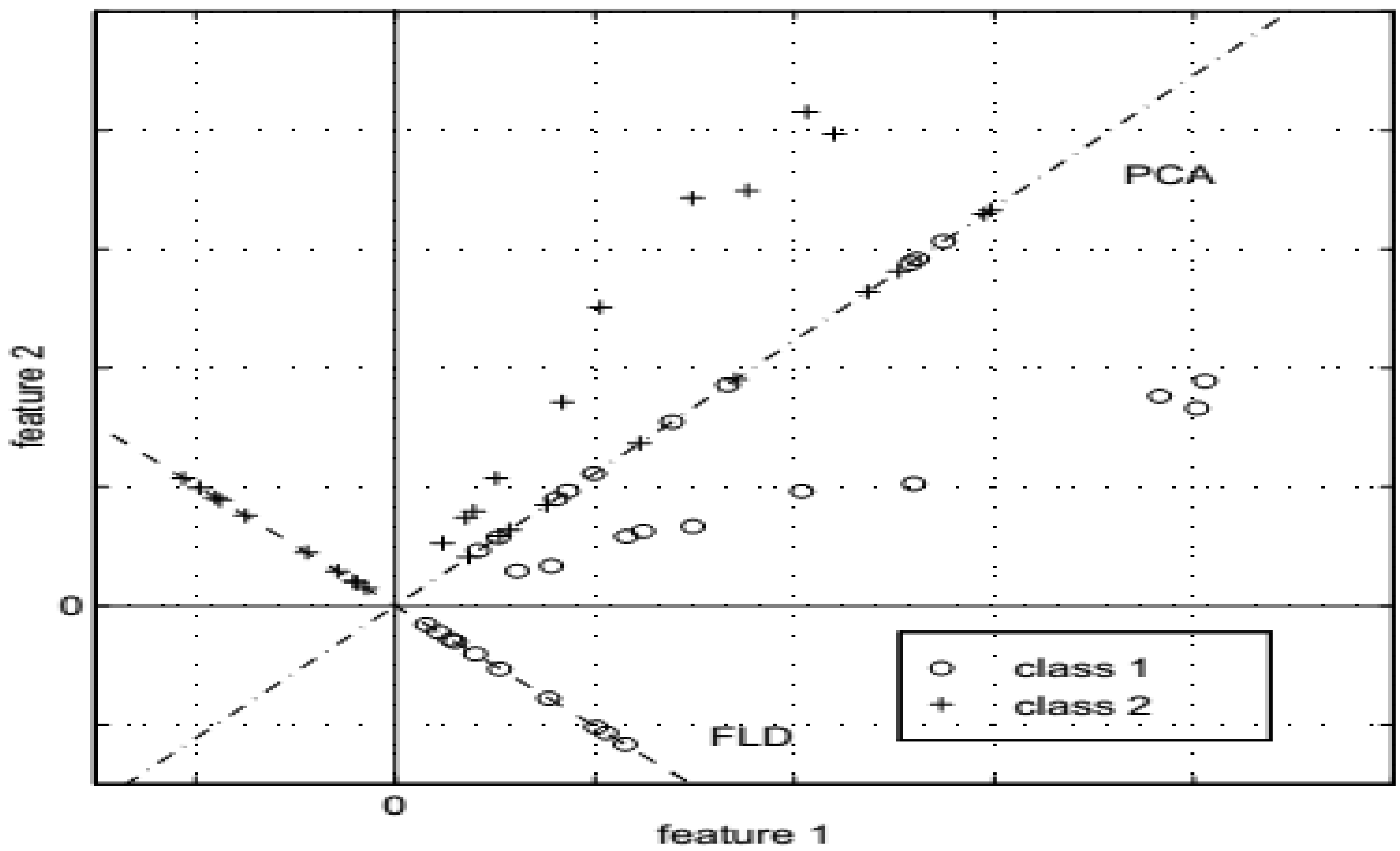
- The scatter within the class is:
  - $S_W = \sum_{i=1}^{c} \sum_{x_j \ni X_i} (x_j - \mu_i)(x_j - \mu_i)^T$
- While the scatter between the classes is:
  - $S_B = \sum_{i=1}^{c} |X_i| (\mu_i - \mu)(\mu_i - \mu)^T$



FDA Methodology

- If S_W is non singular, the optimal projection $\hat{W}$ is chosen such that:
  - $\hat{W} = argmax_W \left(\frac{|W^T S_B W|}{|W^T S_W W|} \right) = [w_1 \; w_2 \; \ldots w_m]$
- w_i are the generalized eigenvectors of $S_B \; and \; S_W$ corresponding to generalized eigenvalues λ_i
  - $S_B w_i = \lambda_i S_W w_i$

Visual example



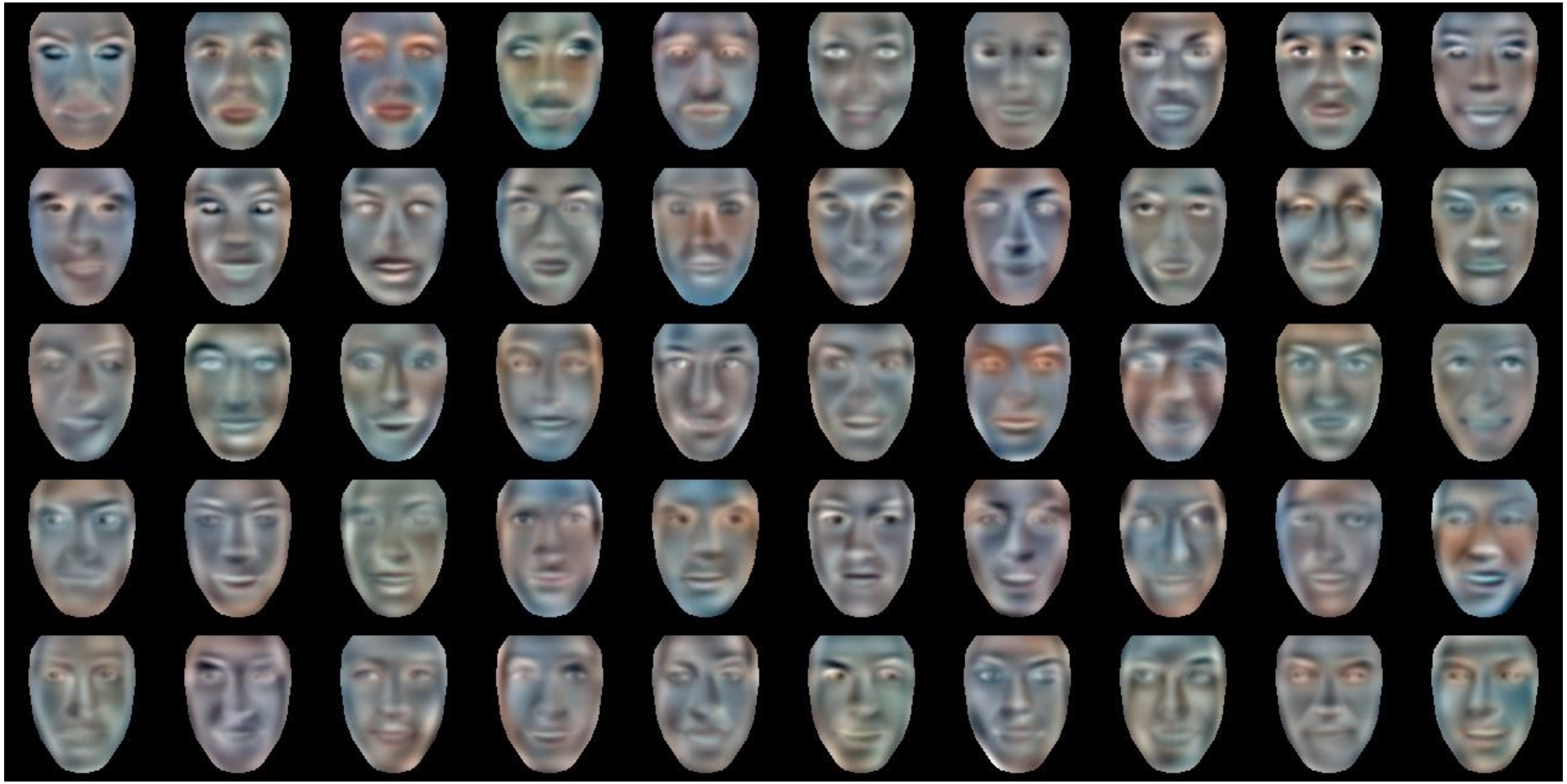
Fisherfaces

- In face recognition S_W is always singular!
  - This is because the number of pixels in each image is always larger than the number of images in the training set
- We can avoid this problem by projecting the image set to a lower dimensional space such that S_W is non singular
- This is done with PCA to $N - c$ and then using LDA to reduce to $c - 1$

Fisherfaces

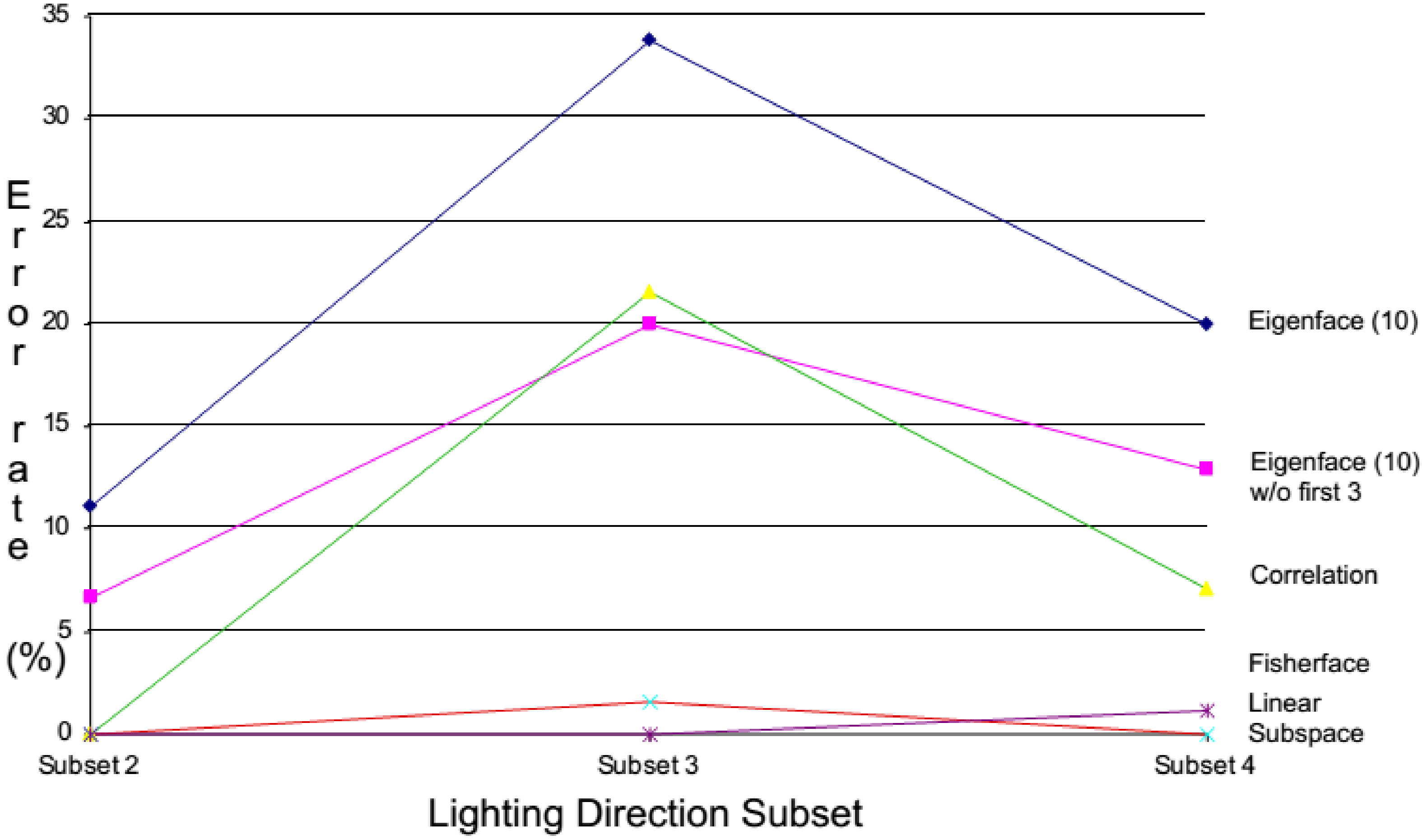
- In other words:
- $\widehat{W} = W_{fld}W_{pca}$
- $W_{pca} = argmax_W |W^T S_T W|$
- $W_{fld} = argmax_W \frac{|W^T W_{pca}^T S_B W_{pca} W|}{|W^T W_{pca}^T S_W W_{pca} W|}$

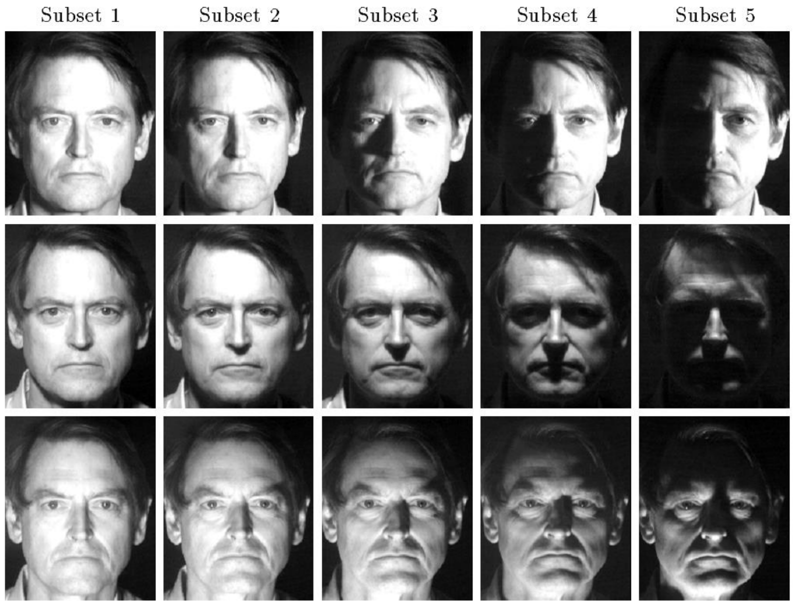
Fisherfaces



Experimental results

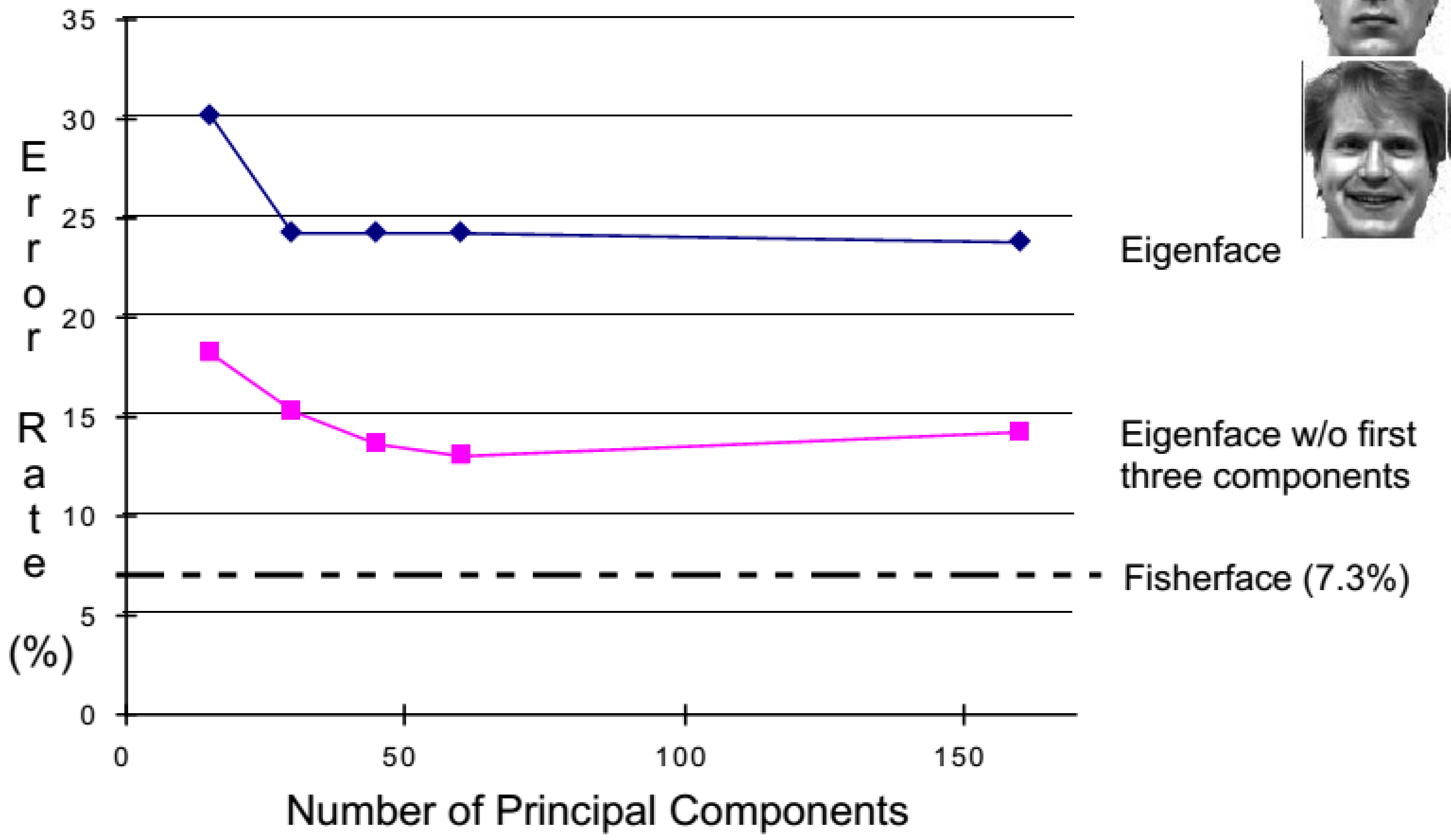
- Variation in lighting





Experimental results

- Variation in facial expressions



Conclusion

1. All methods perform well if the image presented is similar to an image in the training set
2. Fisherfaces appears to be the best over variations in lighting
3. Removing the initial three principal components in Eigenfaces improves performance over lighting variations, but the problem is still present
4. Fisherfaces is best suited for simultaneous changes in lighting and expression

Questions raised

- How well does the Fisherfaces method extend to large databases?
- Can variations in lighting be accommodated if some of the people in the training set are observed under one lighting condition?
- Face detection in Fisherfaces breaks down at extreme lighting conditions
- Performance degrades when shadowed regions dominate

References

1. Face Recognition Using Principal Components Analysis (PCA) M. Turk, A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, 3(1), pp. 71-86, 1991.
2. Turk, M., Pentland, A.: *Eigenfaces for recognition*. J. Cognitive Neuroscience **3** (1991) 71–86.
3. Pradeep Buddharaju's slides on fisherfaces
4. Belhumeur, P.,Hespanha, J., Kriegman, D.: *Eigenfaces vs. Fisherfaces: recognition using class specific linear projection*. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 711–720.

Thank you for listening!