Face Recognition Using Fuzzy Fisherface Classifier

Presenters:
Nilesh Padwal
Vivek K.
Rajat Rastogi

Contents

Introduction

- **Growing interest in biometric authentication**
  - National ID cards, Airport security, Surveillance.
  - Fingerprint, iris, hand geometry, gait, voice, vein and face.
- **Face recognition offers several advantages over other biometrics:**
  - Covert operation.
  - Public acceptance.
  - Data required is easily obtained and readily available.
- **Approaches include:**
  - Feature analysis, Appearance-Based.

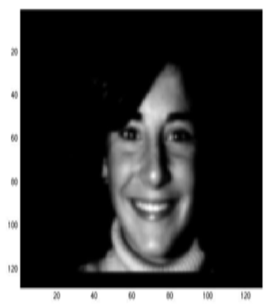
PCA Based Approach (Eigenface)

- Developed in 1991 by M.Turk
- Relatively simple
- PCA seeks directions that are efficient for representing the data
- Reduces the dimension of the data
- Speeds up the computational time

Eigenfaces, the algorithm

- Original Images $(I_1, I_2, \ldots\ldots\ldots, I_M)$

$$=\begin{pmatrix} a_1 \\ a_2 \\ \mathrm{M} \\ a_{N^2} \end{pmatrix}$$



$$=\begin{pmatrix} b_1 \\ b_2 \\ \mathrm{M} \\ b_{N^2} \end{pmatrix}$$

……

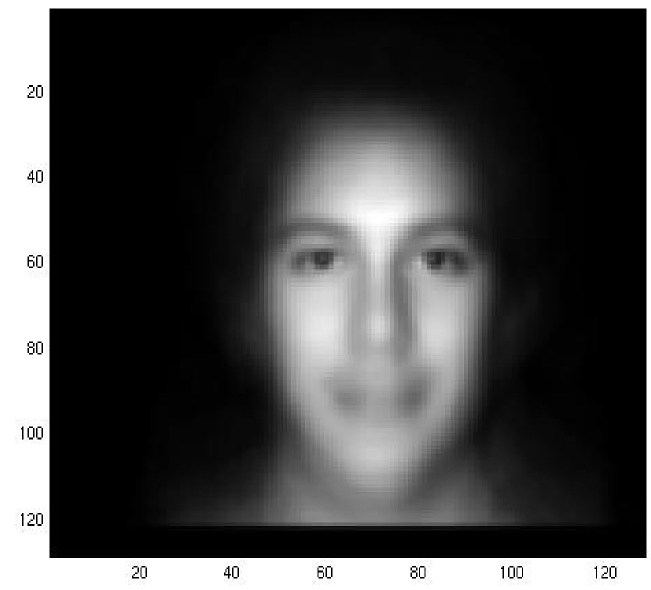
$$=\begin{pmatrix} h_1 \\ h_2 \\ \mathrm{M} \\ h_{N^2} \end{pmatrix}$$

Eigenfaces, the algorithm

- The mean face can be computed as:

$$\Psi = \frac{1}{M}\begin{pmatrix} a_1 & +b_1 & +\cdots & +h_1 \\ a_2 & +b_2 & +\cdots & +h_2 \\ \vdots & \vdots & & \vdots \\ a_{N^2} & +b_{N^2} & +\cdots & +h_{N^2} \end{pmatrix}, \quad where\ M = 8$$

Mean-Face



Eigenfaces, The Algorithm

- Then subtract it from the training faces ($\Phi_i = I_i - \Psi$)

$$\vec{a}_m = \begin{pmatrix} a_1 - m_1 \\ a_2 - m_2 \\ \vdots \quad \vdots \\ a_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{b}_m = \begin{pmatrix} b_1 - m_1 \\ b_2 - m_2 \\ \vdots \quad \vdots \\ b_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{c}_m = \begin{pmatrix} c_1 - m_1 \\ c_2 - m_2 \\ \vdots \quad \vdots \\ c_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{d}_m = \begin{pmatrix} d_1 - m_1 \\ d_2 - m_2 \\ \vdots \quad \vdots \\ d_{N^2} - m_{N^2} \end{pmatrix},$$

$$\vec{e}_m = \begin{pmatrix} e_1 - m_1 \\ e_2 - m_2 \\ \vdots \quad \vdots \\ e_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{f}_m = \begin{pmatrix} f_1 - m_1 \\ f_2 - m_2 \\ \vdots \quad \vdots \\ f_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{g}_m = \begin{pmatrix} g_1 - m_1 \\ g_2 - m_2 \\ \vdots \quad \vdots \\ g_{N^2} - m_{N^2} \end{pmatrix}, \quad \vec{h}_m = \begin{pmatrix} h_1 - m_1 \\ h_2 - m_2 \\ \vdots \quad \vdots \\ h_{N^2} - m_{N^2} \end{pmatrix}$$

Eigenfaces, The Algorithm

- Now we build the matrix which is N^2 by M

$$A = \left[\vec{a}_m \;\; \vec{b}_m \;\; \vec{c}_m \;\; \vec{d}_m \;\; \vec{e}_m \;\; \vec{f}_m \;\; \vec{g}_m \;\; \vec{h}_m \right]$$

- The covariance matrix which is N^2 by N^2

$$Cov = AA^{\mathrm{T}}$$

Eigenfaces, The Algorithm

- Compute the Eigenvectors(N^2), u_i of AA^T
- Matrix AA^T is very large, so computing all Eigenvectors not practical
- Compute the Eigenvectors(M), v_i of A^TA
- AA^T and A^TA have the same eigenvalues and their eigenvectors are related as follows!!

$$u_i = Av_i$$

- Keep only K eigenvectors (corresponding to K largest values)

Eigenfaces, The Algorithm

- Each training image is projected to face space using

$$w_j = u_j^T \Phi_i$$

- Each Φ_i can be represented as a vector Ω_i as follows!!!

$$\Omega_i = \begin{bmatrix} w_1^i \\ w_2^i \\ \cdots \\ w_K^i \end{bmatrix}, \quad i = 1, 2, \ldots, M$$

Eigenfaces, The Algorithm

- For each test image Ω is found
- $e_i=|\Omega- \Omega_i|$
- Test image is assigned to nearest training sample in the face space

Disadvantages of Eigenface Approach

- Sensitive to large variations in lighting
- Facial Expressions

Because it maximizes the total scatter across all classes but it retains the unwanted variations due to lighting and facial expressions

Different Lighting Conditions

- Same person appears different and PCA suffers

courtesy:Source [4]

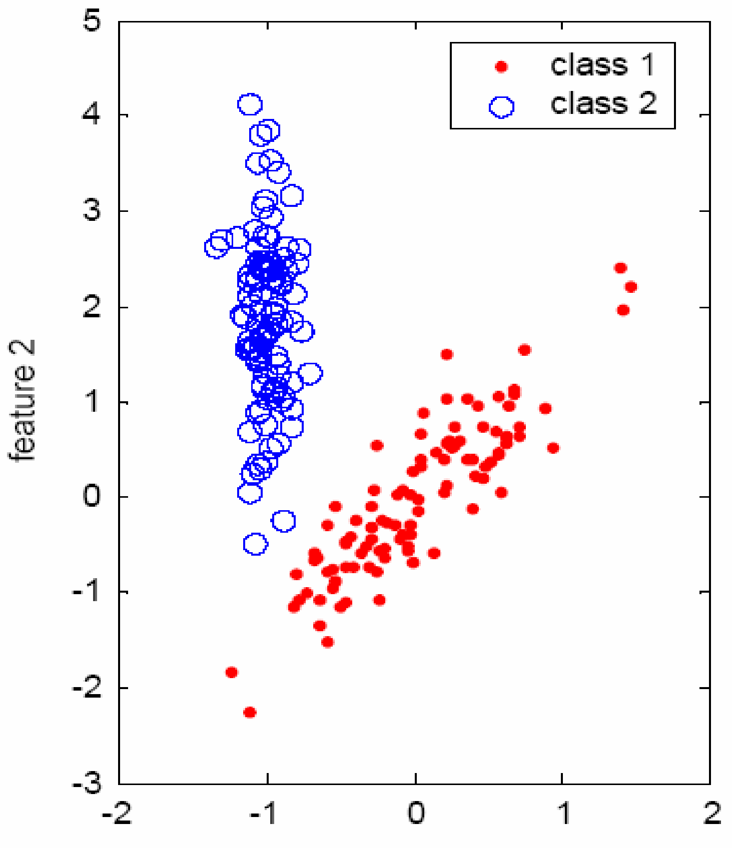
Fisherface Approach

- It is class specific method
  - Shapes the scatters in order to make it more reliable for classification

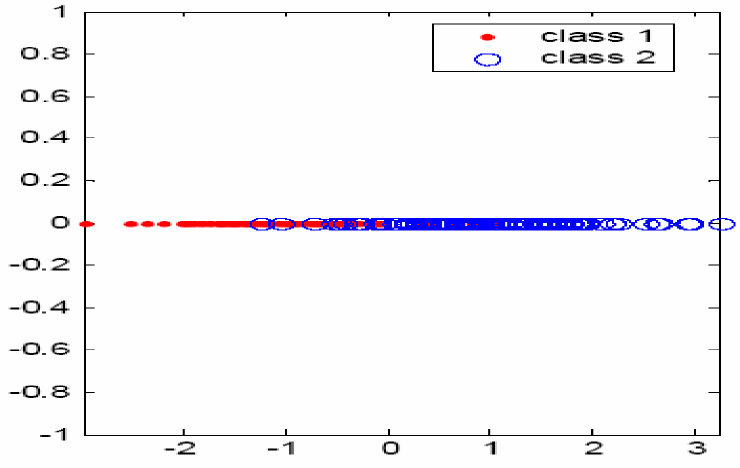
Principle:

- Projects the image set to a lower dimension space using PCA , followed by the FLD phase
- PCA helps us achieve non-singularity of S_W prior to computation of optimal projection W_{FLD}

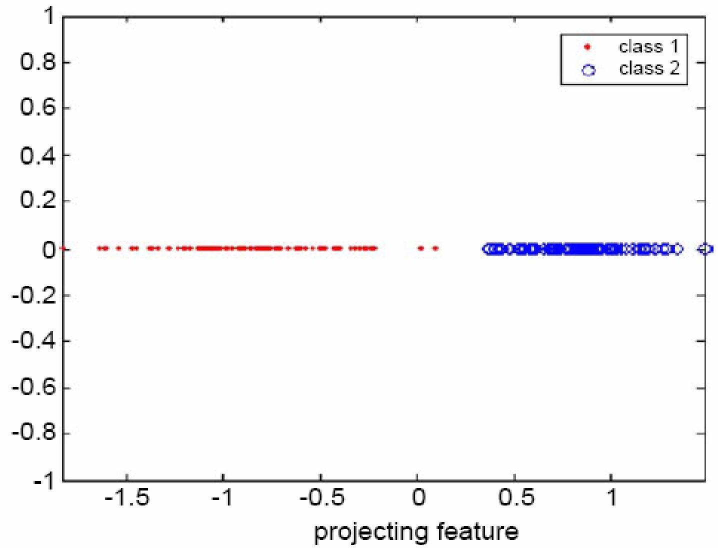
Comparison of PCA and FLD for Two Class Data



(a)



(b)



Fisher Linear Discriminant

- Between Class Scatter Matrix S_B

$$S_{\mathrm{B}} = \sum_{i=1}^{c} N_i (\mathbf{m}_i - \overline{\mathbf{m}})(\mathbf{m}_i - \overline{\mathbf{m}})^{\mathrm{T}}$$

N_i- Number of samples in class X_i
m_i- mean image of class X_i

$\overline{m}$ -mean of all the images

- Within Class Scatter Matrix S_W

$$S_{\mathrm{W}} = \sum_{i=1}^{c} \sum_{x_k \in C_i} (\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^{\mathrm{T}} = \sum_{i=1}^{c} S_{\mathrm{W}_i}$$

c- number of classes in training samples

Fisher Linear Discriminant

- Optimal Projection Matrix
  - It maximizes the ratio of the determinant of between class scatter matrix of projected patterns to the determinant of within class scatter matrix of projected patterns

$$W_{\mathrm{FLD}} = \arg\max_{W} \frac{|W^{\mathrm{T}} S_{\mathrm{B}} W|}{|W^{\mathrm{T}} S_{\mathrm{W}} W|} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \cdots \quad \mathbf{w}_m]$$

- Where $\{\mathbf{w}_i | i=1, 2, \ldots, m\}$ is the set of eigenvectors of S_B and S_W corresponding to m largest eigenvalues $\{\lambda_i | i=1, 2, \ldots, m\}$

$$S_{\mathrm{B}} \mathbf{w}_i = \lambda_i S_{\mathrm{W}} \mathbf{w}_i, \quad i = 1, 2, \ldots, m.$$

- Rank of S_B is c-1 and rank of S_W is at most N-c

Fisherface Approach

- In the face recognition problem S_W matrix is always singular (number of images in learning set N is much smaller the number of pixels in each image)
- Fisherface avoids this problem by projecting the image set to a lower dimension space using PCA and then applying standard FLD

Fisherface Approach

- Optimal projection matrix

$$W_{opt}^T = W_{fld}^T W_{pca}^T$$

where

$$W_{pca} = \arg\max_W \left| W^T S_T W \right|$$

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

- Optimization for W_{PCA} is performed over $n \times (N - c)$ matrices with orthonormal columns

- While the optimization for W_{FLD} is performed $(N - c) \times m$ over matrices with orthonormal columns

Fuzzy Fisherface Approach

- More sophisticated usage of class assignment of patterns (faces)
- Classification results affect the within-class and between-class scatter matrices

Algorithm

- Given set of feature vectors transformed by the PCA,

$$X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\},$$

- Partition matrix

$$\mathrm{U} = [\ \mu_{ij}\ ] \quad \text{for } i = 1, 2, \ldots, c \text{ and } j = 1, 2, \ldots, N$$

Which satisfies, $\sum_{i=1}^{c} \mu_{ij} = 1$

$$0 < \sum_{j=1}^{N} \mu_{ij} < N$$

The Computations Of Membership Degrees

- Compute the Euclidean distance matrix between pairs of feature vectors in the training,
- Set diagonal elements of this matrix to infinity,
- Sort the distance matrix in ascending order,
- Collect the class labels of the patterns located in the closest neighborhood of the pattern,

The Computations Of Membership Degrees

- Compute the membership grade to class i for jth pattern ,

$$\mu_{ij} \begin{cases} 0.51 + 0.49(n_{ij}/k) & \text{if } i = \text{same as the label of the } jth \text{ pattern} \\ 0.49(n_{ij}/k) & \text{if } i \neq \text{same as the label of the } jth \text{ pattern} \end{cases}$$

where n_{ij} is number of the neighbors of the jth data that belong to ith class

FKNN Initialization

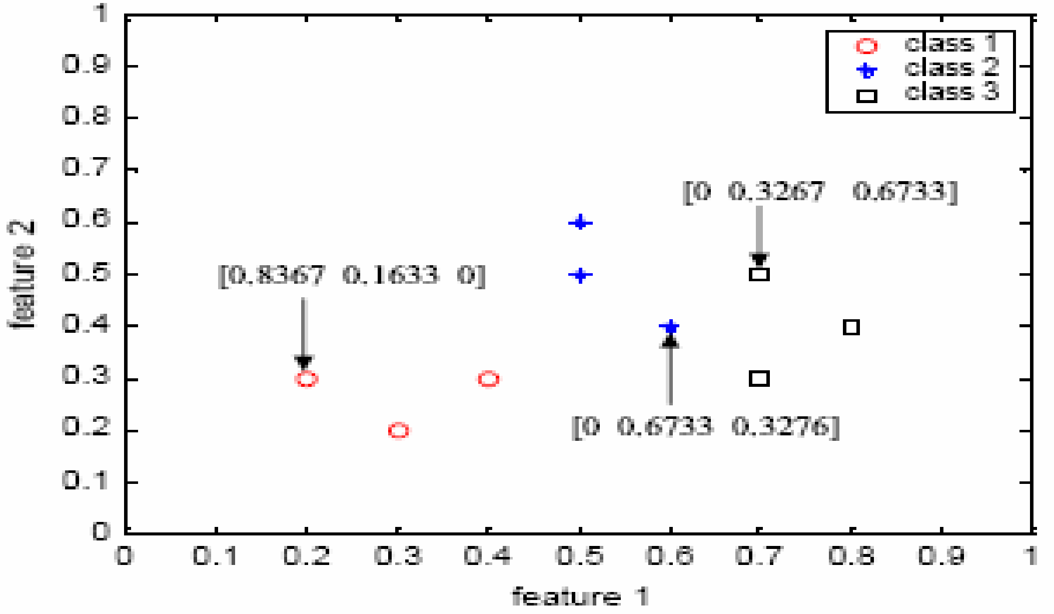


Fig. Fuzzy membership degree using FKNN initialization (k=3).

courtesy:Source [1]

Algorithm

- Results of FKNN are used in computations of mean value and scatter covariance matrices,
- Mean vector of each class

$$\tilde{m}_i = \frac{\sum_{j=1}^{N} \mu_{ij} X_j}{\sum_{j=1}^{N} \mu_{ij}}$$

- The between class and within class fuzzy scatter matrices are respectively,

$$S_{FB} = \sum_{i=1}^{c} N_i (\tilde{m}_i - \overline{m})(\tilde{m}_i - \overline{m})^T$$

$$S_{FW} = \sum_{i=1}^{c} \sum_{x_k \in C_i} (x_k - \tilde{m}_i)(x_k - \tilde{m}_i)^T = \sum_{i=1}^{c} S_{FW_i}$$

Algorithm

- The optimal fuzzy projection $W_{F\text{-}FLD}$ and feature vector transformed by fuzzy fisherface method are given by

$$W_{F-FLD} = \arg \max_{W} \frac{\left| W^T S_{FB} W \right|}{\left| W^T S_{FB} W \right|}$$

$$\tilde{v}_i = W_{F-FLD}^T X_i = W_{F-FLD}^T E^T (z_i - \bar{z})$$

Flowchart

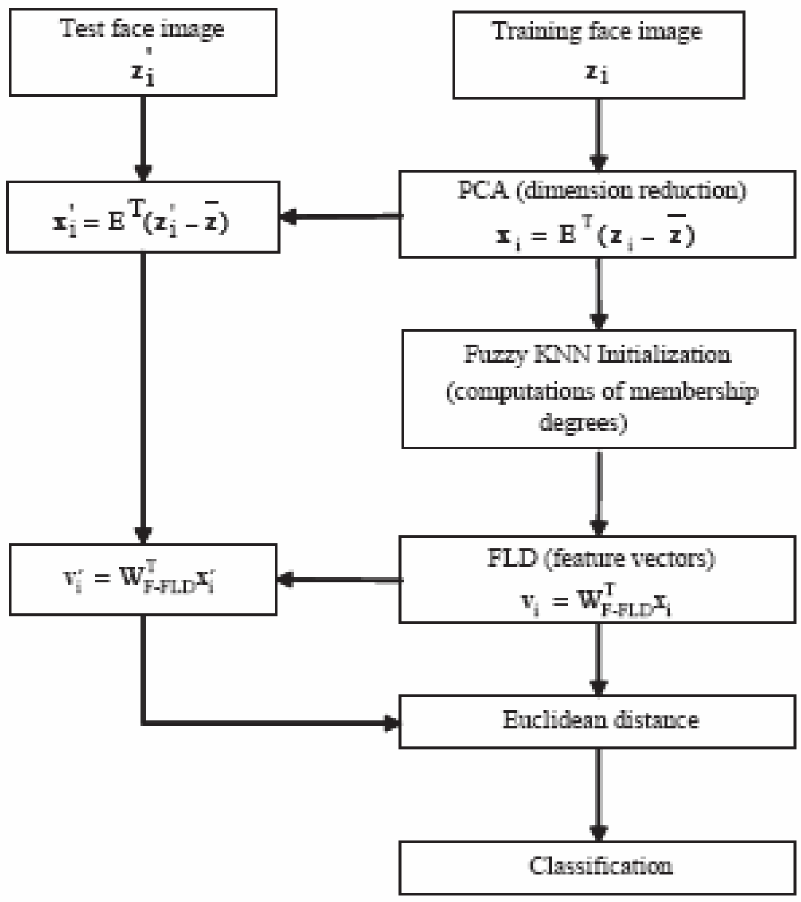


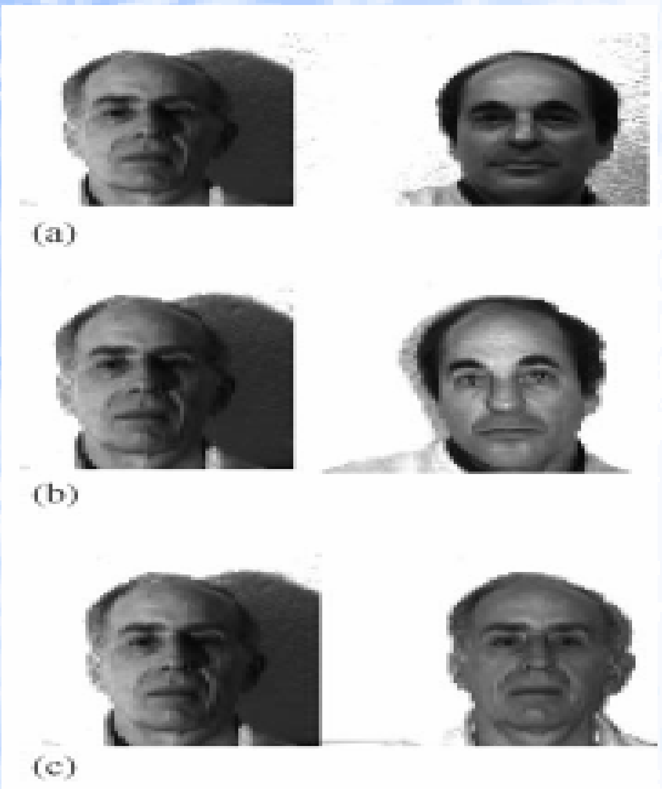
Fig. A general flow of computing for the fuzzy fisherface method.

courtesy:Source [1]

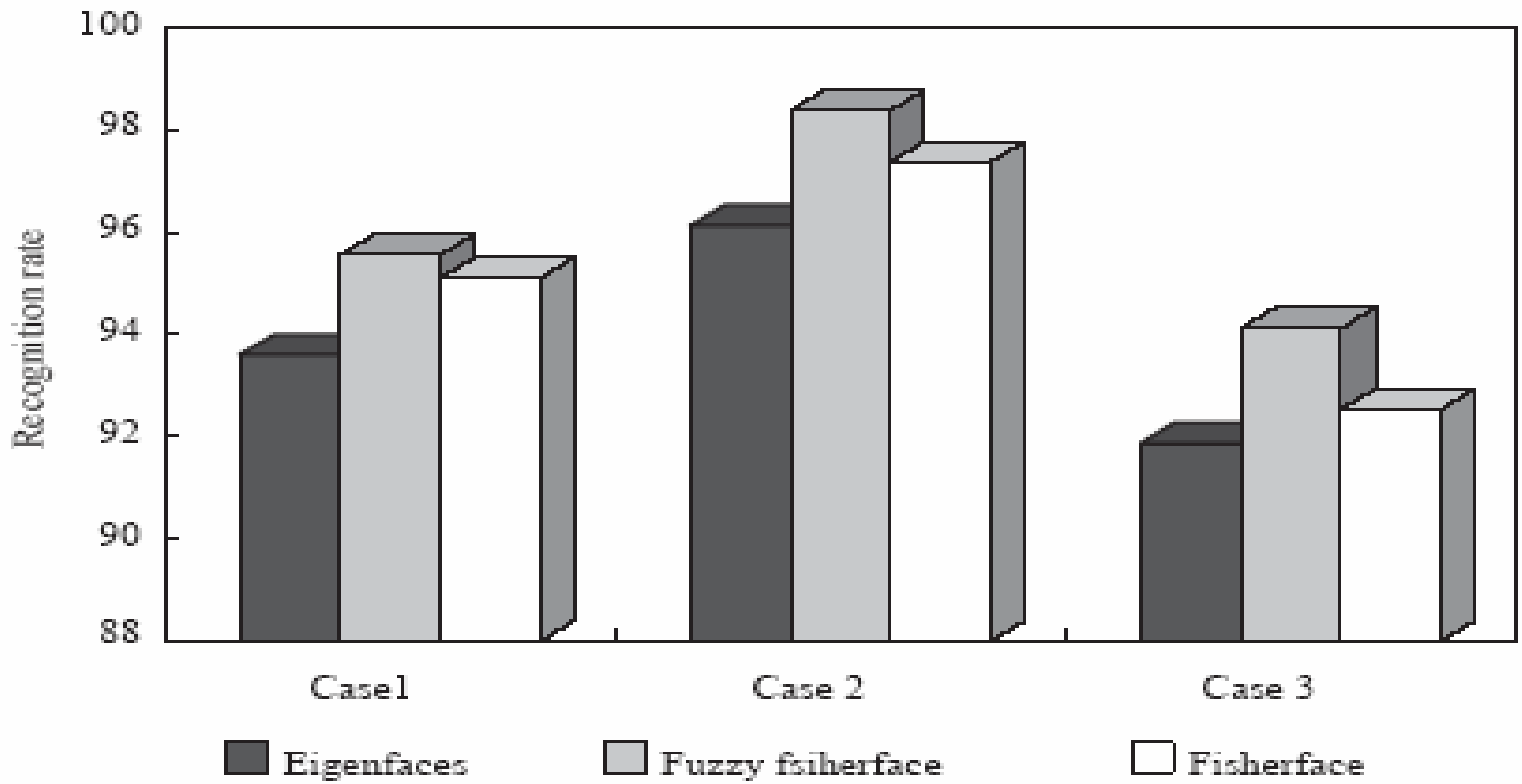
Comparison

- Eigenface
- Fisherface
- Fuzzy Fisherface

Test Image **Recognized Image**



Comparison of Recognition Rates



Conclusion

- Fuzzy fisherface approach outperform the other two methods for the datasets considered.
- Sensitivity variations in illumination and facial expression reduced substantially.
- Fuzzy sets can efficiently manage the vagueness and ambiguity of face images degraded by poor illumination component.

References

- Keun-Chang Kwak, Witold Pedrycz : Face Recognition Using Fuzzy Fisherface Classifier, Science Direct Journal Of Pattern Recognition Society 38(2005),1717-1732
- Turk, M., Pentland, A.: Eignefaces for Recognition. Journal of Cognitive Neuroscience, Vol.3, (1991) 72-86
- Turk, M., Pentland, A.: Face Recognition Using Eignefaces. In Proc. IEEE Conf. On Computer Vision and Pattern Recognition. (1991) 586-591
- Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: Face Recognition using class specific linear projection. In Proc. ECCV, (1996) 45-58