Subspace Analysis for Facial Image Recognition: A Comparative Study

Yongbin Zhang, Lixin Lang and Onur Hamsici

Outline

1. Subspace Analysis: Linear vs Kernel
2. Appearance-based Facial Image Recognition.
3. Databases : *FERET* and *JAFFE*.
4. Experimental Results.
5. Concluding Remarks.

Linear and Nonlinear Methods

Linear Subspace

Unsupervised

- PCA
- ICA

Supervised

- LDA

Nonlinear Subspace

- Kernel PCA
- Kernel ICA
- Kernel LDA

Principle Component Analysis (PCA)

- Basic Idea
  - Assume samples $X = [x_1, x_2, \ldots, x_p]^T$
  - $Y_i = f_i^T X, \quad f_i^T f_i = 1$
  - $f_1 = \text{argmax\_}f_1 \, Var(Y_1)$
    - $f_2 = \text{argmax\_}f_2 \, Var(Y_2), \; Cov(Y_1, Y_2)=0$
    - • • •
- Data redundancy ---- measured by correlation

Subspace Approaches: Linear Subspace : PCA

PCA Algorithm

- Subspace basis

$$C\, f_i \,{}^{=}\, f_i\, ?_i$$

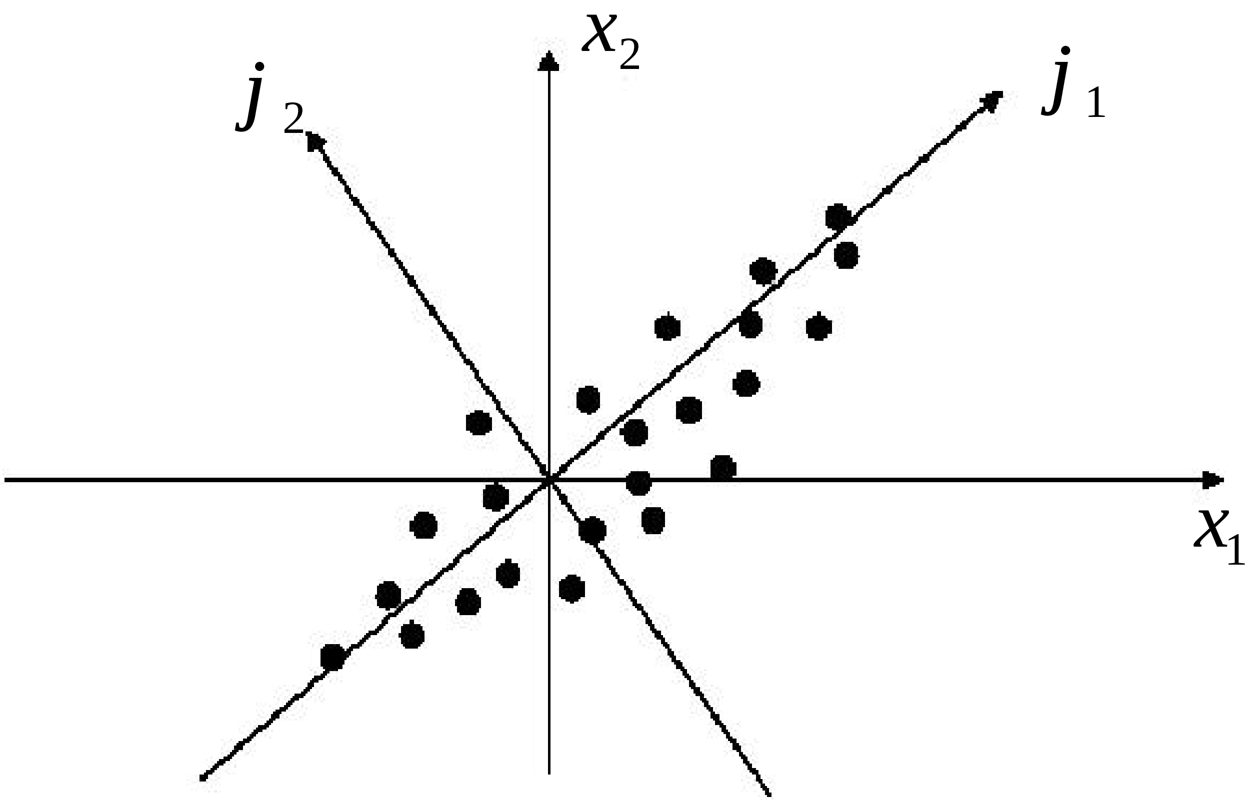
where C is Sample Covariance, and $?_1 = ?_2 = \ldots = ?_p$

- Projection to subspace

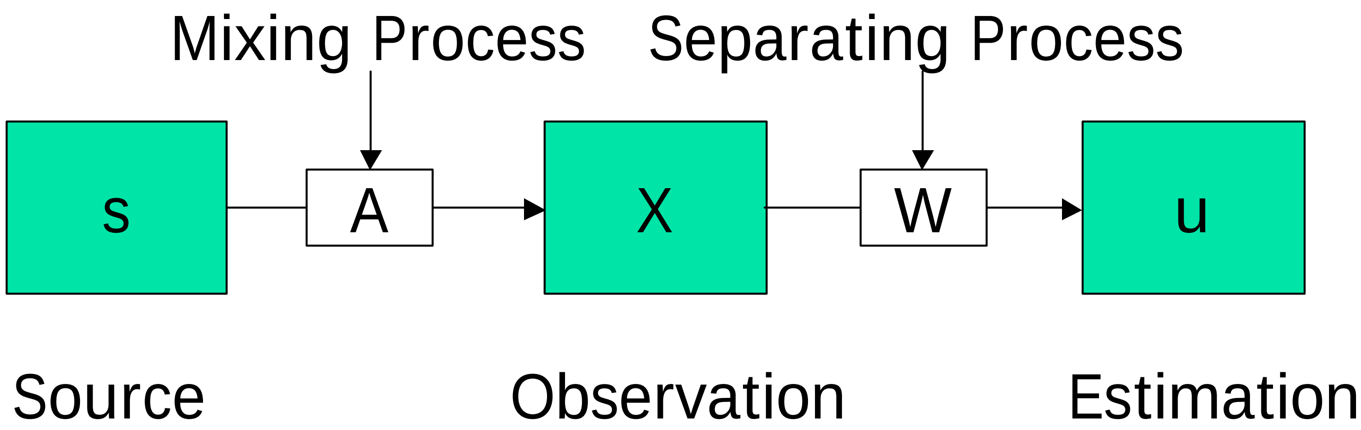
$$Y = F^T(X - E(X))$$

where $F = [f_1\ f_2, \ldots, f_m],\ m = p$

PCA Illustration



Independent Component Analysis (ICA)



Observation : X = As

Estimation : u = Wx = WAs, estimation of s

Courtesy of Bruce A Draper et al.

ICA Algorithm

- No closed-form solution.

  - Minimizing mutual information
  - Maximizing Non-Gaussianity
    - Kurtosis: $E\{x^4) - 3(E\{x^2\})^2$
    - Kurtosis is 0 for Gaussian random variables
  - InfoMax Principle

Linear Discrimination Analysis (LDA)

- Basic Idea:

$$w = \mathbf{arg\,max} \quad \frac{w^T S_b w}{w^T S_w w}$$

$$S_b \mathbf{:} \ \text{between-class scatter matrix}$$

$$S_w \mathbf{:} \ \text{within-class scatter matrix}$$

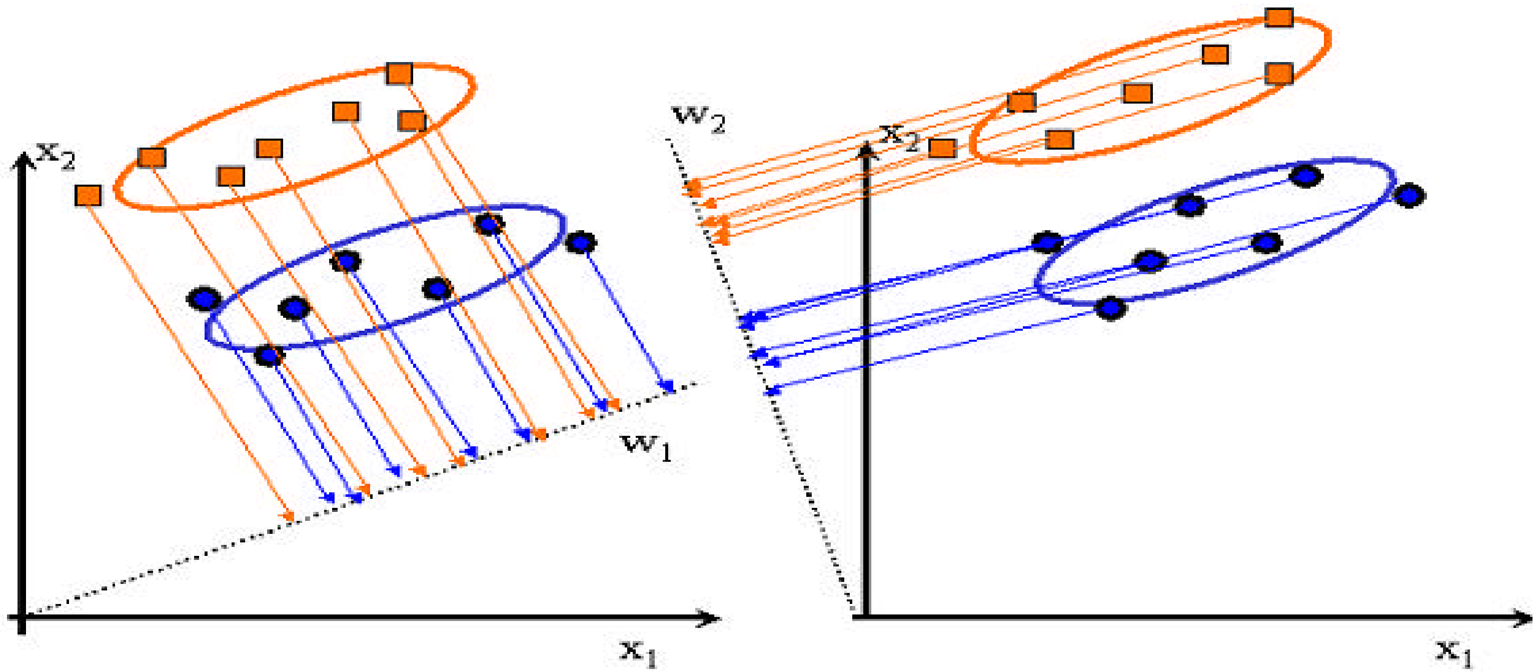
- Explicit knowledge of class information

LDA Algorithm

- Generalized Eigenvector and Eigenvalue

$$S_b f_i = l_i S_W f_i$$

LDA Illustration



Kernel Version

- PCA, ICA and LDA are linear in nature.
- Sample data are nonlinear in general.

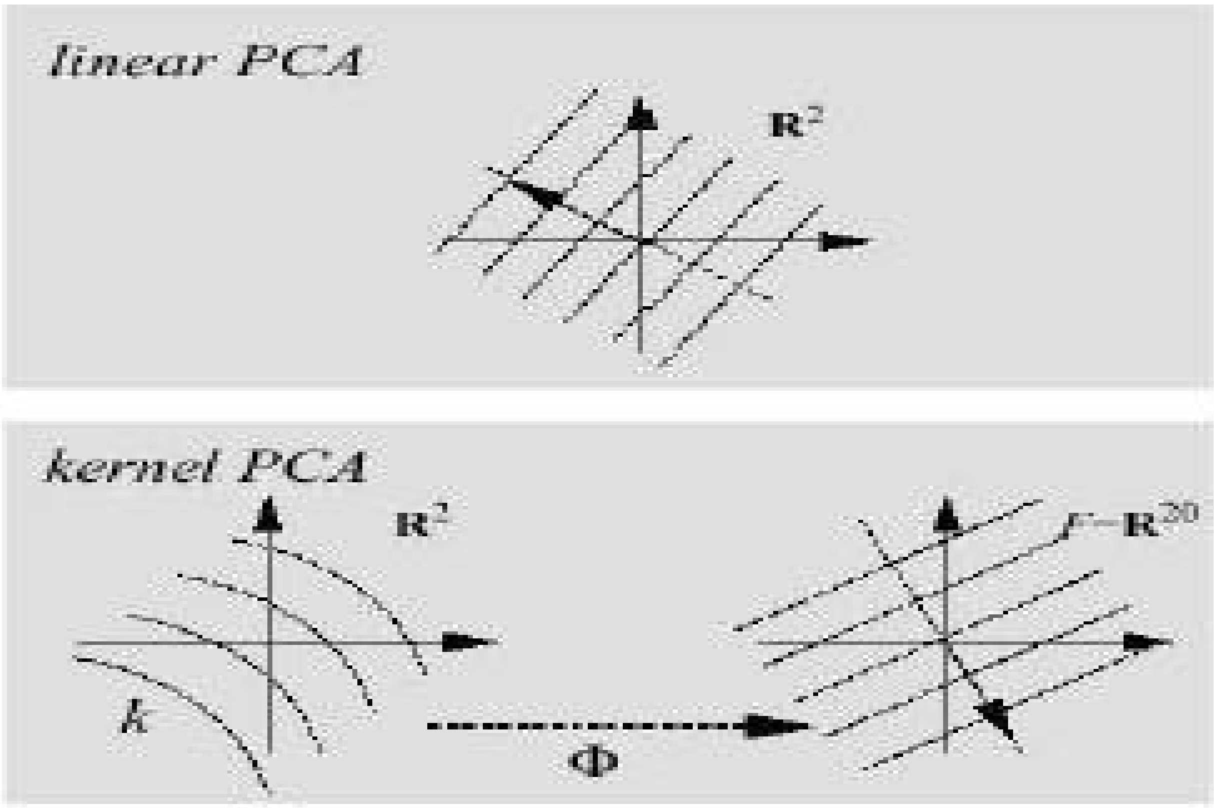
- Map to a nonlinear space and perform PCA, ICA and LDA

  PCA -→ KPCA

  ICA -→ KICA

  LDA -→ KLDA (KDA)

Illustration



Courtesy of Schelkolpf et.al

Kernel Trick

- Compute dot production without mapping function

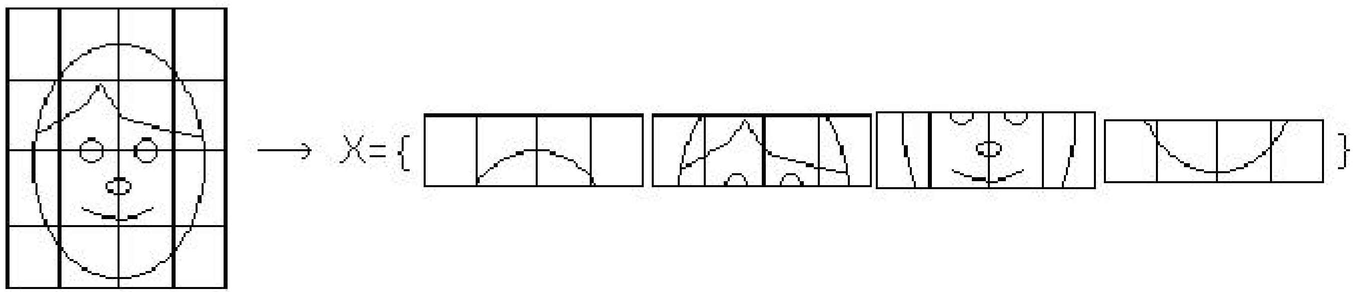
- Kernel Functions

$$\text{Polynomial}: \; k(x, y) = (x \cdot y)^d$$

$$\text{Gaussian}: \quad k(x, y) = \mathbf{exp}\left(- \frac{\left\| x - y \right\|^2}{2s^2} \right)$$

Appearance-Based Technique

- Feature: the intensity of all the pixels in the original image(or image after filtering).



- Pros: Simple, No explicit model needed.

- Cons: Two many variables, most of which dependent.

Why Dimension Reduction is Important?

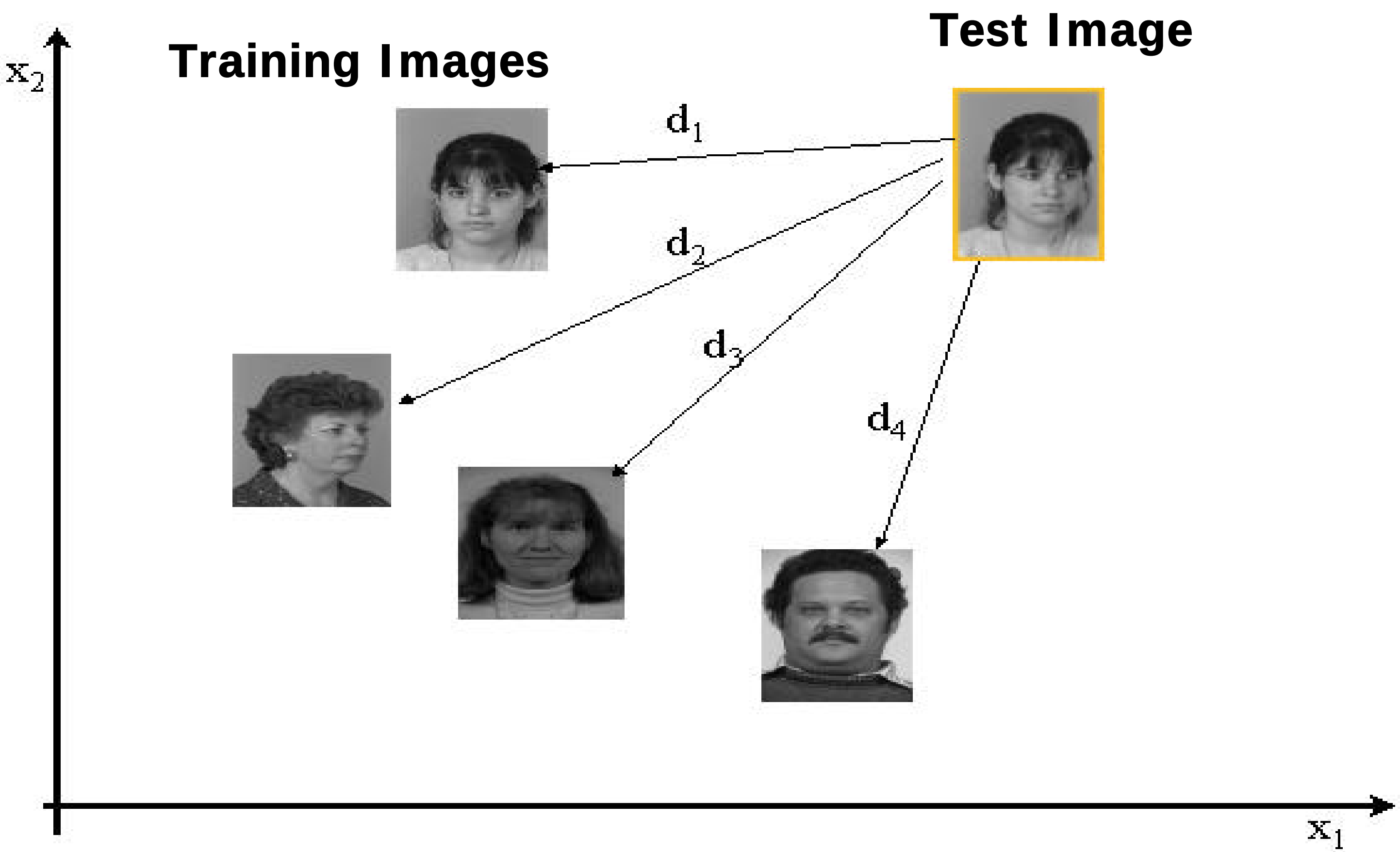
- Curse of Dimensionality.

To achieve generalization, we need at least ten times as many training samples per class as the feature dimension.

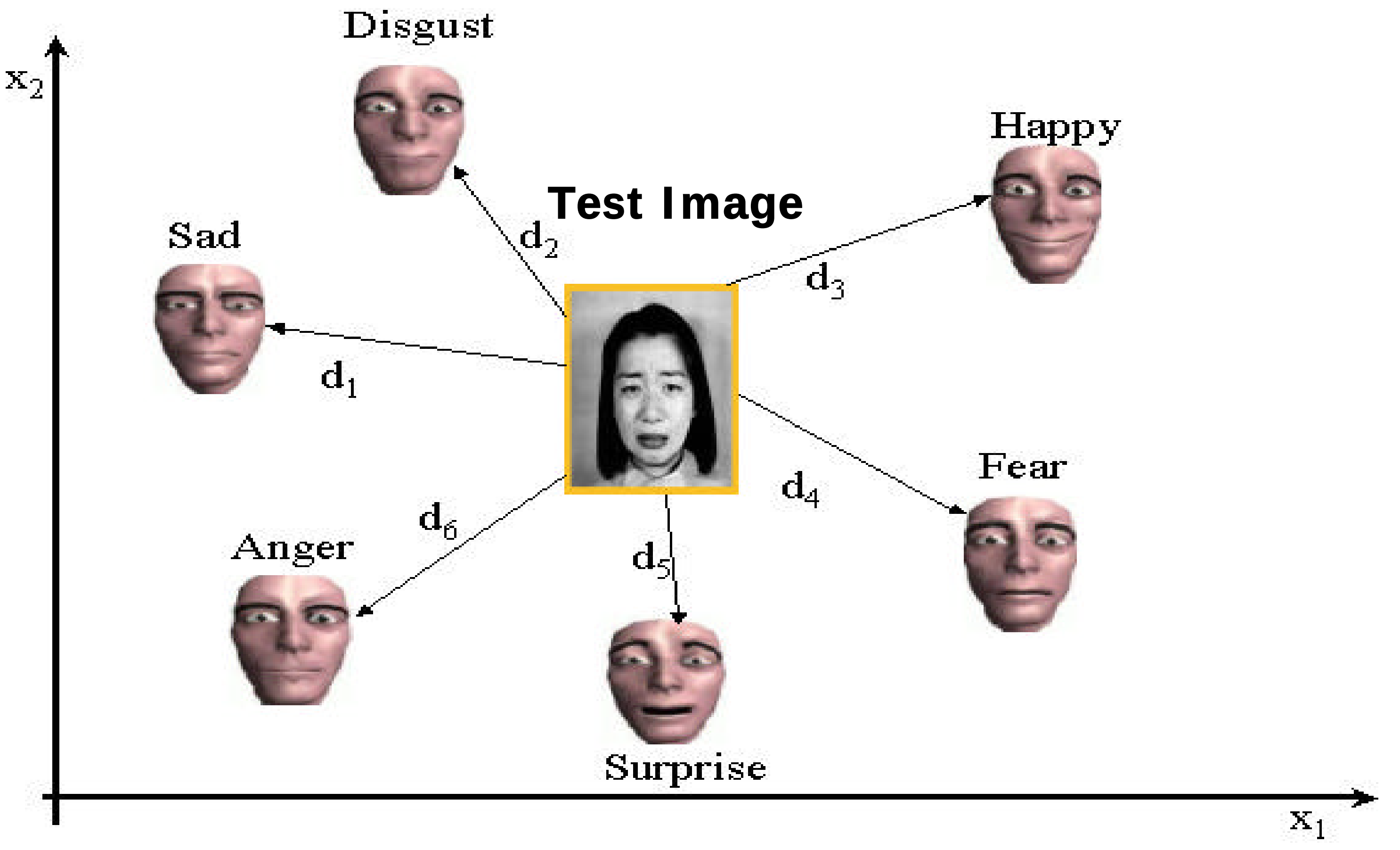
- Reduce Computational Complexity and Storage.

Techniques: PCA, LDA ,ICA vs Kernel Version

Subspace Approach : Face



Subspace Approach : Expression



Data Base : Facial Images

- FERET(Face Recognition Technology)

195 subjects, each with images of 9 different poses.

We will test identity recognition across poses

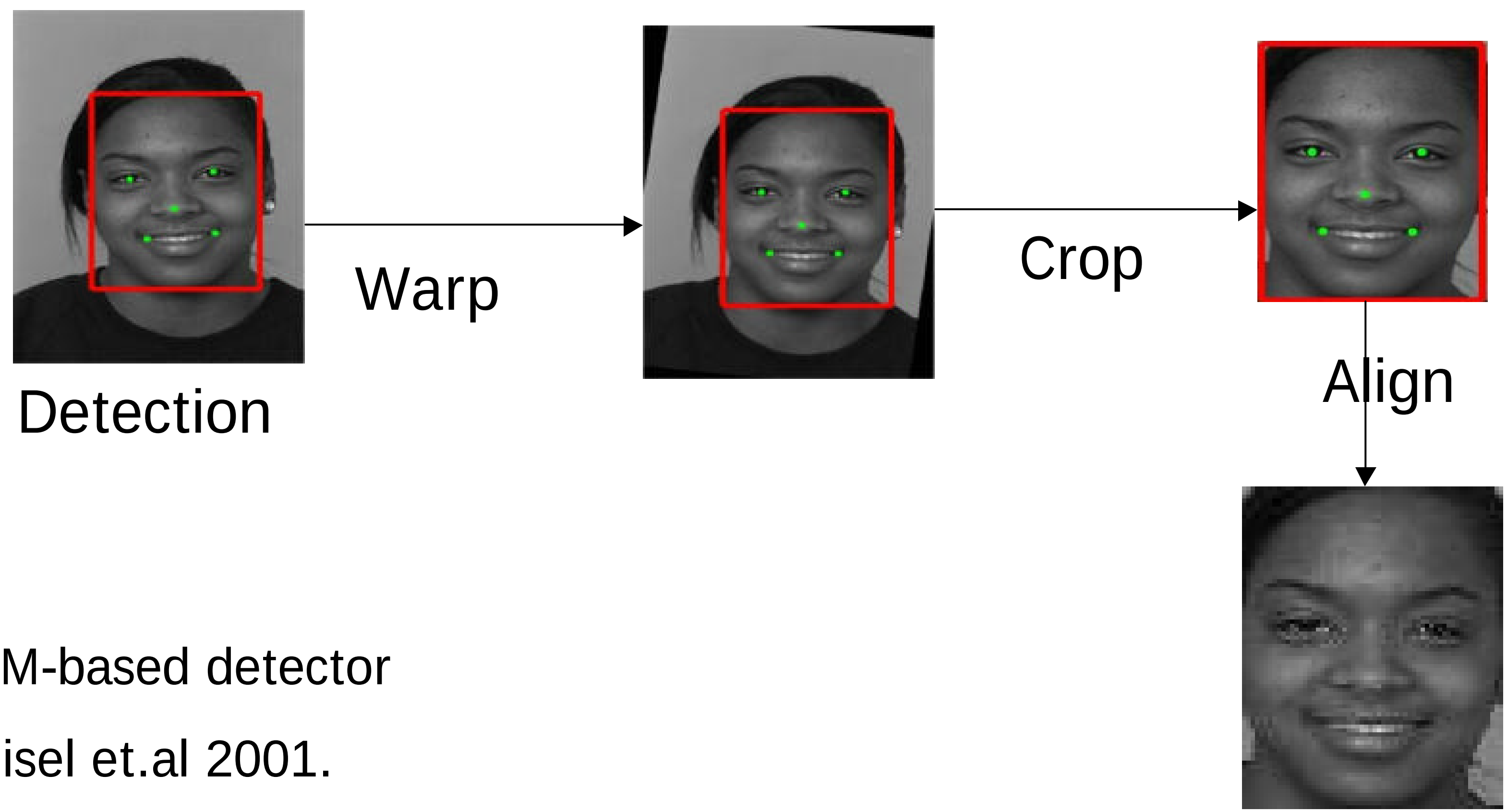
Data Base : Expression Images

- JAFFE(Japanese female expression)

213 images, 7 expression(neutral + 6 basic emotions)

Image Pretreat

- Face Processing: Detection, Warping, Alignment.

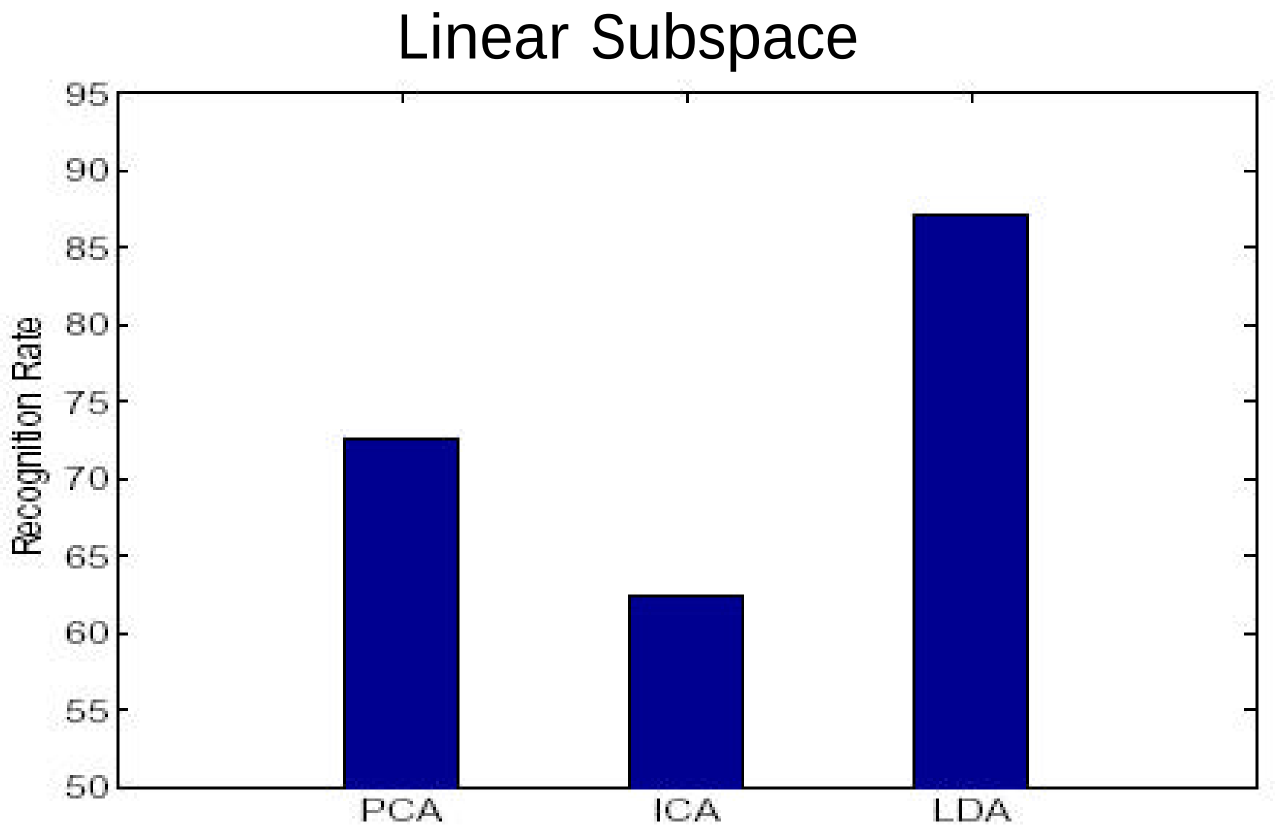


SVM-based detector

Heisel et.al 2001.

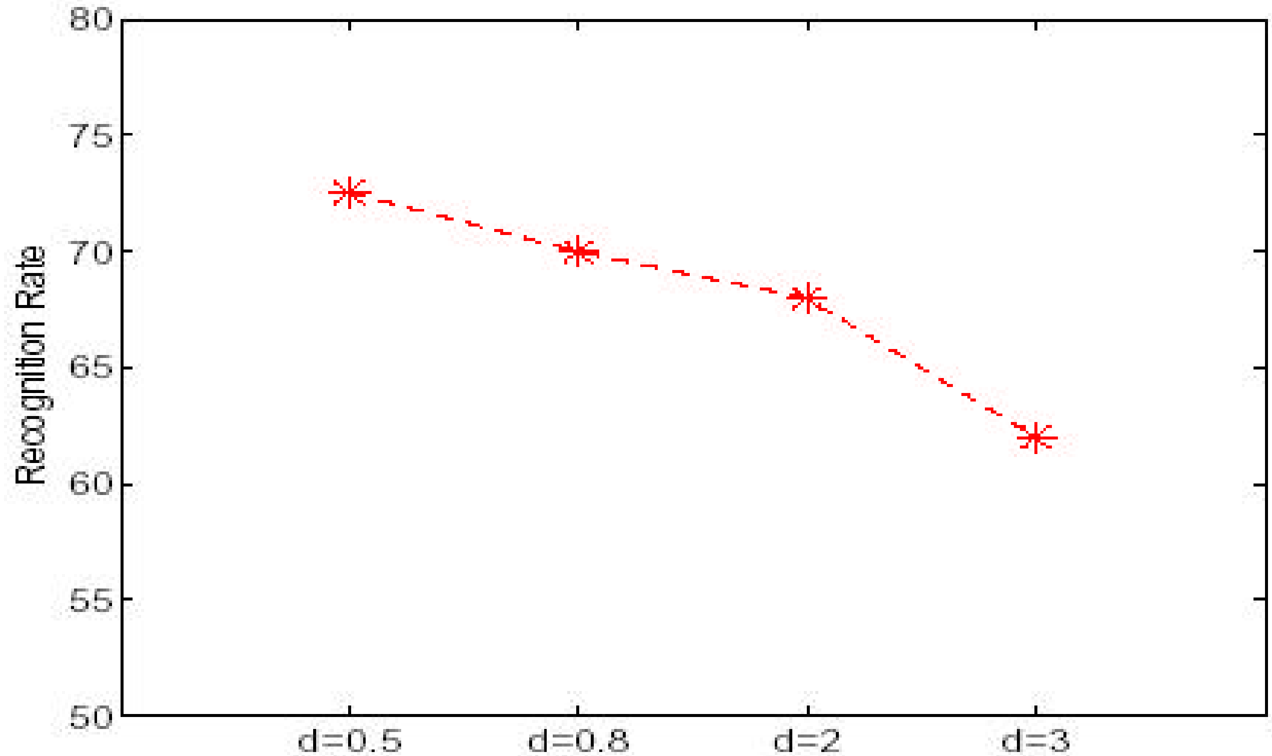
Experimental Results: Identity Recognition

- Leave-one-pose out test.



Experimental Results: Identity Recognition

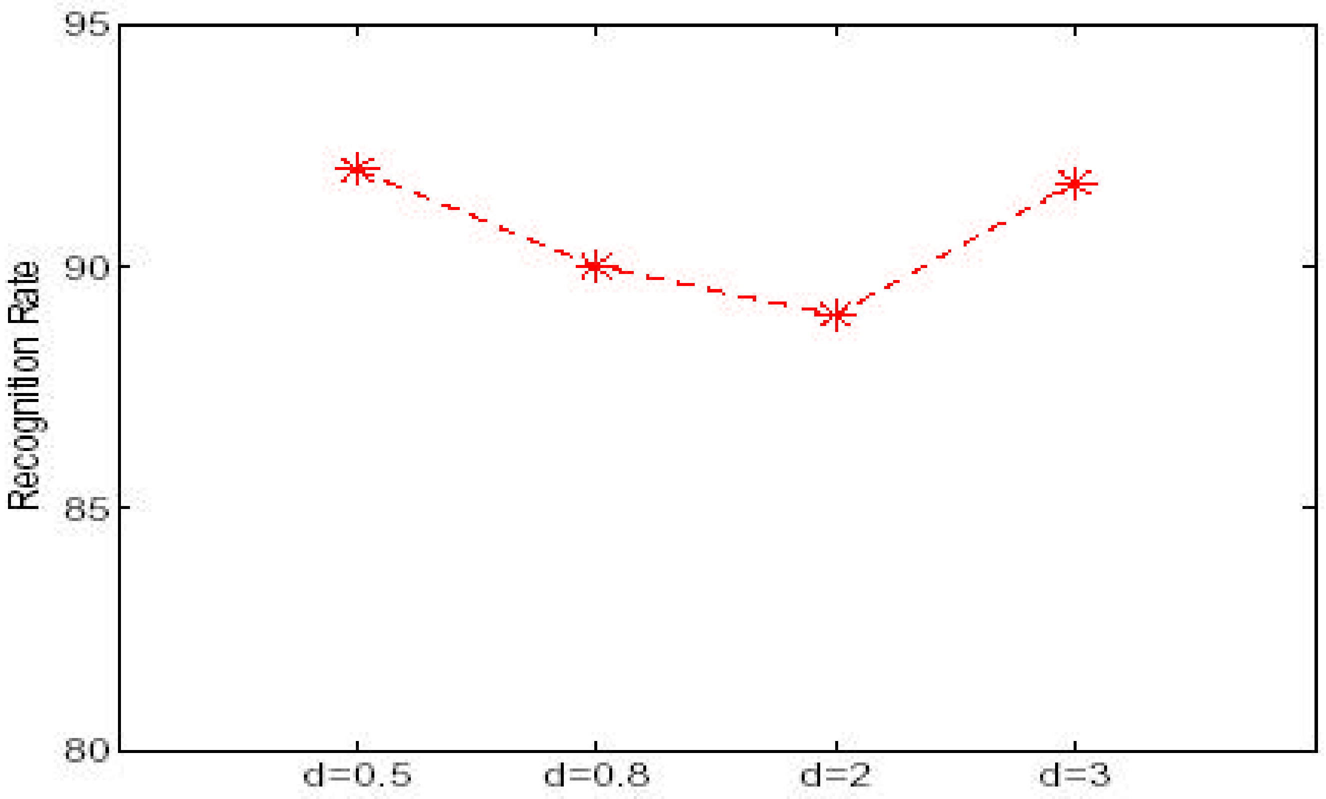
KPCA with different degree of polynomials



Fractional Power: d<1(Liu et.al. 2004)

Experimental Results: Identity Recognition

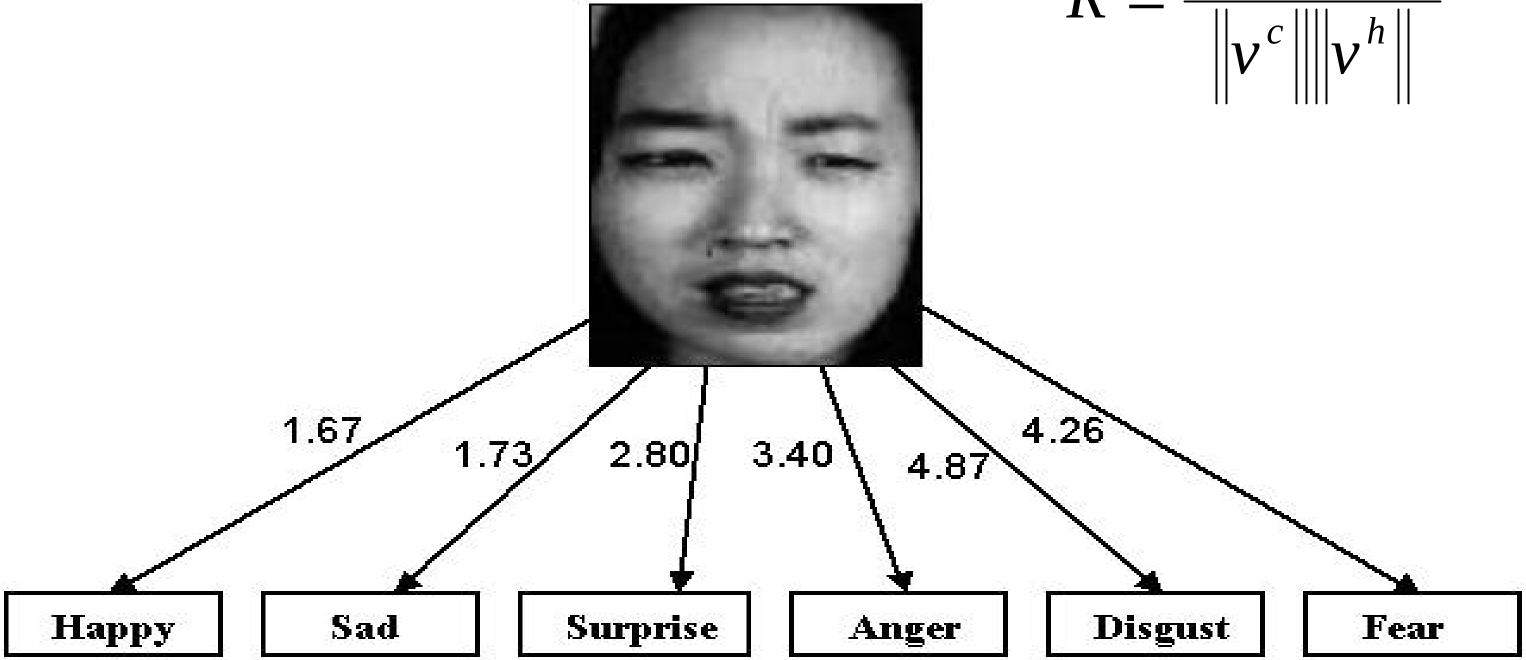
KLDA with different degree of polynomials



Experimental Results: Facial Expression

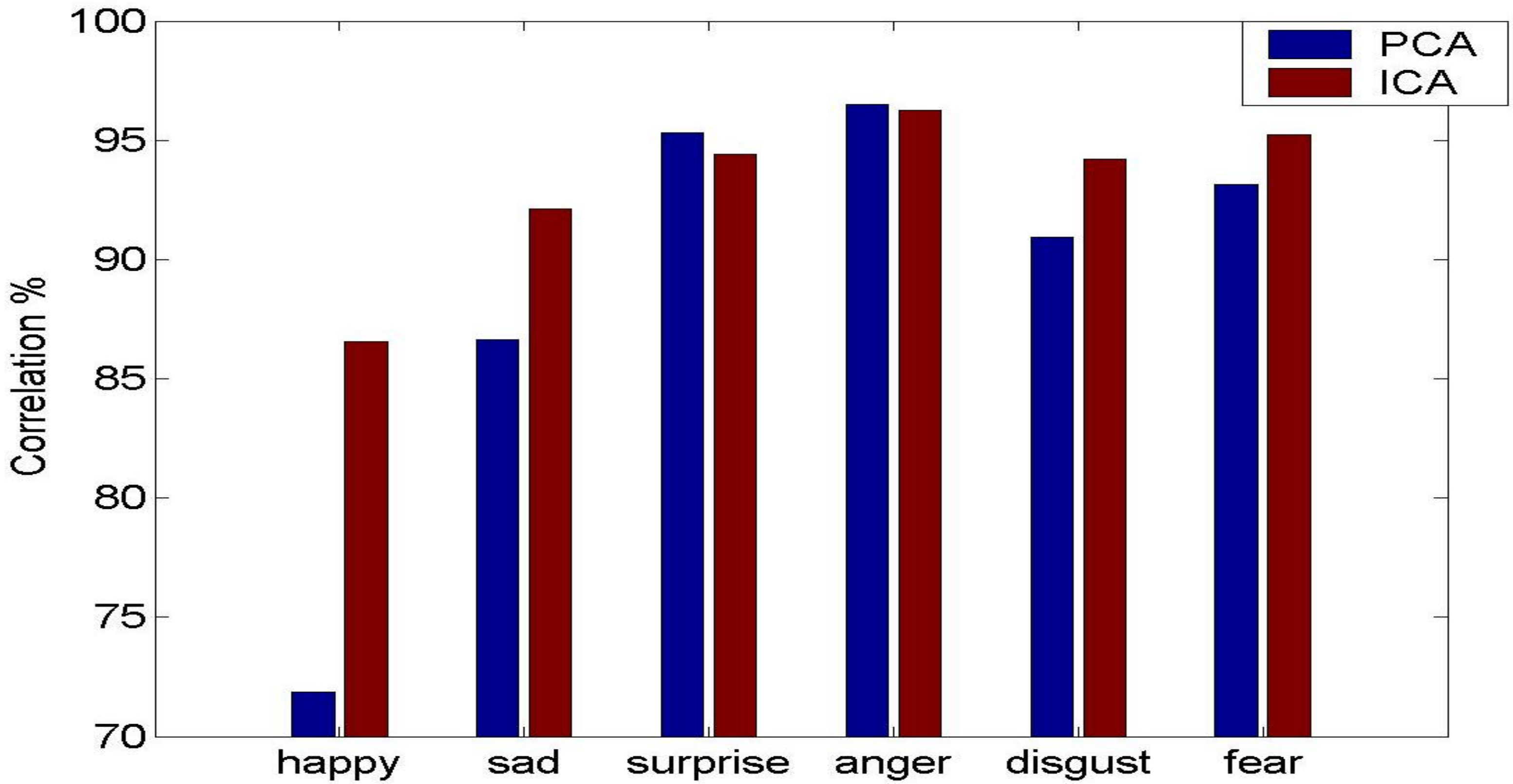
- “Hard” label is not suitable! We will rate the images and compare with what human does.

$$R = \frac{|\langle v^c, v^h \rangle|}{\|v^c\|\|v^h\|}$$



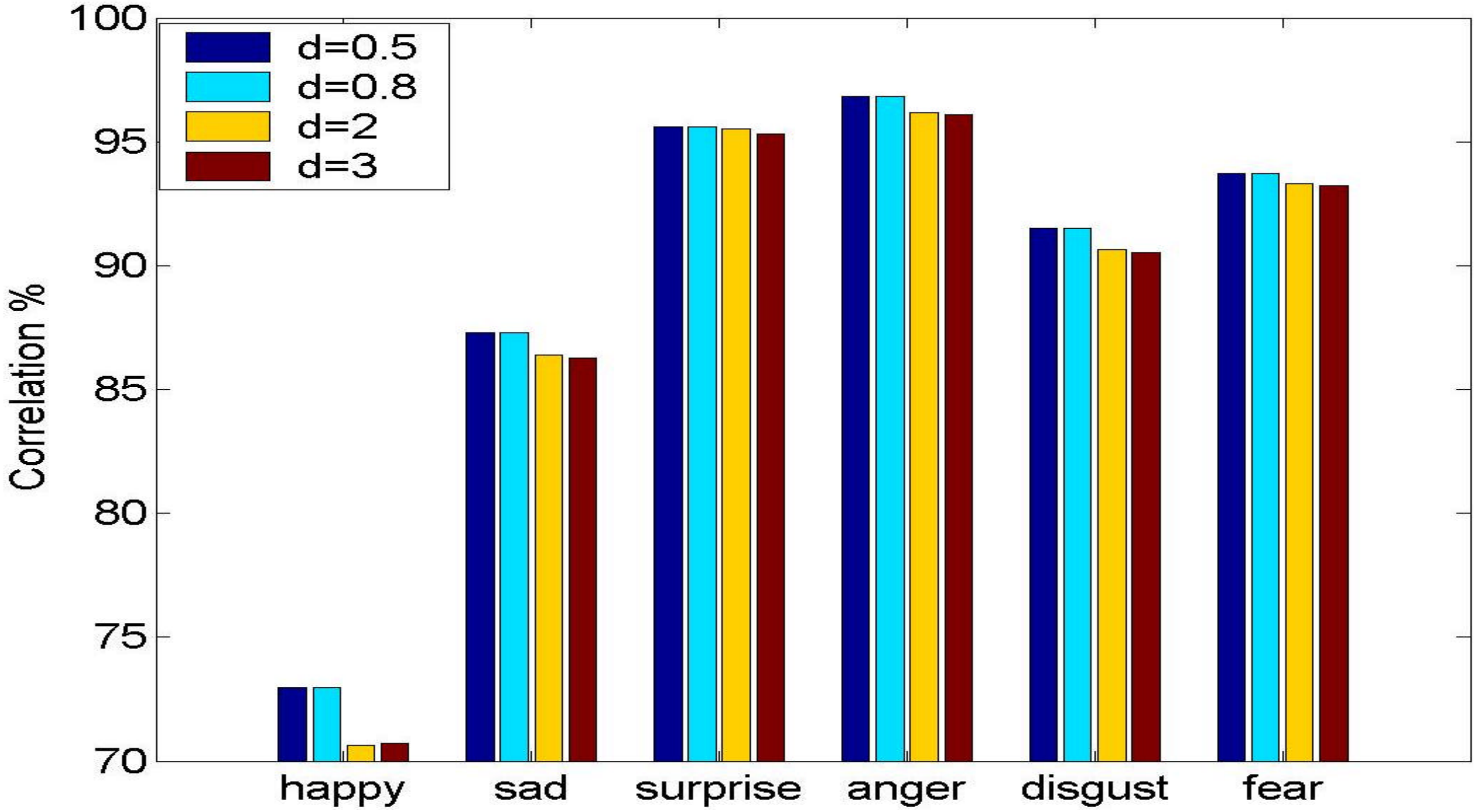
Experimental Results: Facial Expression

Average: PCA : 89%, ICA: 93%



Experimental Results: Facial Expression

Average: 88%～ 89%



Conclusion and discussion

- Identity: KLDA performs the best. LDA>PCA≈KPCA>ICA

- Expression: ICA performs the best. Others performs about the same.

- Training Time: ICA>KLDA>KPCA ≈ LDA>PCA

Conclusion and discussion

- Why ICA is poor in identity, While good in expression?

    Overfit in case 1. Good fit in case 2

- Why d=0.5 performs the best for KPCA in identity recognition?

    Data might be fitted well by sub-gaussian.

Future Work:

Adaptively chose kernels from data!

Questions?