Distance Metric Learning applied to Face Recognition

SEAS, NUM
Suvdaa Batsuuri
2016-10-20

Outline

- Motivation
- Contributions
- Face Recognition
- Distance Metric Learning (DML)
- Incremental Distance Metric Learning
- Experimental Results of Face Recognition
- Conclusions

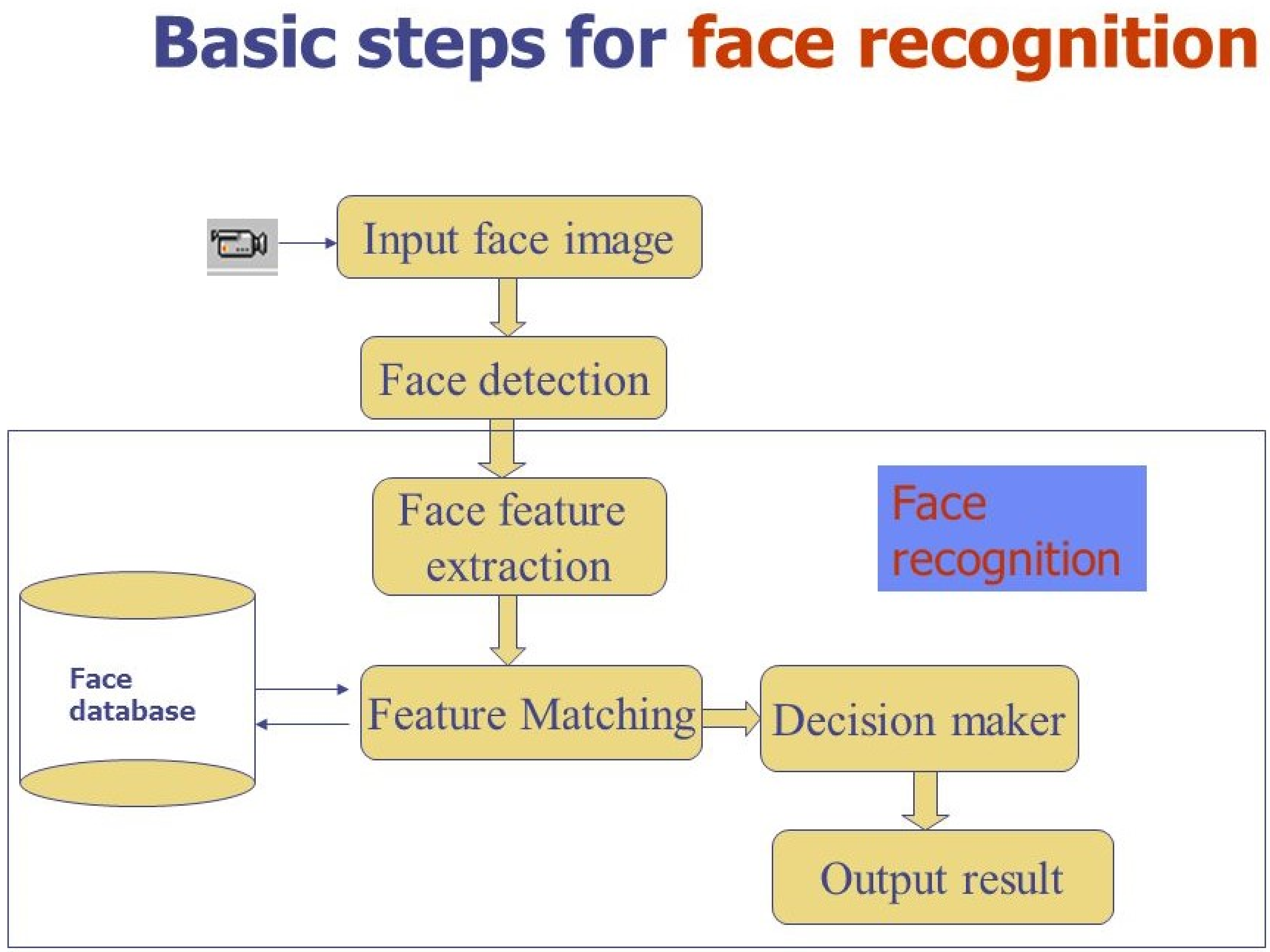
Motivation

- Many machine learning algorithms rely on distance metrics. For example, kNN
- DML improves the performance of the algorithms
- Each DML method has its pros. and cons.
- Therefore, we aimed to find out the most proper DML method for face recognition

Contributions

- Review the state-of –the-art DML methods
- Empirically find out the most suitable DML method for face recognition
- Review the incremental learning methods

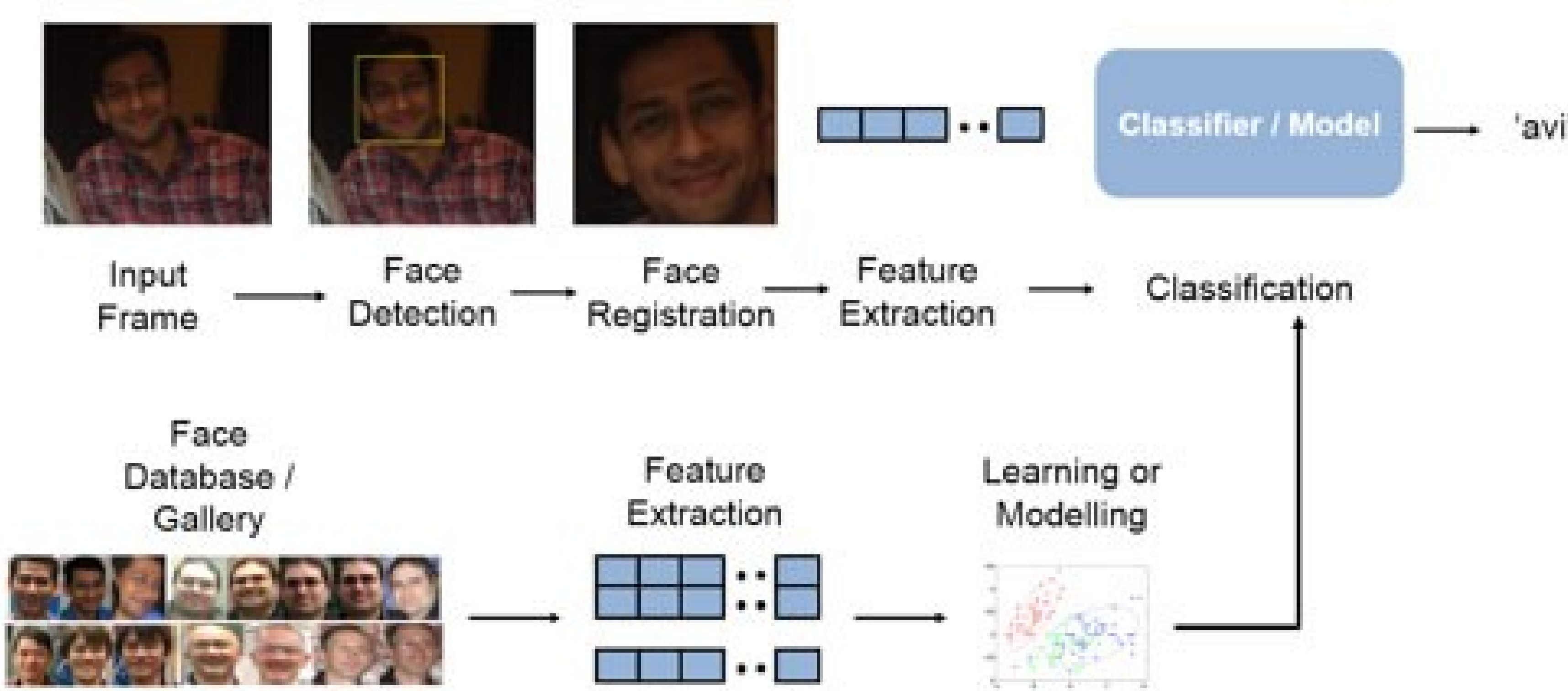
What is Face Recognition?



http://images.slideplayer.com/16/4978543/slides/slide_6.jpg

What is Face Recognition?

- Face recognition steps:



https://www.mathworks.com/cmsimages/98565_wm_face-recognition.jpg

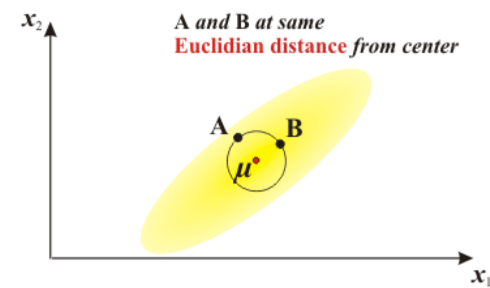
What is Distance Metric Learning?

- How to measure the distance between two vectors?
  - often measured using the Euclidean distance.
- Goal of distance metric learning:
  - to identify an appropriate distance metric that brings “similar” objects close together while separating “dissimilar” objects.
- Distance satisfies :
  - non-negativity, identity, symmetry, triangle inequality
- Recent researches have shown that
  - using a more appropriate distance metric can improve the performances significantly.

Formulation of DML

– We formulate squared Euclidean distance function *(d)* between the two vectors:

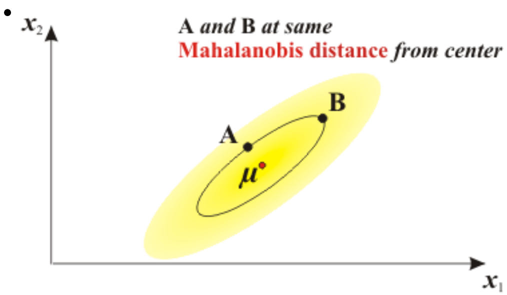
$$d(\mathbf{x}_1, \mathbf{x}_2) = \|\mathbf{x}_1 - \mathbf{x}_2\|^2 = (\mathbf{x}_1 - \mathbf{x}_2)^T(\mathbf{x}_1 - \mathbf{x}_2)$$



– Let covariance matrix is

$$\Sigma = \sum_{i,j} (\mathbf{x}_i - \mu)(\mathbf{x}_j - \mu)^T$$

– , where μ is mean vector. Then original Mahalanobis distance function *(d_M)* become as follow:



$$d_M(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1 - \mathbf{x}_2)^T \Sigma^{-1} (\mathbf{x}_1 - \mathbf{x}_2)$$

– In general, the distance function is:

$$\boxed{d_A(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}} \mathbf{A} (\mathbf{x}_1 - \mathbf{x}_2)}$$

where **A** is a positive semi-definite (PSD) matrix (*d x d*).

Another view of DML

- If A is not PSD, then d_A could be negative.
- In practical, the set of PSD matrices is a convex set.
- It can view as the squared Euclidean distance after applying a linear transformation.
- Decompose $\mathbf{A} = \mathbf{G}^T\mathbf{G}$ via we have

$$
\begin{aligned}
d_A(x_1, x_2) &= (x_1 - x_2)^T A (x_1 - x_2) = (x_1 - x_2) G^T G (x_1 - x_2) \\
&= (Gx_1 - Gx_2)^T (Gx_1 - Gx_2) = \|Gx_1 - Gx_2\|_2^2
\end{aligned}
$$

Categorization of DML

- Depending on the availability of the training examples, DML algorithms can be divided into two categories:
  - supervised DML
  - unsupervised DML

Supervised DML methods

- The supervised DML methods use labels information of data and they are divided into 2 categories:
  - Local
  - Global
- The global DML methods try to satisfy all the constraints simultaneously. They keep all the data points the same classes close, while separating all the data points from different classes.
- The local DML methods try to satisfy the constraints in a local region around each data point instead of all pairwise constraints.

Unsupervised DML methods

Unsupervised DML (manifold learning)

- learns an underlying low-dimensional manifold where geometric relationships (e.g. distance) between most of the observed data are preserved.
- Popular methods:
  - PCA, MDS, LLE, Isomap and so forth.

State-of-the-art DML methods

- Principal Component Analysis (PCA)
  - Euclidean distance
  - Mahalanobis distance
- Neighborhood Component Analysis (NCA)
  - https://papers.nips.cc/paper/2566-neighbourhood-components-analysis.pdf
- The Large Margin Nearest Neighbor (LMNN)
  - http://jmlr.csail.mit.edu/papers/volume10/weinberger09a/weinberger09a.pdf
  - http://www.cs.cornell.edu/~kilian/code/lmnn/lmnn.html
- Energy Classifier
  - http://yann.lecun.com/exdb/publis/pdf/lecun-icdar-keynote-07.pdf

Principal Component Analysis

- An unsupervised, global and linear DML
- Learn transformation matrix by maximizing the variance
- Also useful for dimension reduction

$$\max_{\mathbf{L}}(\mathbf{L}^{\mathrm{T}}\mathbf{S}_{\mathrm{T}}\mathbf{L}) \text{ subject to: } \mathbf{L}\mathbf{L}^{\mathrm{T}} = \mathbf{I}.$$

Neighborhood Component Analysis

- A supervised, local and linear DML method that learns a Mahalanobis distance metric for KNN by maximizing the leave-one-out cross validation.

The probability of classifying xi correctly, $p_i = \sum_{j \in C_i} p_{ij}$

weighted counting involving pairwise distance

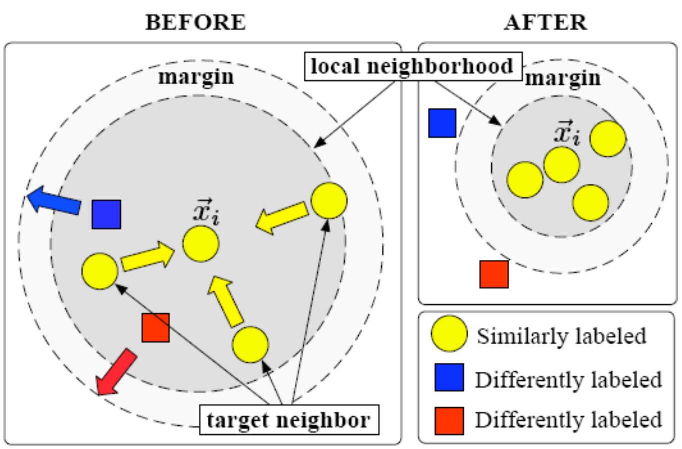
$$p_{ij} = \frac{\exp\left(-\left\|\mathbf{Ax_i} - \mathbf{Ax_j}\right\|^2\right)}{\sum_{k \neq i} \exp\left(-\left\|\mathbf{Ax_i} - \mathbf{Ax_k}\right\|^2\right)} \qquad p_{ii} = 0$$

The expected number of correctly classification points:

$$f(A) = \sum_{i=1}^{n} p_i,$$

Large Margin Nearest Neighbors

- A supervised, local and linear DML method that learns the Mahalanobis distance metric to maximize the margin between the classes for KNN classifier



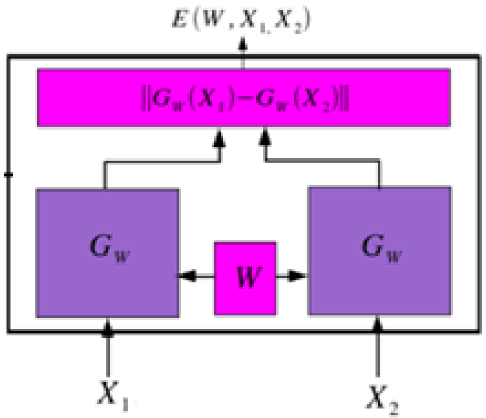
$$\varepsilon(L) = \sum_{ij} \eta_{ij} \left\| L(\vec{x}_i - \vec{x}_j) \right\|^2 + c \sum_{ijl} \eta_{ij} (1 - y_{il}) \left[1 + \left\| L(\vec{x}_i - \vec{x}_j) \right\|^2 - \left\| L(\vec{x}_i - \vec{x}_l) \right\|^2 \right]_+$$

Energy classifier

- Energy classifier learns a function that maps input patterns into a target space using L1 norm, based on the pairs of the faces of same or different persons.

$$E_W(X_1, X_2) = \|G_W(X_1) - G_W(X_2)\|$$

$$\begin{aligned} L\left(W, (Y, X_1, X_2)^i\right) \\ &= (1 - Y)L_G\left(E_W(X_1, X_2)^i\right) \\ &+ YL_I\left(E_W(X_1, X_2)^i\right) \end{aligned}$$



Incremental DML methods

Why is incremental DML method needed?

- Real applications, constraints are only available incrementally, thus necessitating methods that can perform online updates to the learned metric.
- Small sample size problem
- Memory and time complexity

What is problem?

- How to update the distance metric (transformation matrix)

Recent status of Incremental Learning

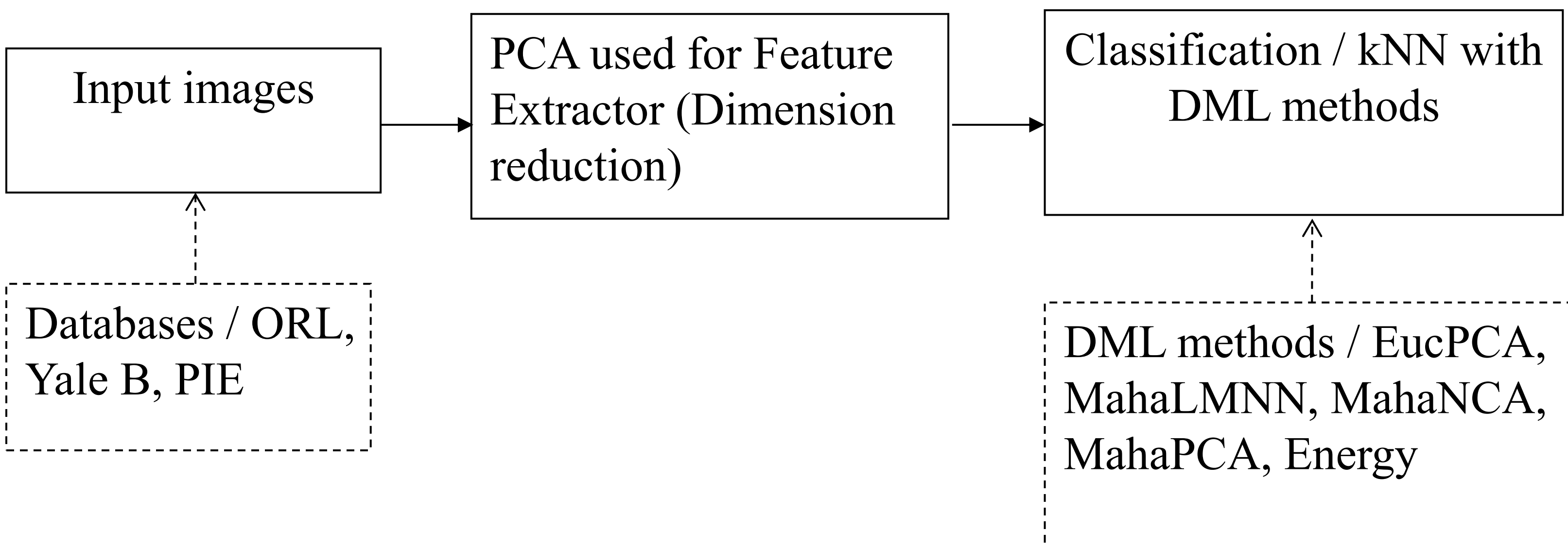
Some incremental versions of DML:

- PCA => IPCA
- SVD => R-SVD
- NCA => no incremental version
- LMNN => no incremental version

Experimental Results

Face Recognition process

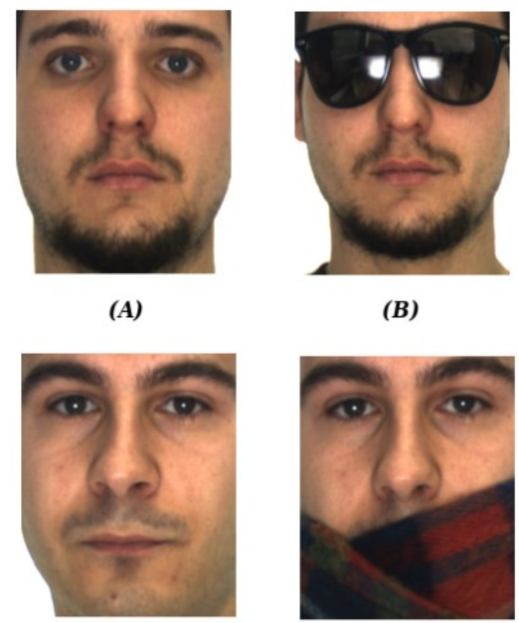
- General scheme

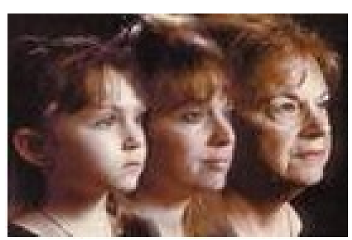


http://www.face-rec.org/databases/

Face Recognition

- Problems
  - Pose, Illumination, Expression, Occlusion, aging and so on.



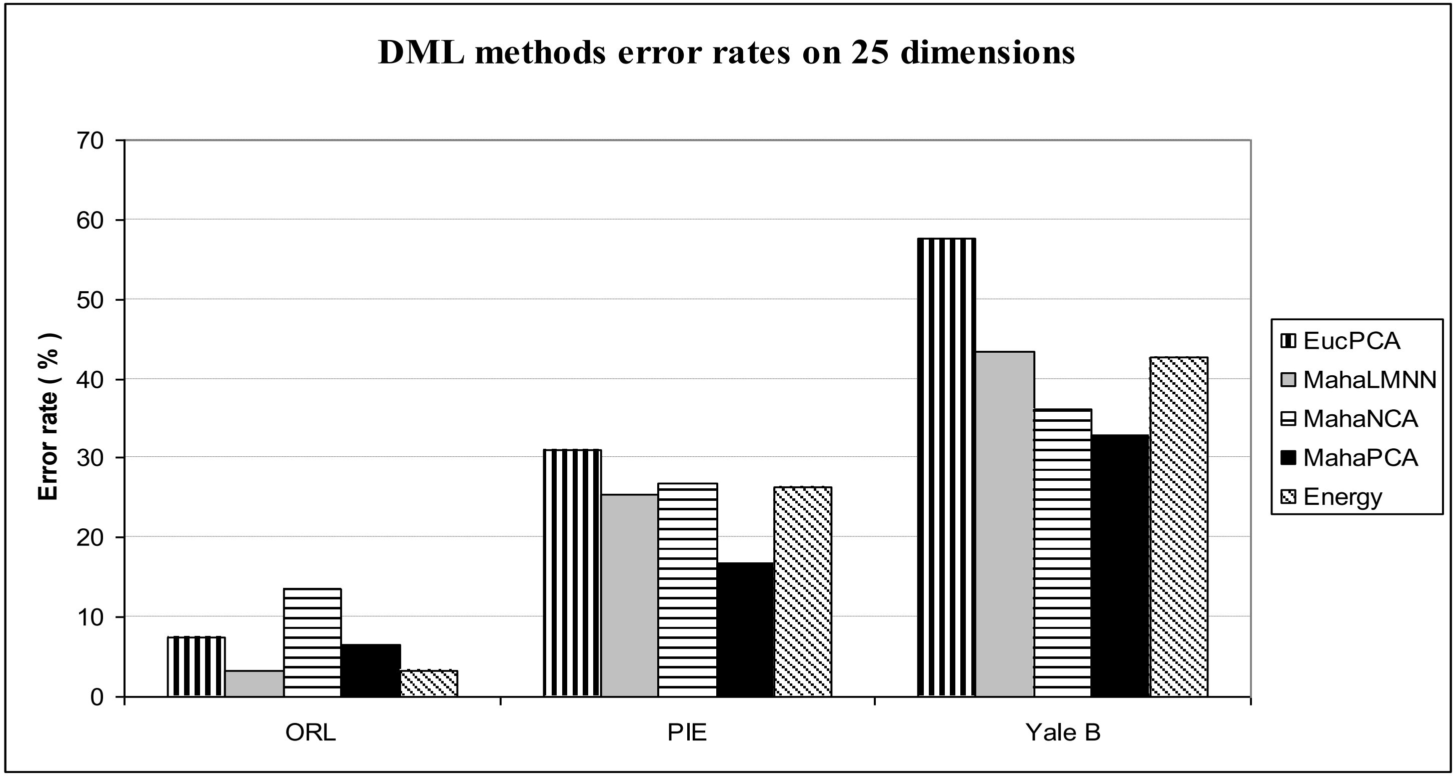


Face Database

- Tested databases: ORL, PIE and Yale B

Database	# of classes	Variations	Image size	Used dimension	# of training images	# of test images
ORL	40	Various	112x92	35	280	120
PIE	15	Pose, Illumination, Exp	50x50	35	1260	522
Yale B	38	Pose, Illumination	54x54	35	956	412

Comparison in same dimension



Result ordered by recognition rates:

MahaPCA > MahaNCA > Energy ≈ MahaLMNN > EucPCA

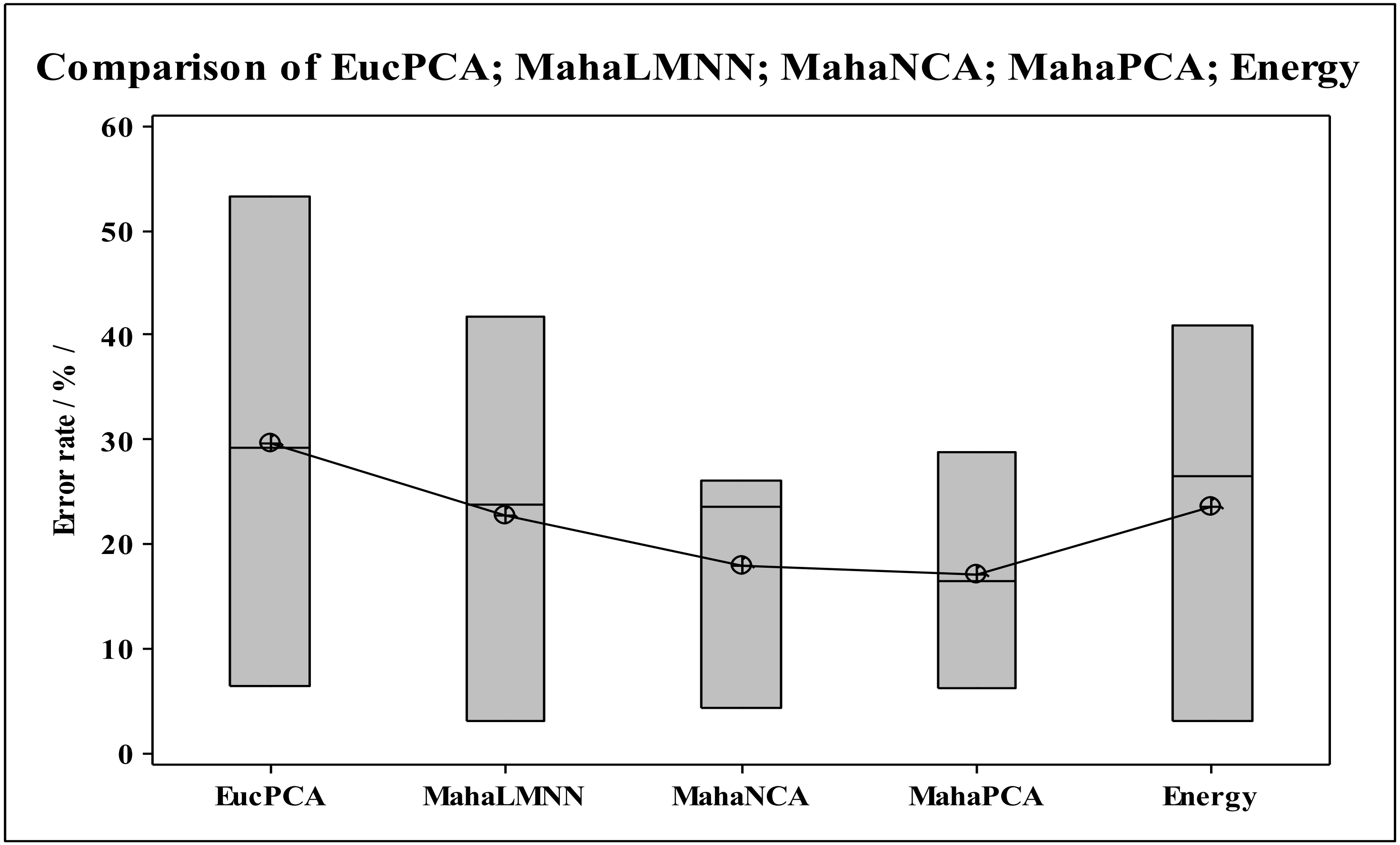
Comparison of the best performances

Methods	ORL (dim)	PIE (dim)	Yale B (dim)
EucPCA	6,58±2,09 (30)	29.31±15.54(30)	53,24±5,67 (30)
MahaLMNN	3,08±1,24 (30)	23.74±11.86(30)	41,57±17,73 (30)
MahaNCA	4,41±2,43 (10)	26.15±10.45(30)	23,59±9,79 (20)
MahaPCA	6,25±1,67 (20)	16.48±7.82(30)	28,83±7,24 (20)
Energy	3,08±1,11 (30)	26.64±11.47(30)	40,99±17,68 (30)

Result ordered by recognition rates:

MahaNCA >MahaPCA > MahaLMNN ≈ Energy > EucPCA

Comparison of the best performances

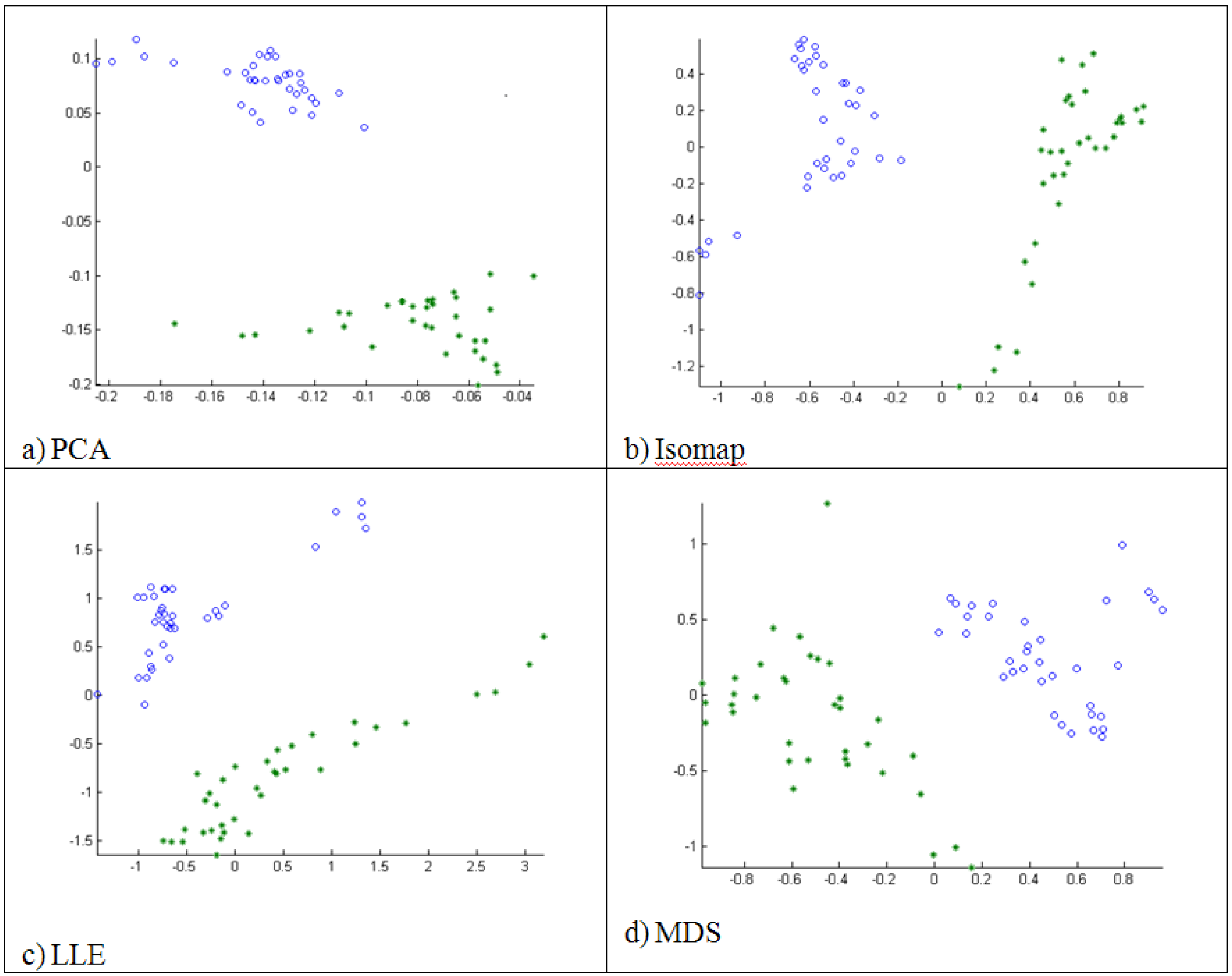


Result ordered by recognition rates:

MahaNCA >MahaPCA > MahaLMNN ≈ Energy > EucPCA

Unsupervised DML results

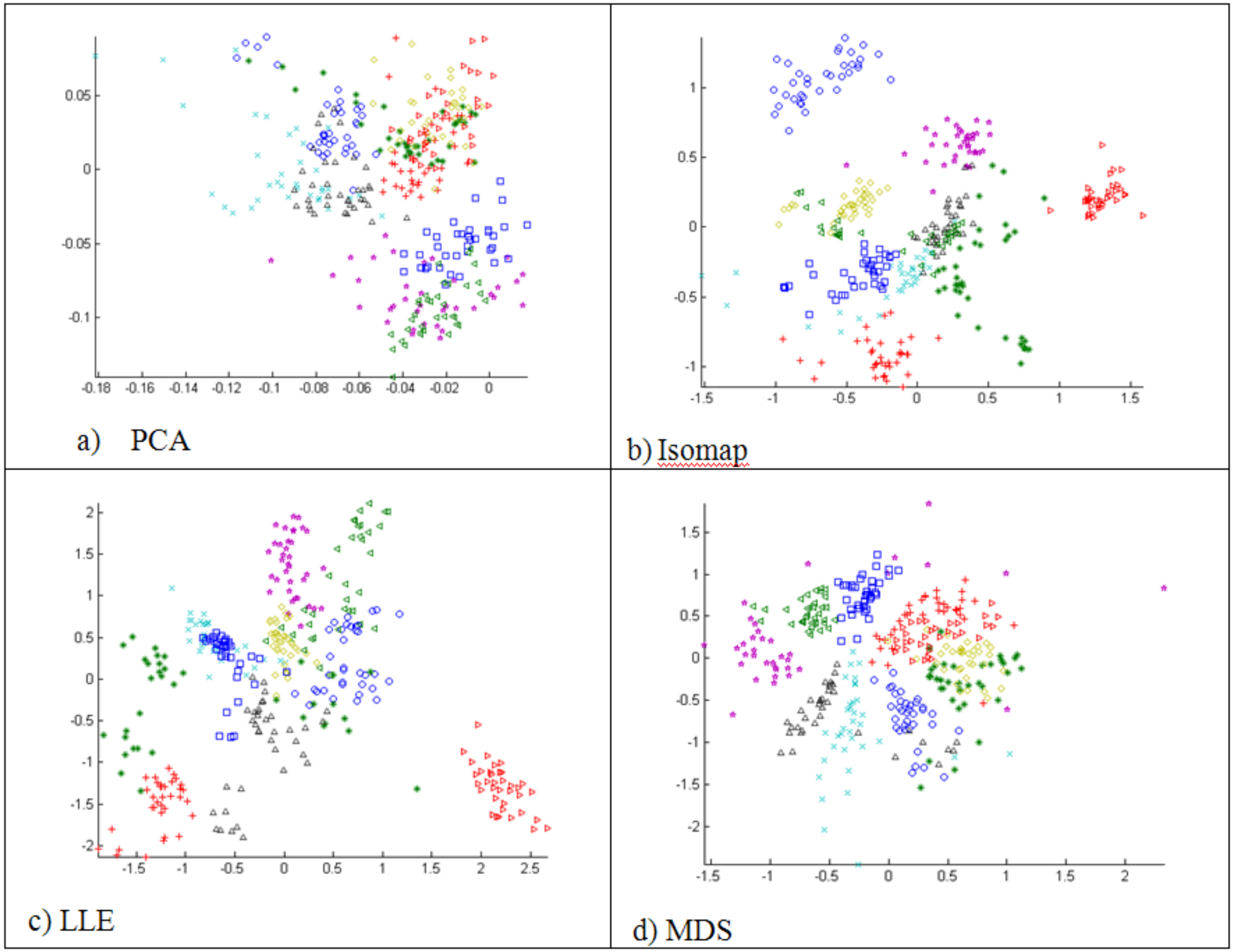
Projection result of 2-people (Yale B database)



a) PCA b) Isomap c) LLE d) MDS

Unsupervised DML results

Projection result of 10-people (Yale B database)



Conclusions and discussions

- Mahalanobis based PCA is still competitive in face recognition on our used databases.
- LLE and Isomap are projected the data more separable, and LLE and PCA were the fastest.
- Categorized the incremental DML methods into 4 categories according to their updating methods.
- Our future work is to design incremental DML for face recognition. For example, to decide updating rule for transformation matrix for incremental NCA and Incremental LMNN.

Thanks for your attention