Machine Learning for Biometrics

Dong XU

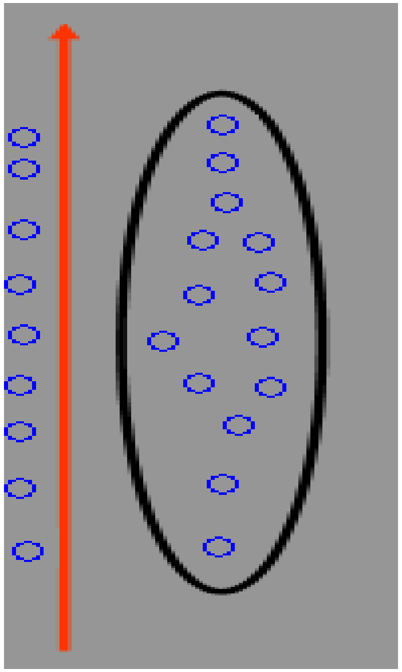
School of Electrical and Information Engineering

University of Syndey

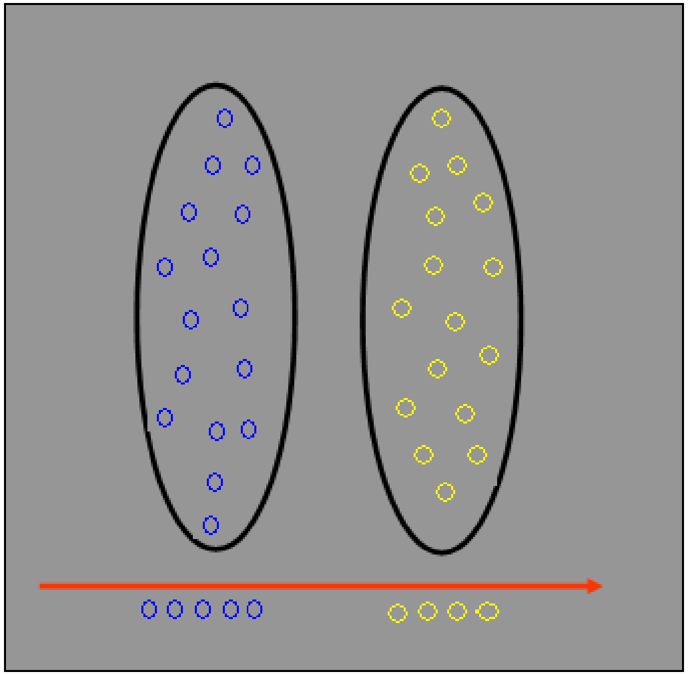
Outline

- ✓ Dimensionality Reduction for Tensor-based Objects
- ✓ Graph Embedding: A General Framework for Dimensionality Reduction
- ✓ Learning using Privileged Information for Face Verification and Person Re-identification

What is Dimensionality Reduction?



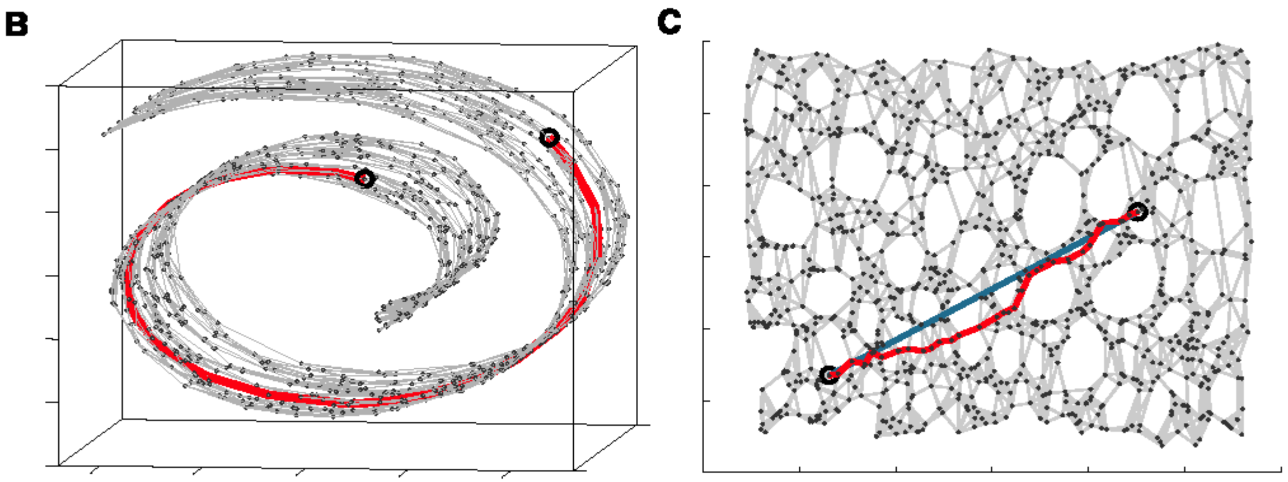
PCA



LDA

Examples: 2D space to 1D space

What is Dimensionality Reduction?

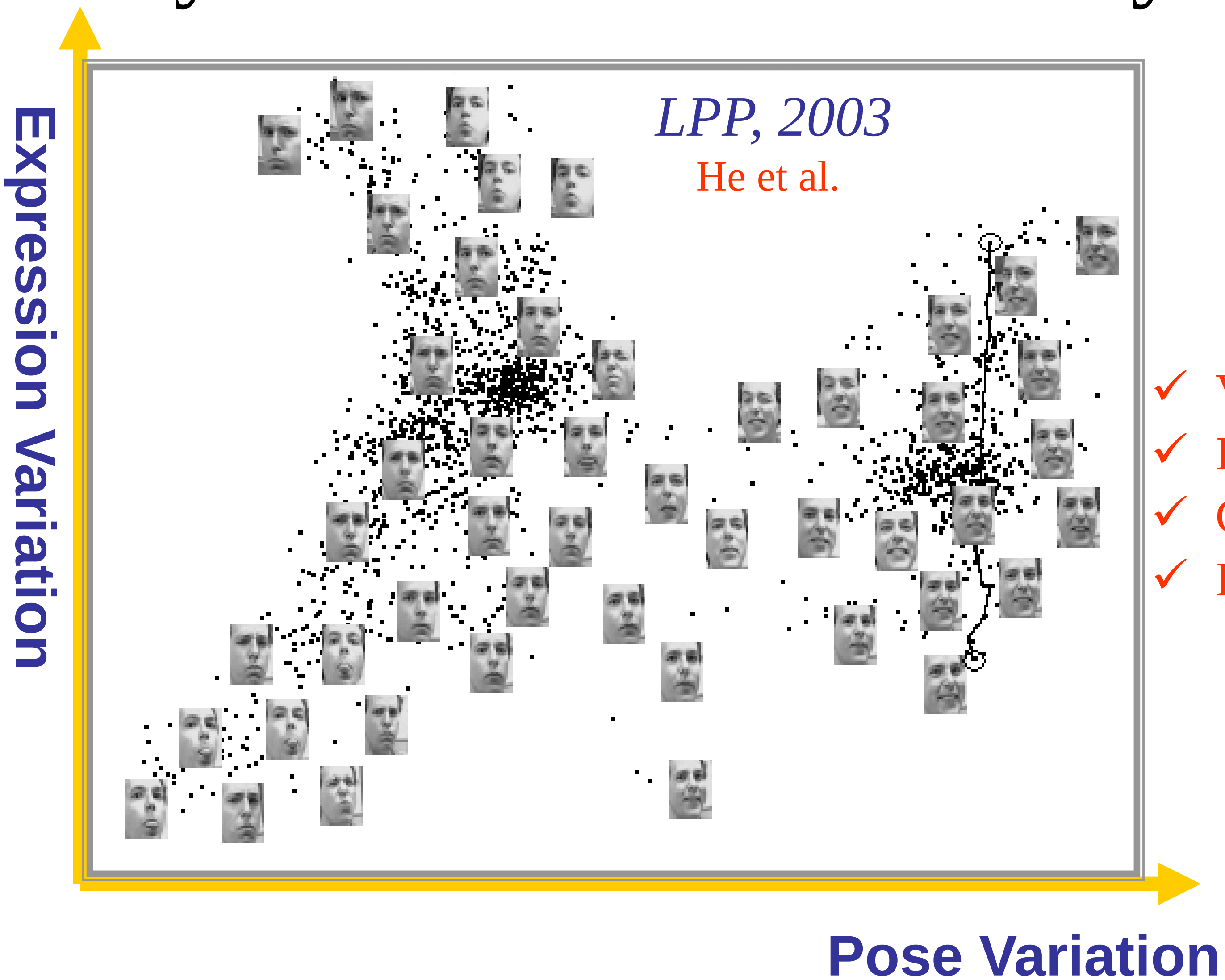


Example: 3D space to 2D space

ISOMAP: Geodesic Distance Preserving

J. Tenenbaum et al., 2000

Why Conduct Dimensionality Reduction?



- ✓ Visualization
- ✓ Feature Extraction
- ✓ Computation Efficiency
- ✓ Broad Applications
  - Face Recognition
  - Human Gait Recognition
  - CBIR

Uncover intrinsic structure

Outline

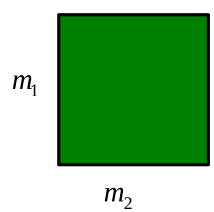
- ✓ Dimensionality Reduction for Tensor-based Objects
- ✓ Graph Embedding: A General Framework for Dimensionality Reduction
- ✓ Learning using Privileged Information for Face Verification and Person Re-identification

What is Tensor?

Tensors are arrays of numbers which transform in certain ways under coordinate transformations.

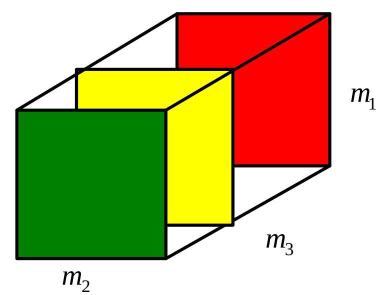
$x \in \mathrm{R}^{m_1}$

Vector



$X \in \mathrm{R}^{m_1 \times m_2}$

Matrix

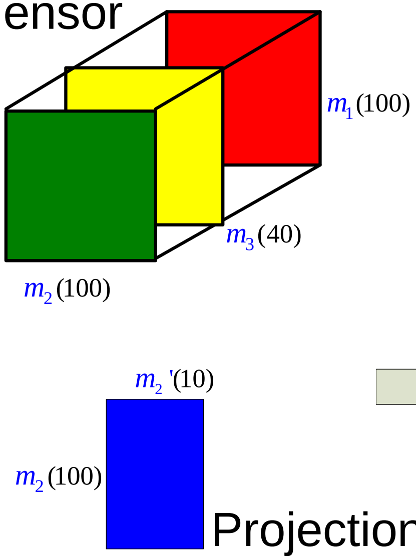


$\mathbf{X} \in \mathrm{R}^{m_1 \times m_2 \times m_3}$

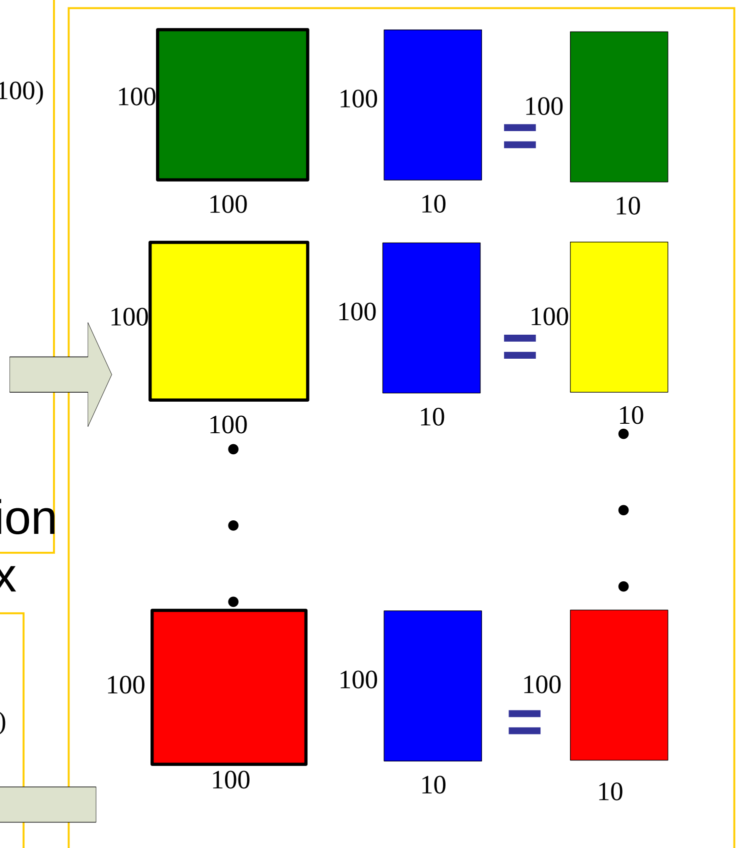
3rd-order Tensor

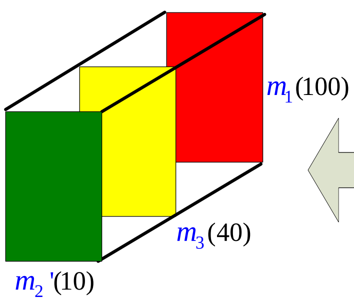
Definition of Mode-k Product

Original Tensor



Projection Matrix





New Tensor

Projection:

high-dimensional space

-> low-dimensional space

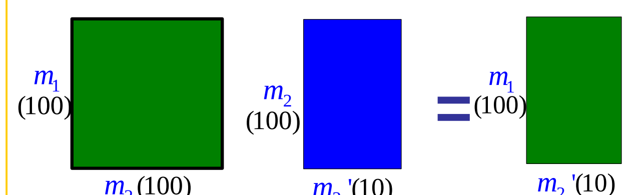
Reconstruction:

low-dimensional space

-> high-dimensional space

Product for two Matrices

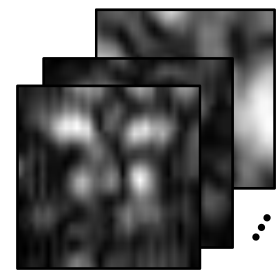
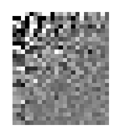
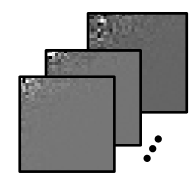
$$Y = XU \qquad Y_{ij} = \sum_{k=1}^{m_2} X_{ik} U_{kj}$$



Original Matrix | Projection Matrix | New Matrix

Notation: $\mathbf{Y} = \mathbf{X} \boxed{\times_k} U$

Data Representation in Dimensionality Reduction

	Vector	*Matrix*	*3rd-order Tensor*	
		Gray-level Image	*Filtered Image*	*Video Sequence*
High Dimension				
Low Dimension				

Examples

PCA, LDA

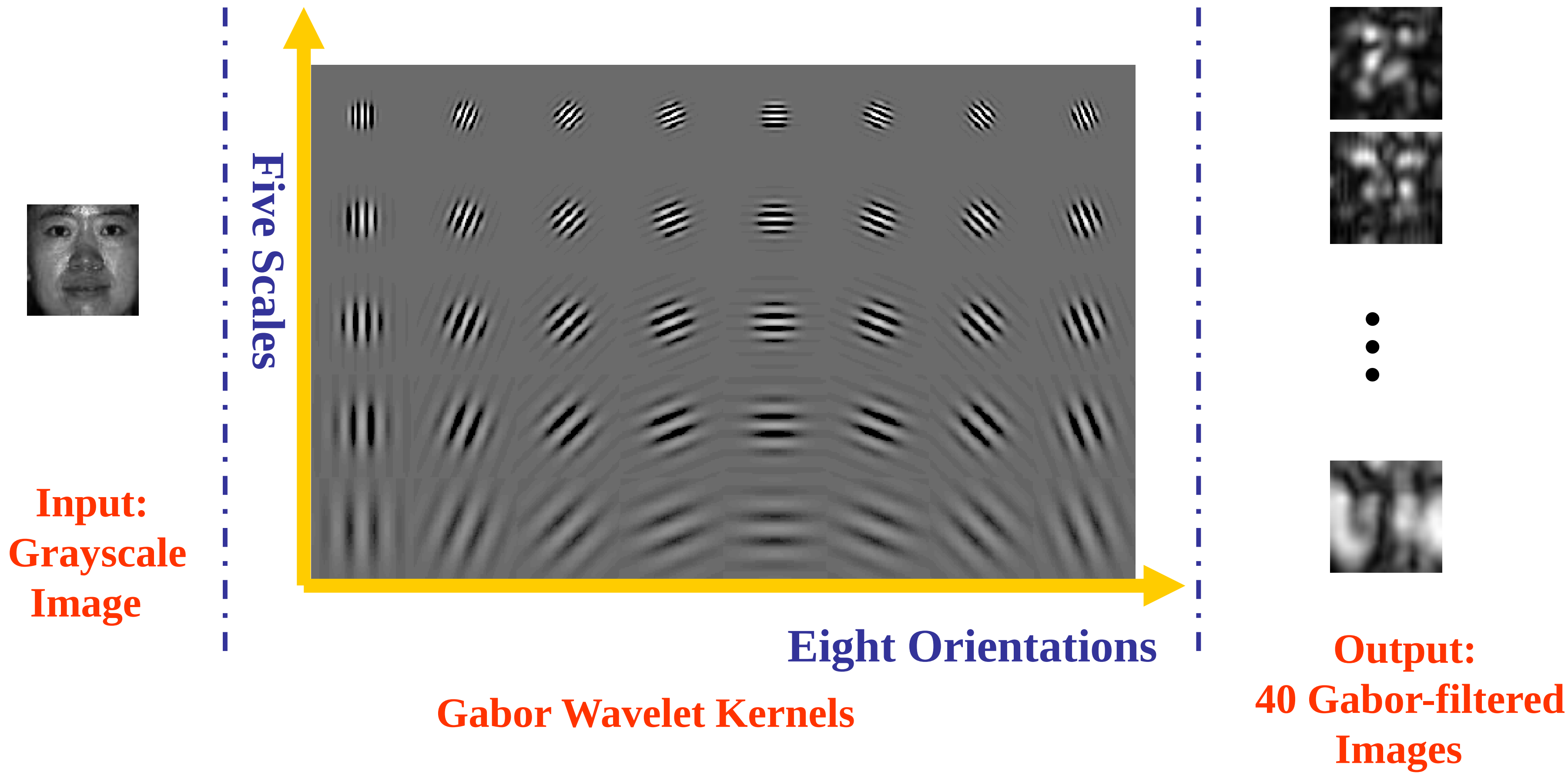
Tensorface, 2002
M. Vasilescu and
D. Terzopoulos

Rank-1
Decomposition, 2001
A. Shashua
and A. Levin
Low rank approximation
of matrix
J. Ye

Our Work
Xu et al., 2005
Yan et al., 2005

What is Gabor Features?

Gabor features can improve recognition performance in comparison to grayscale features. *C Liu and H Wechsler, T-IP, 2002*



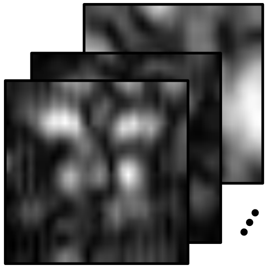
Why Represent Objects as *Tensors* instead of Vectors?

- *Natural Representation*

  Gray-level Images (2D structure)

  Videos (3D structure)

  Gabor-filtered Images (3D structure)



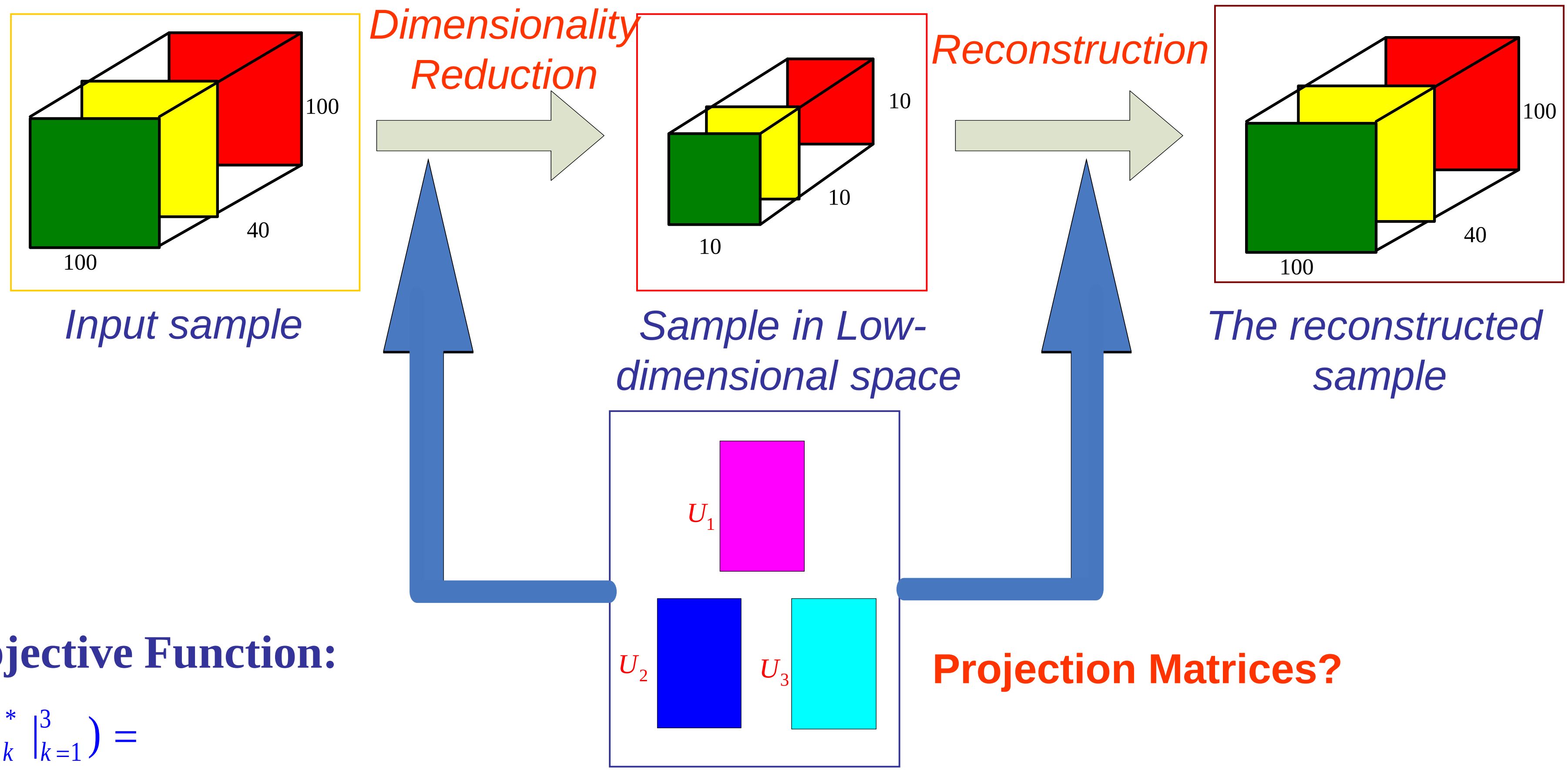
- *Enhance Learnability in Real Application*

  Curse of Dimensionality (*Gabor-filtered image*: 100*100*40 -> *Vector*: *400,000*)

  Small sample size problem

- *Reduce Computation Cost*

Concurrent Subspace Analysis as an Example (Criterion: *Optimal Reconstruction*)



Objective Function:

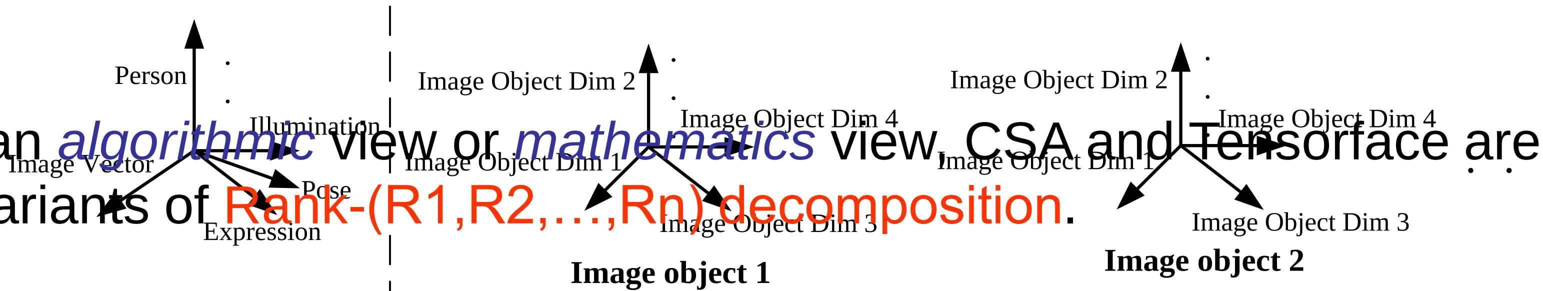
$$(U_k^* |_{k=1}^3) = \arg\min_{U_k|_{k=1}^3} \sum_i \| \mathbf{X}_i \times_1 U_1 U_1' \ldots \times_3 U_3 U_3' - \mathbf{X}_i \|^2$$

D. Xu, S. Yan, H. Zhang and et al., *CVPR, 2005*

Connection to Previous Work –*Tensorface*

(M. Vasilescu and D. Terzopoulos, 2002)

From an *algorithmic* view or *mathematics* view, CSA and Tensorface are both variants of Rank-(R1,R2,…,Rn) decomposition.

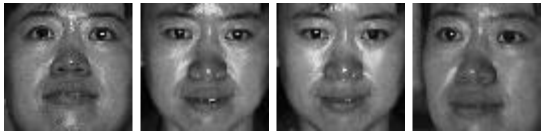


(a) Tensorface

(b) CSA

	Tensorface	CSA
Motivation	Characterize *external* factors	Characterize internal factors
Input: Gray-level Image	*Vector*	Matrix
Input: Gabor-filtered Image (Video Sequence)	*Not address*	3rd-order tensor
When equal to PCA	The number of images per person are only *one* or are *a prime number*	Never
Number of Images per Person for Training	*Lots of images* per person	One image per person

Experiments: *Database Description*

	Number of Persons (Images per person)	Image Size (Pixels)	Example Images
Simulated Video Sequence	60 (1)	64×64×13	
ORL database	40 (10)	56×46	
CMU PIE-1 sub-database	60 (10)	64×64	
CMU PIE-2 sub-databases	60 (10)	64×64	

Experiments: *Object Reconstruction (1)*

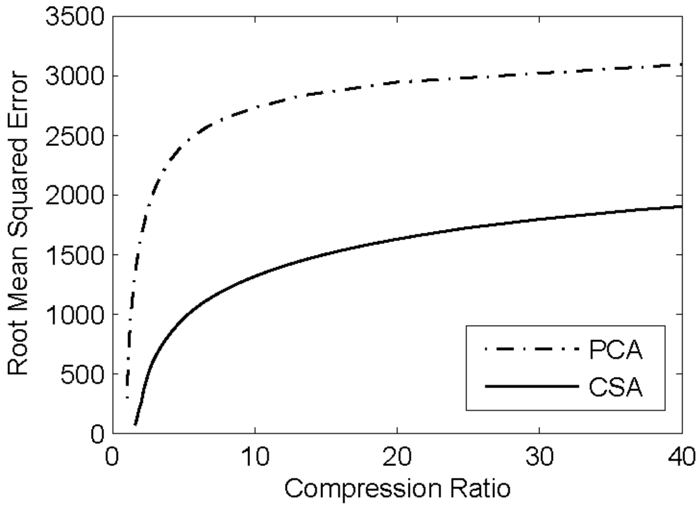
Input: Gabor-filtered images

ORL database

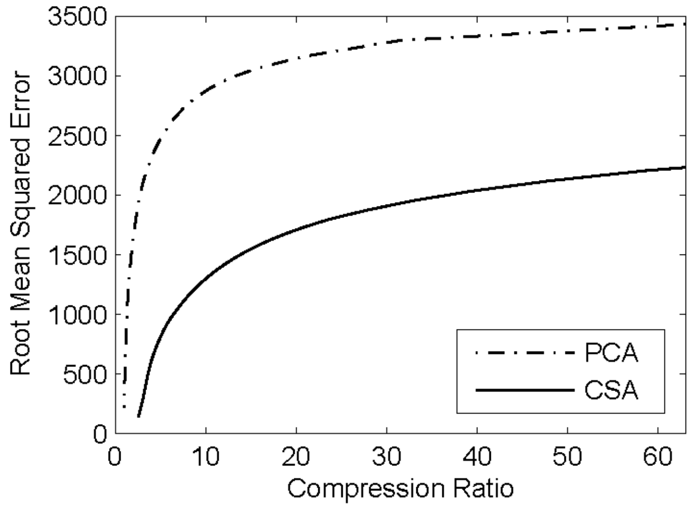
CMU PIE-1 database

Objective Evaluation Criterion:

Root Mean Squared Error (**RMSE**) and Compression Ratio (**CR**)



ORL database



CMU PIE-1 database

Experiments: *Object Reconstruction (2)*

Input: Simulated video sequence

Original Images

Reconstructed Images from PCA

Reconstructed Images from CSA

Experiments: *Face Recognition*

Input: Gray-level images and Gabor-filtered images

ORL database

CMU PIE database

Algorithm	CMU PIE-1	CMU PIE-2	ORL
PCA (Gray-level feature)	70.1%	28.3%	76.9%
PCA (Gabor feature)	80.1%	42.0%	86.6%
CSA (Ours)	***90.5%***	***59.4 %***	***94.4%***

Summary

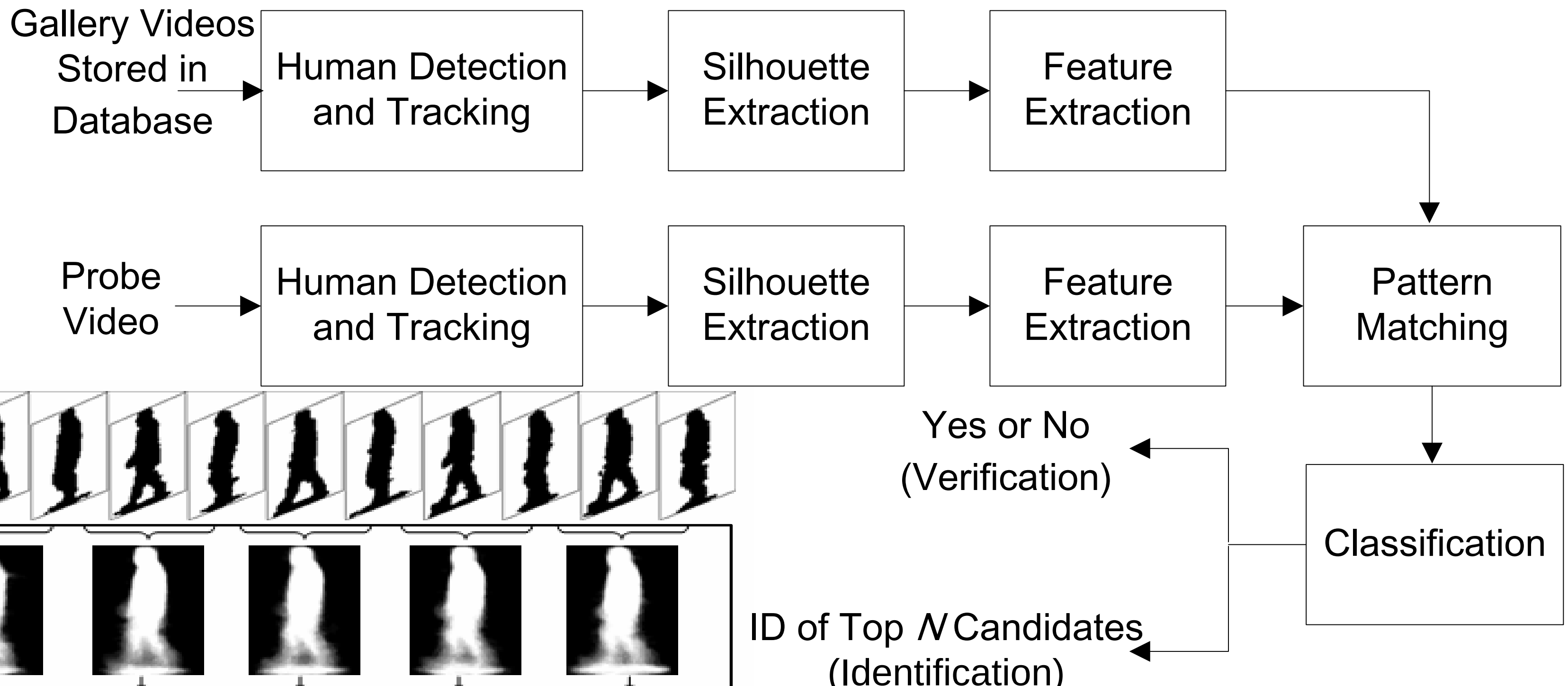
- This is the first work to address dimensionality reduction with a tensor representation of arbitrary order.
- Opens a new research direction.

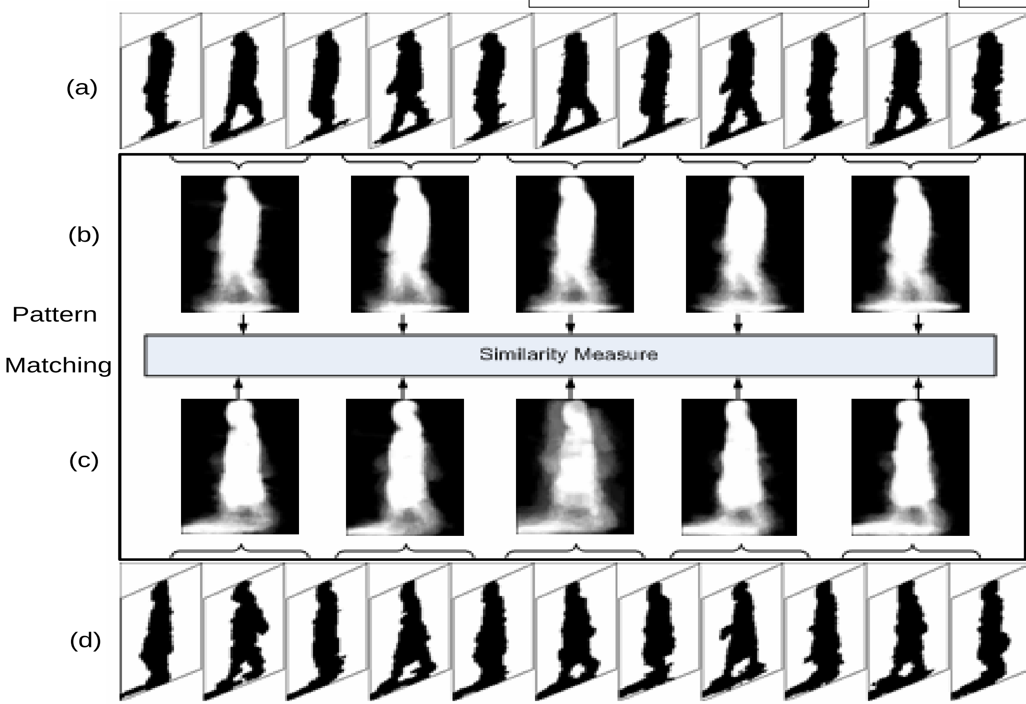
Bilinear and Tensor Subspace Learning (New Research Direction)

- Concurrent Subspace Analysis (CSA), *CVPR 2005 and T-CSVT 2008*
- Discriminant Analysis with Tensor Representation (DATER): *CVPR 2005 and T-IP 2007*
- Rank-one Projections with Adaptive Margins (RPAM): *CVPR 2006 and T-SMC-B 2007*
- Enhancing Tensor Subspace Learning by Element Rearrangement: *CVPR 2007 and T-PAMI 2009*
- Discriminant Locally Linear Embedding with High Order Tensor Data (DLLE/T): *T-SMC-B 2008*
- Convergent 2D Subspace Learning with Null Space Analysis (NS2DLDA)*: T-CSVT 2008*
- Semi-supervised Bilinear Subspace Learning*: T-IP 2009*
- Applications in Human Gait Recognition
  - CSA+DATER: *T-CSVT 2006*
  - Tensor Marginal Fisher Analysis (TMFA): *T-IP 2007*

Other researchers also published several papers along this direction!!!

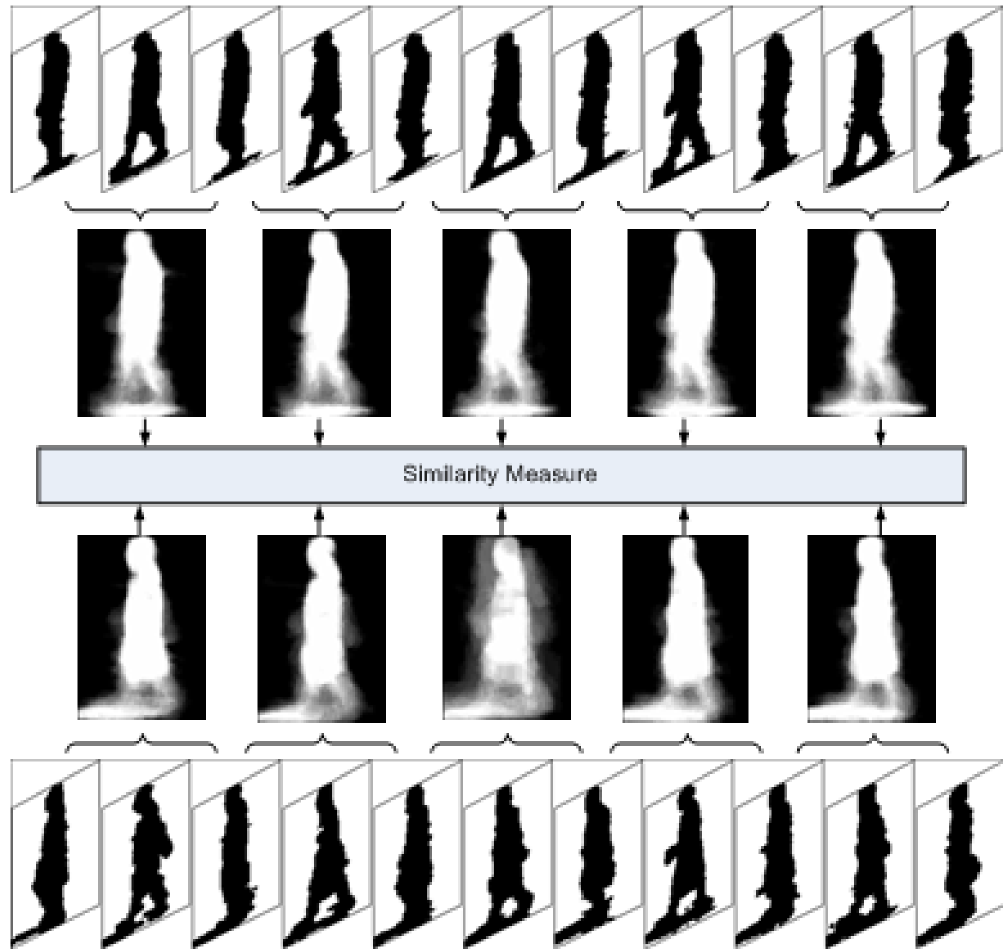
Human Gait Recognition: Basic Modules





(a) (d): The extracted silhouettes from one probe and gallery video;
(b) (c): The gray-level Gait Energy Images (GEI).

Human Gait Recognition with *Matrix Representation*



D. Xu, S. Yan, H. Zhang and et al., *T-CSVT, 2006*

USF HumanID

Experiment (Probe)	# of Probe Sets	Difference between Gallery and Probe Set
A (G, A, L, NB, M/N)	122	View
B (G, B, R, NB, M/N)	54	Shoe
C (G, B, L, NB, M/N)	54	View and Shoe
D (C, A, R, NB, M/N)	121	Surface
E (C, B, R, NB, M/N)	60	Surface and Shoe
F (C, A, L, NB, M/N)	121	Surface and View
G (C, B, L, NB, M/N)	60	Surface, Shoe, and View
H (G, A, R, BF, M/N)	120	Briefcase
I (G, B, R, BF, M/N)	60	Briefcase and Shoe
J (G, A, L, BF, M/N)	120	Briefcase and View
K (G, A/B, R, NB, N)	33	Time, Shoe, and Clothing
L (C, A/B, R, NB, N)	33	Time, Shoe, Clothing, and Surface

1. Shoe types: A or B; 2. Carrying: with or without a briefcase; 3. Time: May or November; 4. Surface: grass or concrete; 5. Viewpoint: left or right

Human Gait Recognition: Our Contributions

Top ranked results on the benchmark USF HumanID dataset

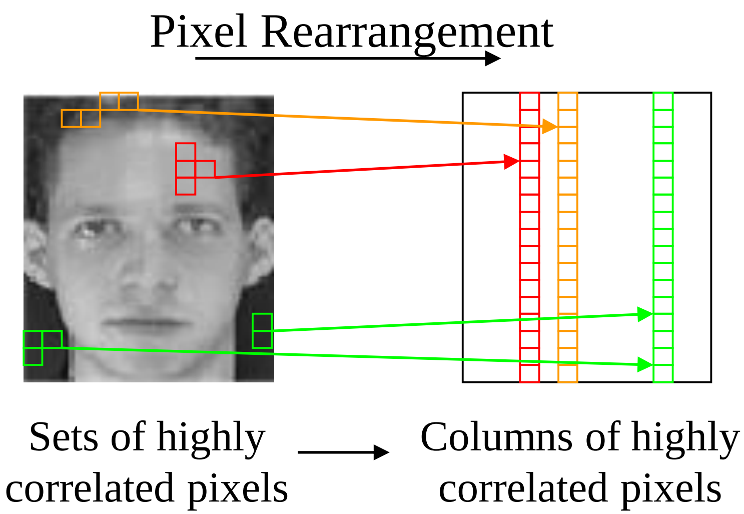
Methods	Average Rank-1 Results (%)
Our Recent Work (Ours, TIP 2012)	70.07
DNGR (Sarkar's group, TPAMI 2006)	62.81
Image-to-Class distance (Ours, TCSVT 2010)	61.19
GTDA (Maybank's group, TPAMI 2007)	60.58
Bilinear Subspace Learning method 2: MMFA (Ours, TIP 2007)	59.9%
Bilinear Subspace Learning method 1: CSA + DATER (Ours, TCSVT 2006)	58.5%
PCA+LDA (Bhanu's group, TPAMI 2006)	57.70%

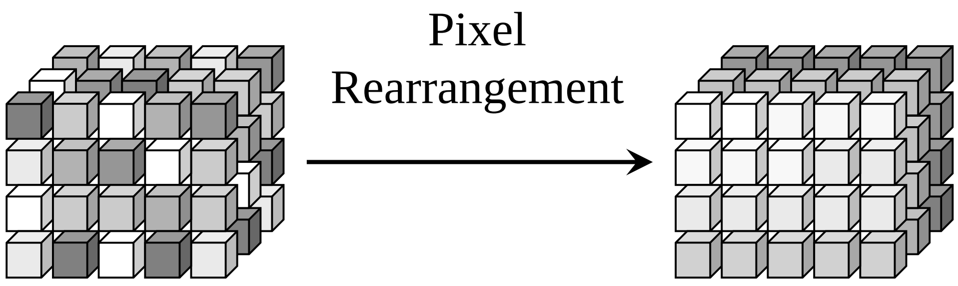
*The DNGR method additionally uses the manually annotated silhouettes, which are not publicly available.

How to Utilize *More* Correlations?

Potential Assumption in Previous Tensor-based Subspace Learning:

Intra-tensor correlations: Correlations among the features within certain tensor dimensions, such as rows, columns and Gabor features…





D. Xu, S. Yan et al., *T-PAMI 2009*

Problem Definition

- The task of enhancing correlation/redundancy among 2^{nd}–order tensor is to search for a pixel rearrangement operator R, such that

$$R^* = \arg\min_R \{ \min_{U,V} \sum_{i=1}^{N} \| X_i^R - UU^T X_i^R V V^T \|^2 \}$$

1. X_i^R is the rearranged matrix from sample X_i
2. The column numbers of U and V are predefined

After pixel rearrangement, we can use the rearranged tensors as input for concurrent subspace analysis

Solution to Pixel Rearrangement Problem

Initialize U_0, V_0

Compute reconstructed matrices

$$X_{i,n}^{Rec} = U_{n-1}U_{n-1}^T X_i^{R_{n-1}} V_{n-1}V_{n-1}^T$$

Optimize operator ***R***

$$R_n = \arg\min_R \sum_{i=1}^{N} \| X_i^R - X_{i,n}^{Rec} \|^2$$

Optimize ***U*** **and** ***V***

$$(U_n, V_n) = \arg\min_{U,V} \sum_{i=1}^{N} \| X_i^{R_n} - UU^T X_i^{R_n} VV^T \|^2$$

n=n+1

$$Note: \sum_{i=1}^{N} \| X_i^{R_n} - X_{i,n}^{Rec} \|^2 \geq \sum_{i=1}^{N} \| X_i^{R_n} - U_{n-1}U_{n-1}^T X_i^{R_n} V_{n-1}V_{n-1}^T \|^2$$

How to optimize R

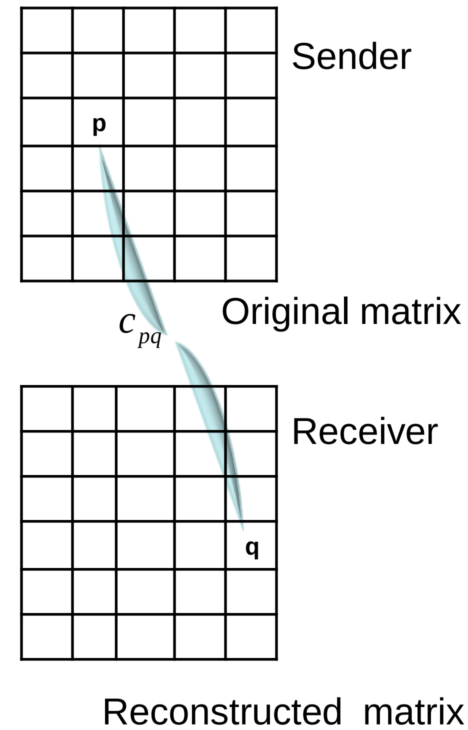
- It is an integer programming problem

$$R^{*} = \arg\min_{R} \sum_{i=1}^{N} \| X_i^{R} - X_{i,n}^{Rec} \|^2$$

$$\min_{R} \sum_{p,q} c_{pq} R_{pq} \quad st.$$

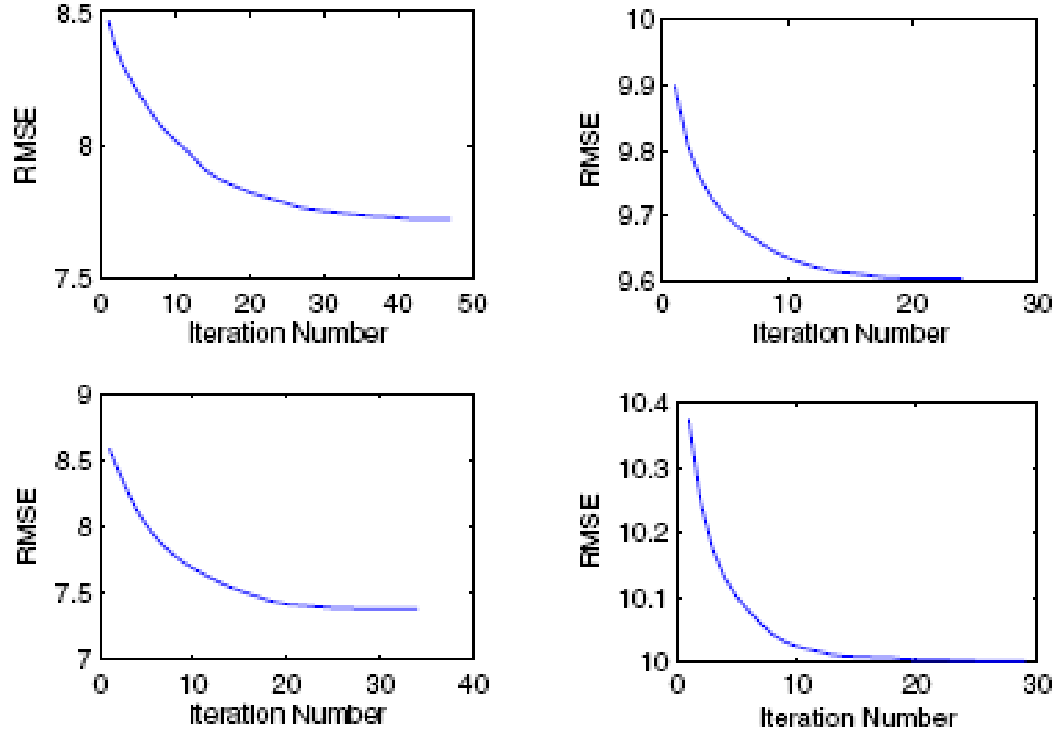
$$1: 0 \le R_{pq} \le 1; \quad 2: \sum_{p} R_{pq} = 1; \; 3: \sum_{q} R_{pq} = 1$$

$$where \;\; c_{pq} = \sum_{i=1}^{N} | X_i(p) - X_{i,n}^{Rec}(q) |^2$$



1. Linear programming problem has integer solution.
2. We constrain the rearrangement within local neighborhood for speedup.

Convergence Speed



2DPCA-ER GLRAM-ER

Figure 2. Algorithm convergence with element rearrangement. Root Mean Squared Error (RMSE) vs. number of iterations on the CMU PIE (top) and FERET (bottom) databases.

Rearrangement Results

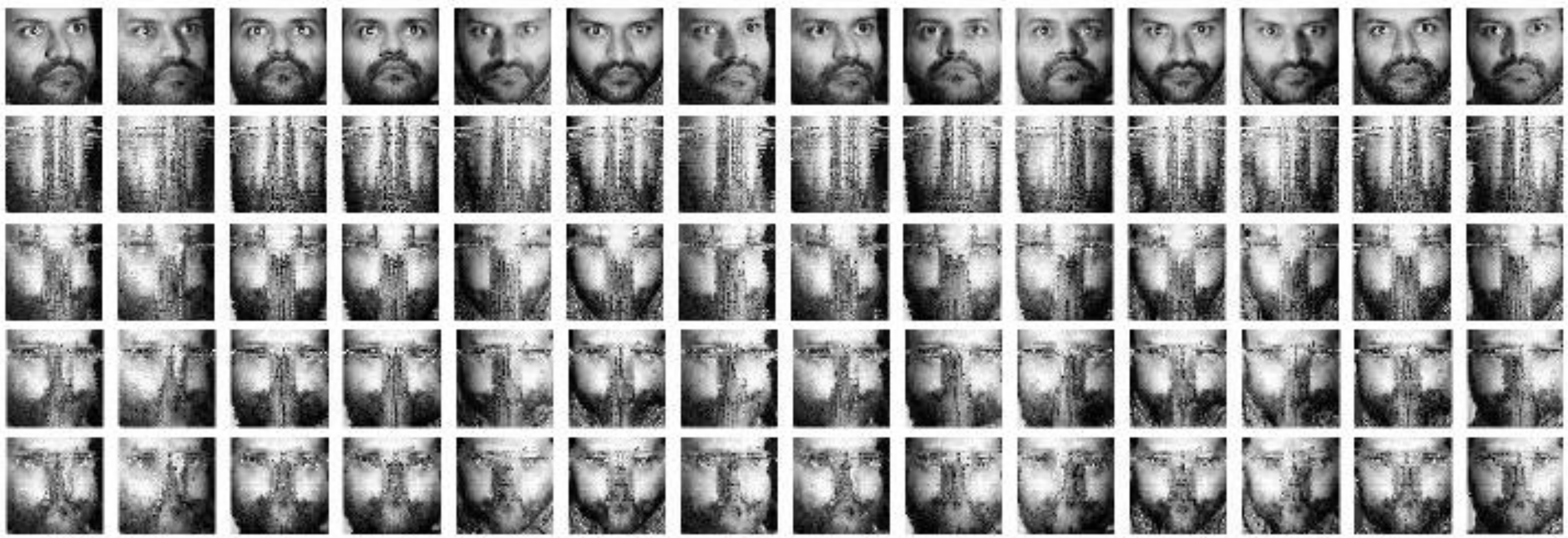


Figure 3. Comparison of fourteen element-rearranged images from the CMU PIE database based on the GLRAM-ER algorithm. The top row displays the original images. The subsequent rows show the rearranged images for $d = 2, 3, 4$, and 5, respectively.

Reconstruction Visualization

Figure 5. Comparison of seven reconstructed images from the FERET database. The top row displays the original images. The subsequent rows show reconstructed results from 2DPCA, 2DPCA-ER, GLRAM, and GLRAM-ER, respectively.

Reconstruction Visualization

Figure 6. Comparison of fourteen reconstructed images from the CMU PIE database. The top row displays the original images. The subsequent rows show reconstructed results from 2DPCA, 2DPCA-ER, GLRAM, and GLRAM-ER, respectively.

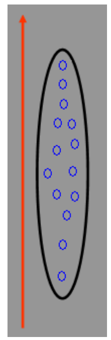
Classification Accuracy

Algorithm	FERET	CMU PIE
PCA	45.1	43.3
LDA	91.8	76.2
MFA	92.1	77.3
2DLDA	94.8	81.9
2DMFA	94.0	82.8
2DLDA-PA	95.6	83.7
2DMFA-PA	96.5	86.4

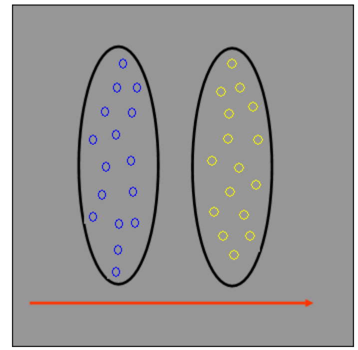
Outline

- ✓Dimensionality Reduction for Tensor-based Objects
- ✓Graph Embedding: A General Framework for Dimensionality Reduction
- ✓Learning using Privileged Information for Face Verification and Person Re-identification

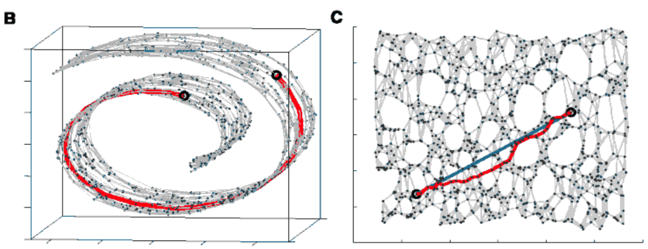
Representative Previous Work



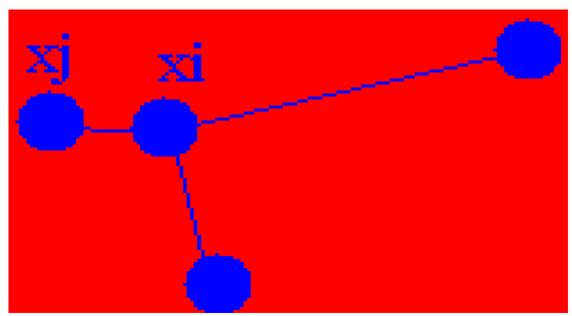
PCA



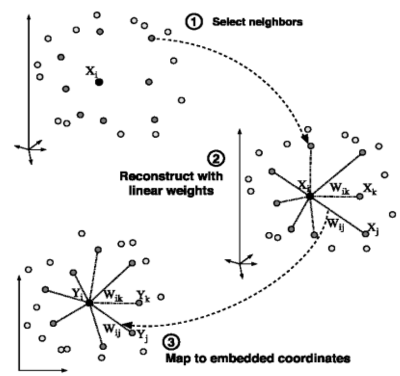
LDA



ISOMAP: Geodesic Distance Preserving
J. Tenenbaum et al., 2000

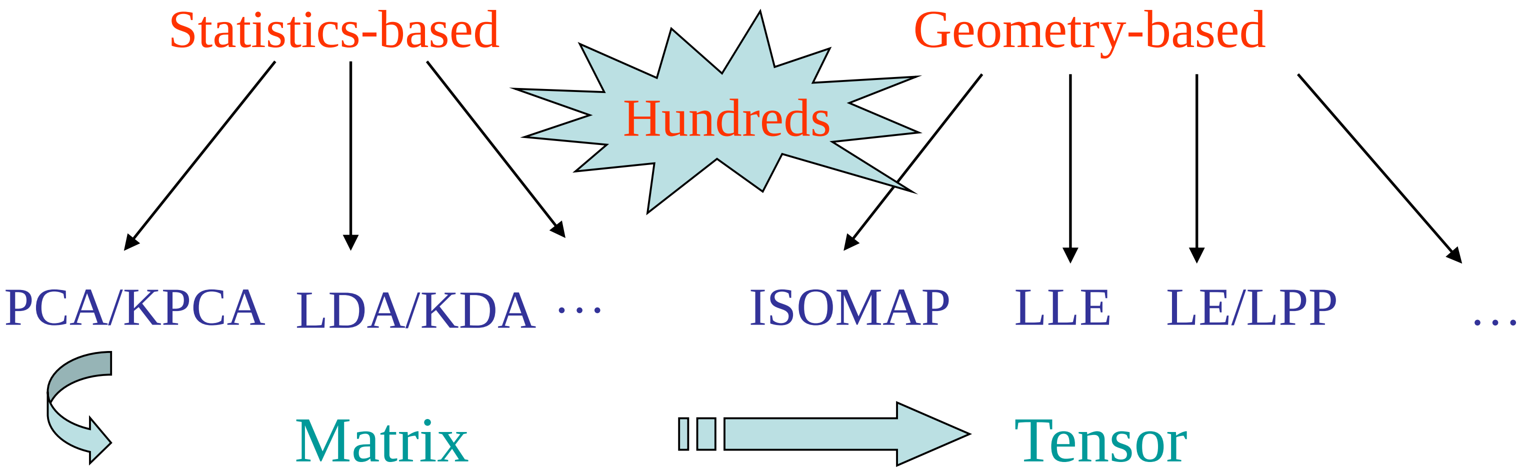


LE/LPP: Local Similarity Preserving, M. Belkin, P. Niyogi et al., 2001, 2003



LLE: Local Neighborhood Relationship Preserving
S. Roweis & L. Saul, 2000

Dimensionality Reduction Algorithms



- Any *common perspective* to understand and explain these dimensionality reduction algorithms? Or any *unified formulation* that is shared by them?

- Any *general tool* to guide developing new algorithms for dimensionality reduction?

Our Answers

Type	Direct Graph Embedding	Linearization	Kernelization
Formulation	$\min_{y^T B y = 1} y^T L y$	$y = X^T w$	$w = \sum_i \alpha_i \phi(x_i)$
Example	Original PCA & LDA, ISOMAP, LLE, Laplacian Eigenmap	PCA, LDA, LPP	KPCA, KDA

Type → **Tensorization**

Formulation → $y_i = X_i \times_1 w^1 \times_2 w^2 \cdots \times_n w^n$

Example → CSA, DATER

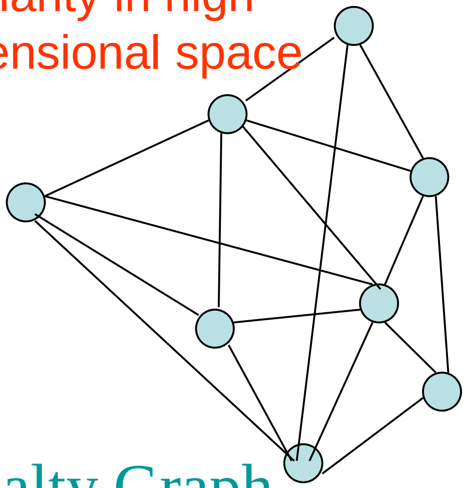
S. Yan, D. Xu et al., *CVPR 2005, T-PAMI 2007*

Direct Graph Embedding

Intrinsic Graph:

$G = [x_i, S_{ij}]$

Similarity in high dimensional space



Penalty Graph

$G^P = [x_i, S^P_{ij}]$

S, S^P: *Similarity matrix (graph edge)*

L, B: *Laplacian matrix from* S, S^P;

$$L = D - S, \quad D_{ii} = \sum_{j \neq i} S_{ij} \quad \forall i$$

Data in high-dimensional space and low-dimensional space (assumed as 1D space here):

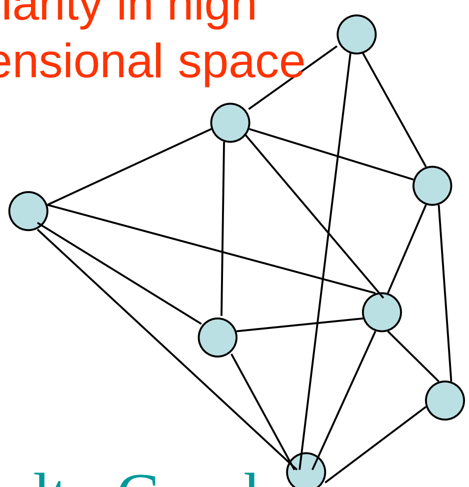
$$X = [x_1, x_2, \ldots, x_N] \quad y = [y_1, y_2, \ldots, y_N]^T$$

Direct Graph Embedding -- Continued

Intrinsic Graph:

$G = [x_i, S_{ij}]$

Similarity in high dimensional space



Penalty Graph

$G^P = [x_i, S_{ij}^P]$

S, S^P: *Similarity matrix (graph edge)*

L, B: *Laplacian matrix from S, S^P;*

$L = D - S, \quad D_{ii} = \sum_{j \neq i} S_{ij} \quad \forall i$

Data in high-dimensional space and low-dimensional space (assumed as 1D space here):

$X = [x_1, x_2, \ldots, x_N] \quad y = [y_1, y_2, \ldots, y_N]^T$

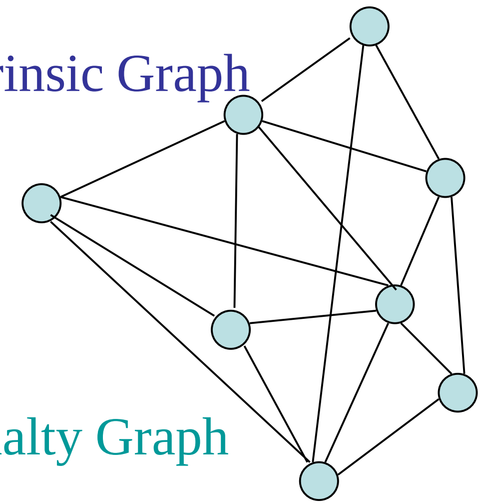
Criterion to Preserve Graph Similarity:

$$y^* = \underset{\substack{y^T y = 1 \text{ or} \\ y^T B y = 1}}{\arg\min} \sum_{i \neq j} \| y_i - y_j \|^2 S_{ij} = \underset{\substack{y^T y = 1 \text{ or} \\ y^T B y = 1}}{\arg\min} \, y^T L y$$

Problem: It cannot handle new test data.

Linearization

Intrinsic Graph



Penalty Graph

Linear mapping function

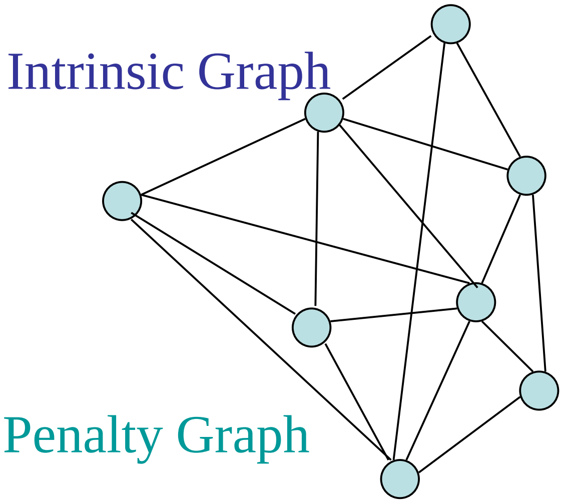
$$y = X^{\mathrm{T}} w$$

Objective function in Linearization

$$w^* = \underset{\substack{w^{\mathrm{T}} w = 1 \ or \\ w^{\mathrm{T}} X B X^{\mathrm{T}} w = 1}}{\arg\min} \; w^{\mathrm{T}} X L X^{\mathrm{T}} w$$

Problem: linear mapping function is not enough to preserve the real nonlinear structure?

Kernelization



Nonlinear mapping: $\phi : x_i \rightarrow \phi(x_i)$

the original input space to another higher dimensional Hilbert space.

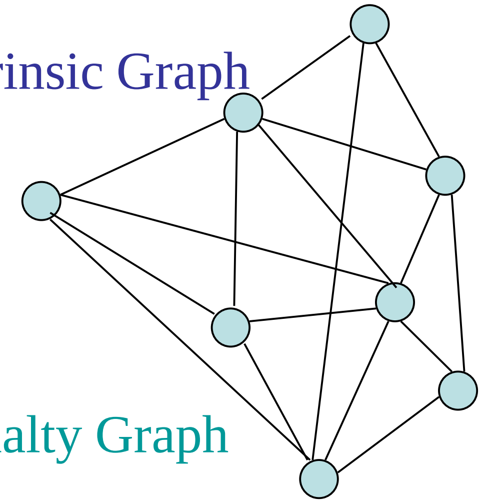
Constraint: $w = \sum_i \alpha_i \phi(x_i)$

Kernel matrix: $K_{ij} = k(x_i, x_j)$ $k(x, y) = \phi(x) \cdot \phi(y)$

Objective function in Kernelization

$$a^* = \arg\min_{\substack{\alpha^T K \alpha = 1 \ or \\ \alpha^T KBK \alpha = 1}} \alpha^T KLK \alpha$$

Tensorization



Low dimensional representation is obtained as:

$$y_i = \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n$$

Objective function in Tensorization

$$(w^1,\ldots,w^n)^* = \arg\min_{f(w^1,\ldots,w^n)=1} \sum_{i\neq j} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n - \mathbf{X}_j \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 S_{ij}$$

where

$$f(w^1,\ldots,w^n) = \sum_{i=1}^{N} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 B_{ii} \qquad or$$

$$f(w^1,\ldots,w^n) = \sum_{i\neq j} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n - \mathbf{X}_j \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 S_{ij}^P$$

Common Formulation

Intrinsic graph

Penalty graph

S, S^P: *Similarity matrix*

L, B: *Laplacian matrix from* S, S^P;

Direct Graph Embedding

$$y^* = \arg\min_{\substack{y^T y = 1 \text{ or} \\ y^T \boxed{B} y = 1}} y^T \boxed{L} y$$

Linearization

$$w^* = \arg\min_{\substack{w^T w = 1 \text{ or} \\ w^T X \boxed{B} X^T w = 1}} w^T X \boxed{L} X^T w$$

Kernelization

$$a^* = \arg\min_{\substack{\alpha^T K \alpha = 1 \text{ or} \\ \alpha^T K \boxed{B} K \alpha = 1}} \alpha^T K \boxed{L} K \alpha$$

Tensorization

$$(w^1, \ldots, w^n)^* = \arg\min_{f(w^1, \ldots, w^n) = 1} \sum_{i \neq j} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n - \mathbf{X}_j \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 \boxed{S_{ij}}$$

where

$$f(w^1, \ldots, w^n) = \sum_{i=1}^{N} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 \boxed{B_{ii}} \text{ or}$$

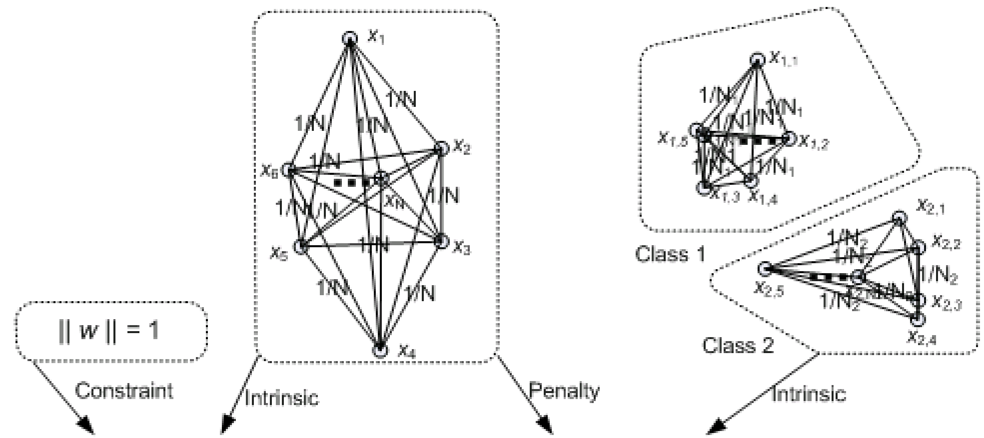
$$f(w^1, \ldots, w^n) = \sum_{i \neq j} \| \mathbf{X}_i \times_1 w^1 \times_2 w^2 \ldots \times_n w^n - \mathbf{X}_j \times_1 w^1 \times_2 w^2 \ldots \times_n w^n \|^2 \boxed{S_{ij}^P}$$

A General Framework for Dimensionality Reduction

Algorithm	S & B Definition	Embedding Type
PCA/KPCA/CSA	$S_{ij} = 1/N,\ i \neq j; B = I$	L/K/T
LDA/KDA/DATER	$S_{ij} = \delta_{l_i, l_j} / n_{l_i},\ B = I - \frac{1}{N} e e^T$	L/K/T
ISOMAP	$S_{ij} = \tau(D_G)_{ij},\ i \neq j;\ \ B = I$	D
LLE	$S = M + M^T - M^T M;\ B = I$	D
LE/LPP	$S_{ij} = \exp\{-\|x_i - x_j\|^2 / t\}$ $if\ \|x_i - x_j\| < \varepsilon$; $B=D$	D/L

***D*: Direct Graph Embedding**
***L*: Linearization**
***K*: Kernelization**
***T*: Tensorization**

General Framework: *PCA and LDA*



(a) Constraint and Intrinsic Graph in PCA (b) Penalty and Intrinsic Graphs in LDA

$$w^* = \arg\min_{w^{\mathrm{T}} w=1} w^{\mathrm{T}} C w$$

$$C = \frac{1}{N}\sum_i (x_i-\bar{x})(x_i-\bar{x})^{\mathrm{T}} = \frac{1}{N}X\left(I-\frac{1}{N}ee^{\mathrm{T}}\right)X^{T}$$

$$e=[1,1,\ldots,1]^{\mathrm{T}}$$

$$W_{ij} = 1/N,\ i\neq j; B = I$$

Principal Component Analysis

$$w^* = \arg\min_w \frac{w^{\mathrm{T}} S_W w}{w^{\mathrm{T}} S_B w} = \arg\min_w \frac{w^{\mathrm{T}} S_W w}{w^{\mathrm{T}} C w}$$

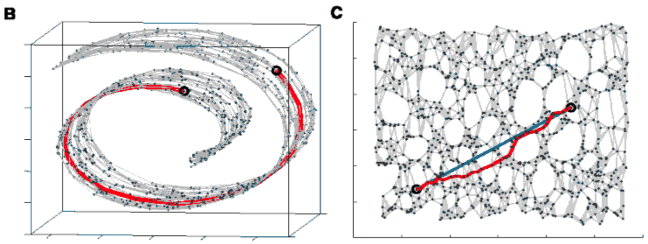
$$S_W = \sum_{i=1}^{N} (x_i-\bar{x}^{l_i})(x_i-\bar{x}^{l_i})^{\mathrm{T}} = X\left(I-\sum_{c=1}^{N_c}\frac{1}{n_c}e^c e^{c\mathrm{T}}\right)X^{T}$$

$$S_B = \sum_{c=1}^{N_c} n_c(\bar{x}^c-\bar{x})(\bar{x}^c-\bar{x})^{\mathrm{T}} = NC - S_W$$

$$W_{ij} = \delta_{l_i,l_j}/n_{l_i},\ B = I-\frac{1}{N}ee^{\mathrm{T}}$$

Linear Discriminant Analysis

General Framework: *ISOMAP*



Geodesic Distance Preserving

Geodesic Distance Matrix

$\boldsymbol{D_G}$

Inner Product Matrix

$$\boldsymbol{\tau}(\boldsymbol{D_G}) = -\boldsymbol{HSH}/2$$

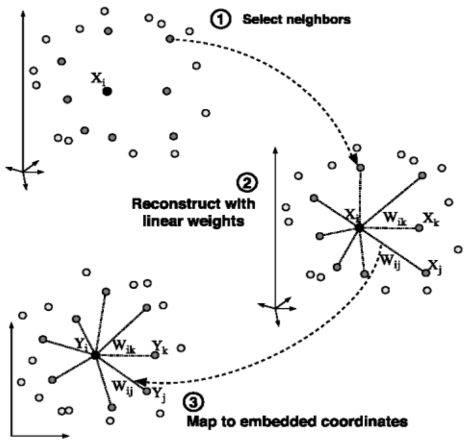
$$\boldsymbol{S_{ij}} = \boldsymbol{D_{ij}^2} \ \ \textbf{and} \ \ \boldsymbol{H_{ij}} = \boldsymbol{\delta_{ij}} - 1/\boldsymbol{N}$$

Multidimensional Scaling $\boldsymbol{y}^* = \boldsymbol{Arg}\max_{\boldsymbol{y}} \boldsymbol{y}^T \boldsymbol{\tau}(\boldsymbol{D_G})\boldsymbol{y}$

$$\begin{aligned}
\sum_j \tau(D_G)_{ij} &= \sum_j (-HSH/2)_{ij} \\
&= \sum_j (-(I - \tfrac{1}{N} ee^{\mathrm{T}}) S (I - \tfrac{1}{N} ee^{\mathrm{T}})/2)_{ij} \\
&= \tfrac{1}{2} \sum_j (-S_{ij} + \tfrac{1}{N} \sum_{i'} S_{i'j} + \tfrac{1}{N} \sum_{j'} (S_{ij'} - \tfrac{1}{N} \sum_{i'} S_{i'j'})) \\
&= (-\tfrac{1}{2} \sum_j S_{ij} + \tfrac{1}{2N} \sum_{jj'} S_{ij'}) + (\tfrac{1}{2N} \sum_{ji'} S_{i'j} - \tfrac{1}{2N^2} \sum_{jj'i'} S_{i'j'}) \\
&= 0
\end{aligned}$$

$$W_{ij} = \tau(D_G)_{ij},\ i \neq j; \quad B = I$$

General Framework: *LLE*



$$\vec{X}_i = \sum_j W_{ij} \vec{X}_j$$

$$\Phi(y) = \sum_i | y_i - \sum_j W_{ij} y_i |^2$$

$$y^* = Arg \min_y y^T M \, y$$

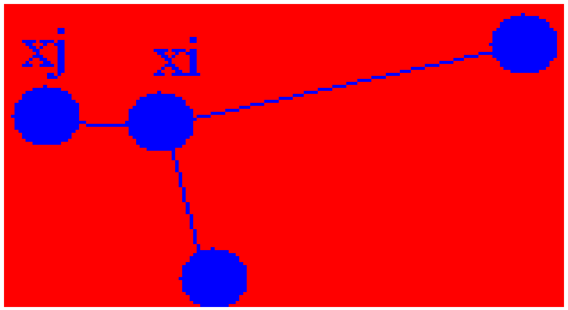
$$where \;\; M_{ij} = \delta_{ij} - w_{ij} - w_{ji} + \sum_k w_{ki} w_{kj}$$

Local neighborhood relationship preserving

$$\begin{aligned}
\sum_j [(I - M^{\mathrm{T}})(I - M)]_{ij} &= \sum_j I_{ij} - M_{ij} - M_{ji} + (M^{\mathrm{T}} M)_{ij} \\
&= 1 - \sum_j (M_{ij} + M_{ji}) + \sum_j \sum_k M_{ki} M_{kj} \\
&= 1 - \sum_j (M_{ij} + M_{ji}) + \sum_k M_{ki} \sum_j M_{kj} \\
&= 1 - \sum_j (M_{ij} + M_{ji}) + \sum_k M_{ki} \\
&= 1 - \sum_j M_{ij} - \sum_j M_{ji} + \sum_k M_{ki} = 0
\end{aligned}$$

$$W = M + M^{\mathrm{T}} - M^{\mathrm{T}} M \,;\; B = I$$

General Framework: *Laplacian Eigenmap/ LPP*



Local similarity preserving

$$W_{ij} = \begin{cases} \exp\{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{t}\}, & \|\mathbf{x}_i - \mathbf{x}_j\|^2 < \varepsilon \\ 0 & \text{otherwise.} \end{cases}$$

$$L = D - W, D_{ii} = \sum_j W_{ij}$$

$$\min_{y^T D y = 1} \sum_{ij} (y_i - y_j)^2 W_{ij}$$

$$\min_{y^T D y = 1} y^T L y$$

Laplacian Eigenmap

$$y = X^T \mathbf{w}$$

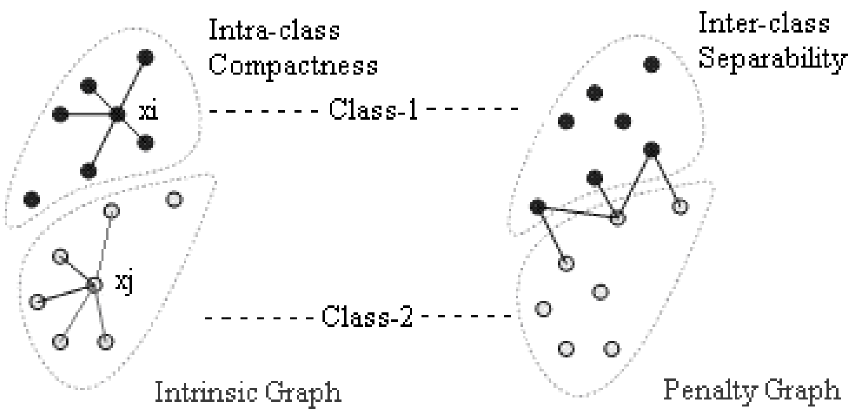
$$XLX^T \mathbf{w} = \lambda XDX^T \mathbf{w}$$

LPP

$$W_{ij} = \begin{cases} \exp\{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{t}\}, & \|\mathbf{x}_i - \mathbf{x}_j\|^2 < \varepsilon \\ 0 & \text{otherwise.} \end{cases}$$

$B = D$

New Dimensionality Reduction Algorithm: *Marginal Fisher Analysis*



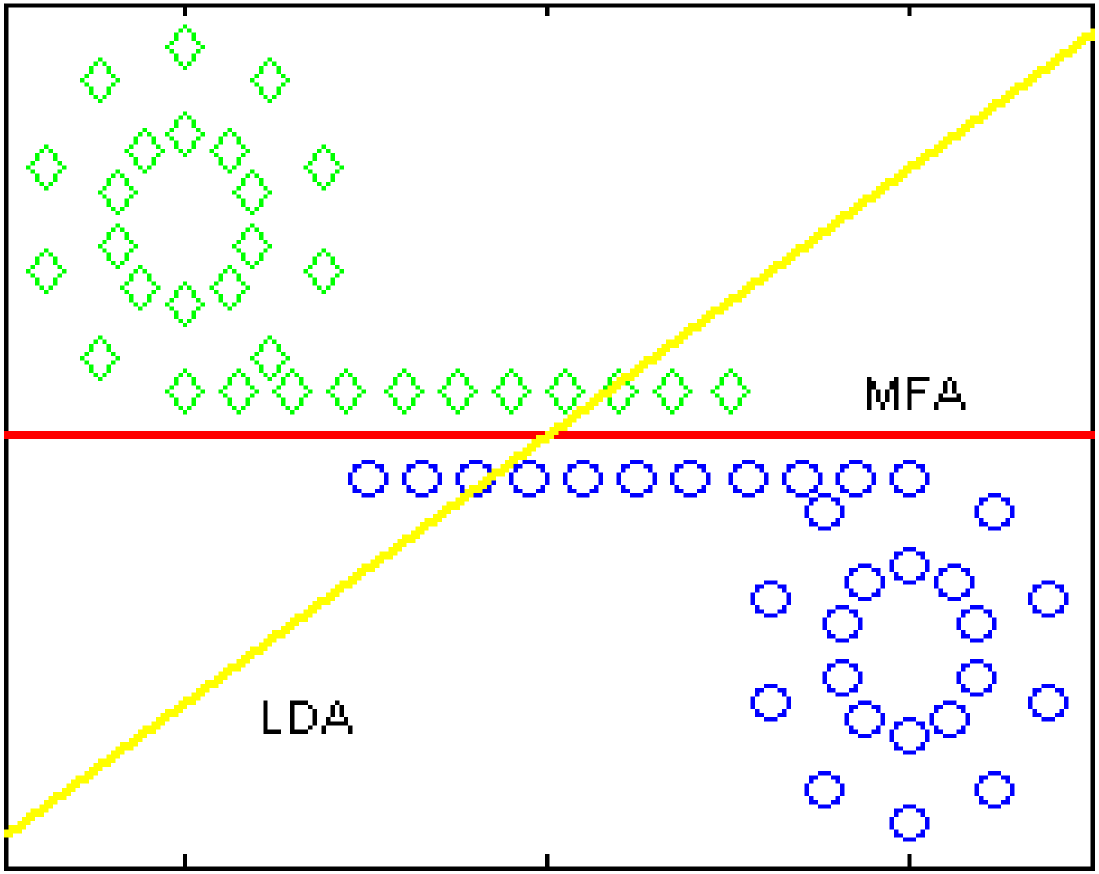
Important Information for face recognition:

1) Label information
2) Local manifold structure (*neighborhood* or *margin*)

$S_{ij} =$ *1*: *if* x_i is among the k_1-nearest neighbors of x_j in the same class;
0 : *otherwise*

$S^P_{ij} =$ *1*: *if* the pair (i,j) is among the k_2 shortest pairs among the data set;
0: *otherwise*

Marginal Fisher Analysis: Advantage



No Gaussian distribution assumption

Experiments: *Face Recognition*

ORL	G3/P7	G4/P6
PCA+LDA (Linearization)	87.9%	88.3%
PCA+MFA (Ours)	89.3%	91.3%
KDA (Kernelization)	87.5%	91.7%
KMFA (Ours)	88.6%	93.8%
DATER-2 (Tensorization)	89.3%	92.0%
TMFA-2 (Ours)	95.0%	96.3%

PIE-1	G3/P7	G4/P6
PCA+LDA (Linearization)	65.8%	80.2%
PCA+MFA (Ours)	71.0%	84.9%
KDA (Kernelization)	70.0%	81.0%
KMFA (Ours)	72.3%	85.2%
DATER-2 (Tensorization)	80.0%	82.3%
TMFA-2 (Ours)	82.1%	85.2%

Summary

- Optimization framework that unifies previous dimensionality reduction algorithms as special cases.

- A new dimensionality reduction algorithm: Marginal Fisher Analysis.

Outline

- ✓Dimensionality Reduction for Tensor-based Objects
- ✓Graph Embedding: A General Framework for Dimensionality Reduction
- ✓Learning using Privileged Information for Face Verification and Person Re-identification

Learning using Privileged Information

Privileged information:
Information available only in the training process (not available in the testing process).

Training: attending classes in the classroom
Testing: taking an exam
Privileged Information: teacher's instruction

SVM+
(Primal Form)

$$\min_{\tilde{\mathbf{w}},\tilde{b},\mathbf{w},b} \quad \frac{1}{2}\left(\|\mathbf{w}\|^2 + \gamma\|\tilde{\mathbf{w}}\|^2\right) + C\sum_{i=1}^{n}\xi(\tilde{\mathbf{x}}_i),$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\phi(\mathbf{x}_i)+b) \geq 1-\xi(\tilde{\mathbf{x}}_i), \quad \xi(\tilde{\mathbf{x}}_i) \geq 0, \quad \forall i,$$

$$\xi(\tilde{\mathbf{x}}_i) = \tilde{\mathbf{w}}'\tilde{\phi}(\tilde{\mathbf{x}}_i) + \tilde{b}$$

oracle function

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i$$

$$\text{s.t.} \quad y_i(\mathbf{w}'\phi(\mathbf{x}_i)+b) \geq 1-\xi_i$$

$$\xi_i \geq 0$$

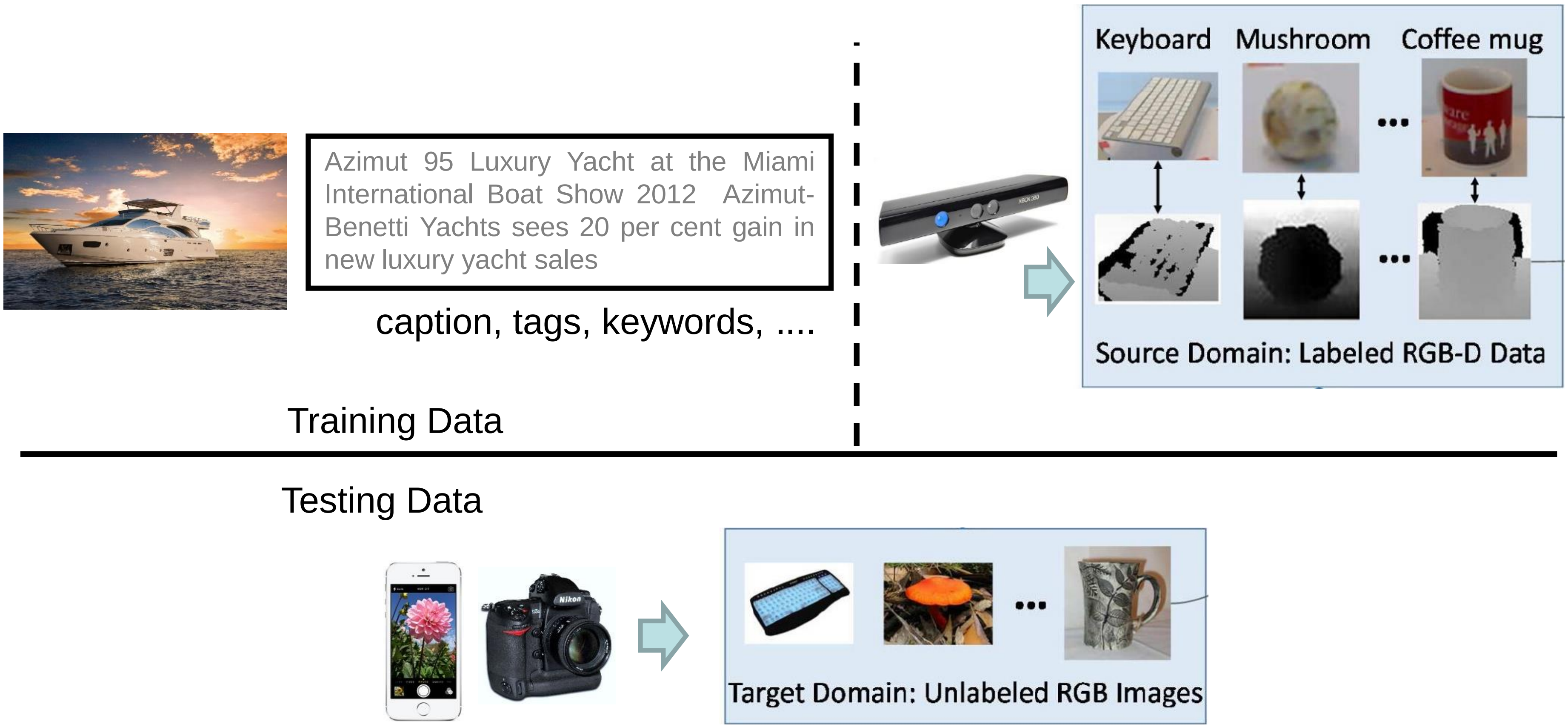
primal form of SVM

x_i : main feature

$\tilde{x}_i$: privileged information

V. Vapnik and A. Vashist, A new learning paradigm: Learning using privileged information. Neural Networks, 544–557, 2009.

Applications in Image and Video Recognition

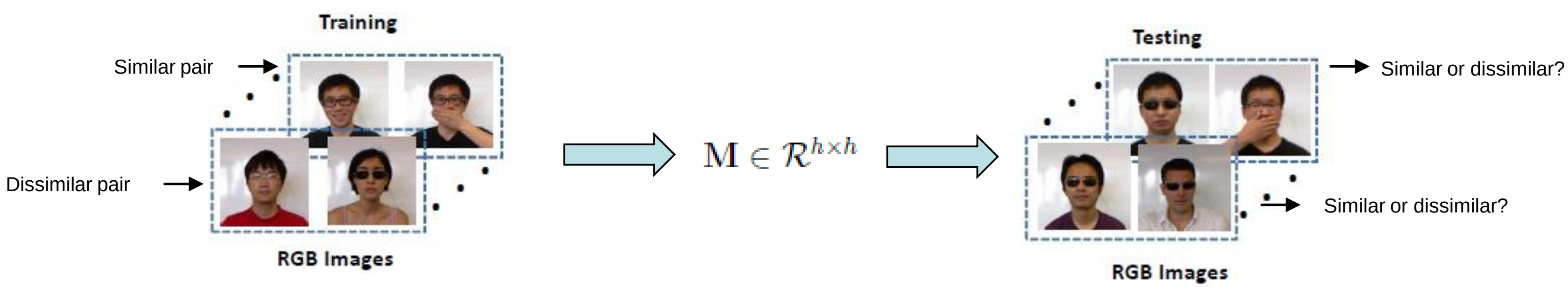


1) Web images/videos are usually associated with additional surrounding textual descriptions (tags, captions, etc.)

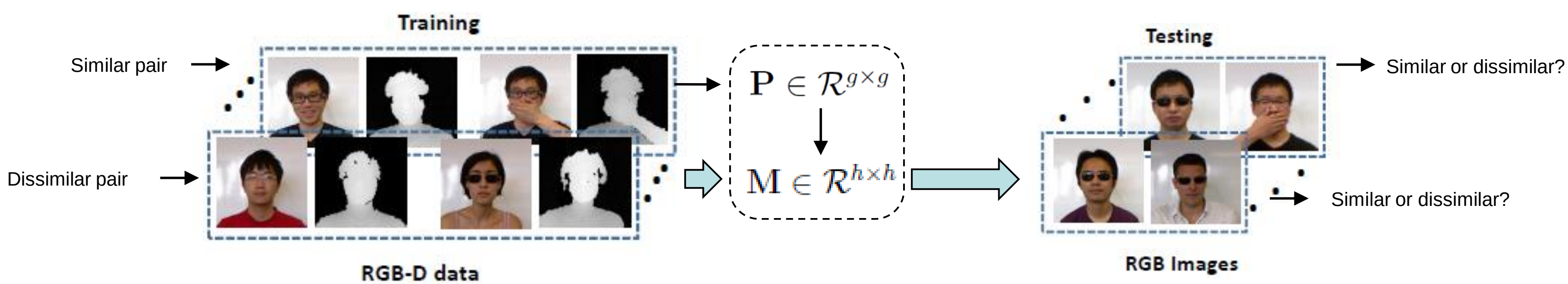
2) RGB-D images from Kinect cameras contain additional depth images

Distance Metric Learning using Privileged Information

- Distance Metric Learning for Face Verification
  - Mahalanobis distance: $d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)'\mathbf{M}(\mathbf{x}_i - \mathbf{x}_j)$



- Distance Metric Learning using Privileged Information
  - Additional depth information is used as privileged information



Information Theoretic Distance Metric Learning (ITML)

- Distance Metric Learning
  - Given training dataset $\{\mathbf{x}_i\}|_{i=1}^{n}$, where $\mathbf{x}_i \in \mathcal{R}^h$ is the feature vector and their labels:

    $(i,j) \in \mathcal{S}$: two samples belong to the same subject

    $(i,j) \in \mathcal{D}$: two samples are from two different subjects

- Information Theoretic Distance Metric Learning

$$
\begin{aligned}
\min_{\mathrm{M} \succeq 0, \xi_{ij}} \quad & D_{ld}(\mathrm{M}, \mathrm{M}^0) + \gamma L(\boldsymbol{\xi}, \boldsymbol{\xi}^0) \\
\text{s.t.,} \quad & d^2_{\mathrm{M}}(\mathbf{x}_i, \mathbf{x}_j) \le \xi_{ij}, \quad (i,j) \in \mathcal{S}, \\
& d^2_{\mathrm{M}}(\mathbf{x}_i, \mathbf{x}_j) \ge \xi_{ij}, \quad (i,j) \in \mathcal{D},
\end{aligned}
$$

where

$$
D_{ld}(\mathrm{M}, \mathrm{M}^0) = \mathrm{tr}(\mathrm{M}(\mathrm{M}^0)^{-1}) - \log\det(\mathrm{M}(\mathrm{M}^0)^{-1}) - h
$$

$$
L(\boldsymbol{\xi}, \boldsymbol{\xi}^0) = D_{ld}(\mathrm{diag}(\boldsymbol{\xi}), \mathrm{diag}(\boldsymbol{\xi}^0))
$$

$$
\xi^0_{ij} = \begin{cases} u & (i,j) \in \mathcal{S}, \\ l & (i,j) \in \mathcal{D} \end{cases}
$$

Information Theoretic Distance Metric Learning using Privileged Information (ITML+)

- Objective Function
  - We learn $\mathbf{M} \in \mathcal{R}^{h\times h}$ and $\mathbf{P} \in \mathcal{R}^{g\times g}$ simultaneously:

$$\min_{\mathbf{M}\succeq 0, \mathbf{P}\succeq 0} \Omega(\mathbf{M},\mathbf{P}) + \gamma \sum_{(i,j)\in\mathcal{S}\cup\mathcal{D}} \ell(d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \xi^0_{ij})$$

$$\text{s.t.,}\quad d^2_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \le d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \quad (i,j)\in S,$$

$$d^2_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \ge d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \quad (i,j)\in \mathcal{D},$$

where

$$\Omega(\mathbf{M},\mathbf{P}) = D_{ld}(\mathbf{M},\mathbf{M}^0) + \lambda D_{ld}(\mathbf{P},\mathbf{P}^0) \quad \ell(d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \xi^0_{ij}) = D_{ld}(d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \xi^0_{ij})$$

  - Discussions:
    - For similar pairs, the learned distance should be more similar

$$d^2_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \le d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \quad \forall (i,j) \in S$$

    - For dissimilar pairs, the learned distance should be more dissimilar $d^2_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \ge d^2_{\mathbf{P}}(\mathbf{z}_i,\mathbf{z}_j), \quad (i,j)\in \mathcal{D}$

X. Xu, W. Li and D. Xu, *T-NNLS 2015*

Experiments: Face Verification

- Setting:
  - Main features (i.e. gradient-LBP features) from RGB images
  - Privileged information (i.e. gradient-LBP features) from depth images

	L2 distance	SVM	ITML	LMNN	ITML-S	SVM+	ITML+
AP	58.82±0.64	66.02±1.92	84.16±0.80	84.28±0.60	83.94±0.99	66.01±0.99	**86.82±0.79**
AUC	70.37±0.19	82.02±0.86	92.76±0.21	92.96±0.25	92.57±0.48	82.98±0.57	**93.80±0.41**

Results (AP% and AUC%) on the EUROCOM face dataset

Experiments: Person Re-identification

- Setting:
  - Main features (i.e. kernel descriptor features) from RGB images
  - Privileged information (i.e. kernel descriptor features) from depth images

	L2 distance	SVM	ITML	LMNN	ITML-S	SVM+	ITML+
Walking	34.59±0.00	31.77±0.19	46.62±0.35	33.05±0.16	46.69±0.82	26.84±1.56	**48.23±0.69**
Still	85.83 ± 0.00	81.17±0.50	92.89±0.16	86.13±0.08	93.01±0.39	79.21±0.76	**95.23 ±0.31**

Results (mean Rank-1 recognition rates%) on the BIWI RGBD-ID dataset

Summary

- A new distance metric learning method to take advantage of additional depth images as privileged information

Thank you