The 4th Chinese Workshop on Machine Learning and Applications
第四届机器学习及其应用研讨会

ParNeC

CCA with Its Applications (Part II)

Chen Songcan (陈松灿)

s.chen@nuaa.edu.cn

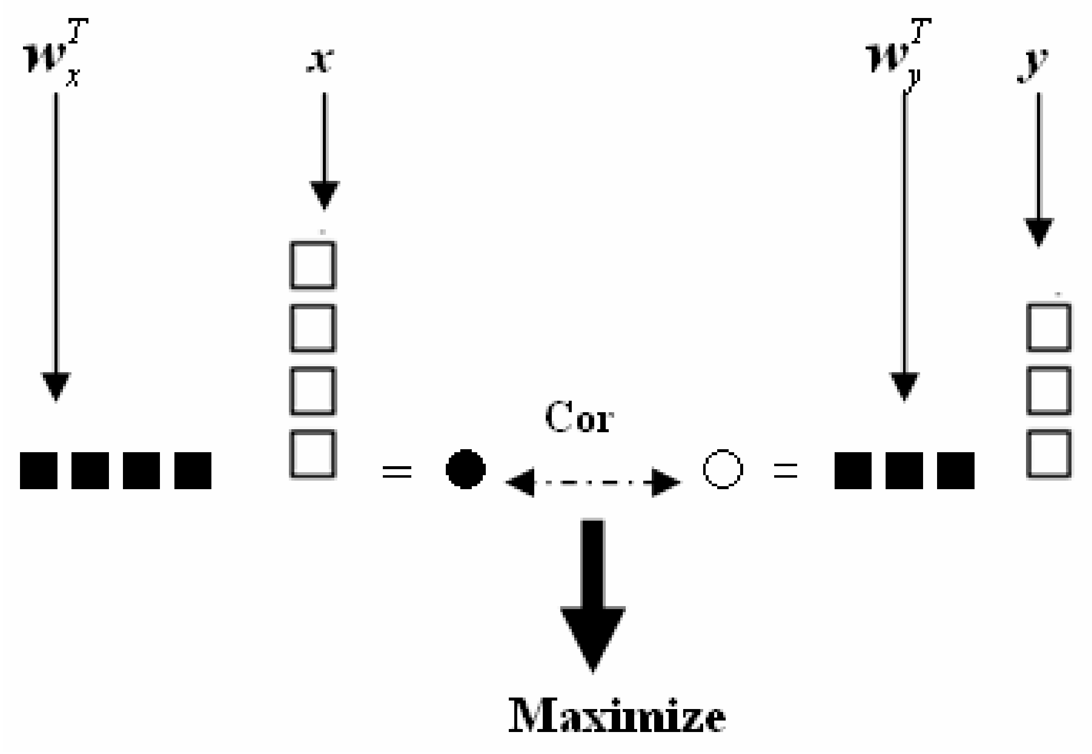
http://parnec.nuaa.edu.cn

Nanjing University of Aeronautics & Astronautics

MLA'06, Nanjing

CCA Illustration

Given X set $\{x_1,\ldots,x_n\} \in R^p$, and Y set $\{y_1,\ldots,y_n\} \in R^q$, CCA aims to simultaneously seek $w_x \in R^p$, $w_y \in R^q$, to ensure [Bor99] (specially use for feature extraction)



CCA formulation

$$\max_{\boldsymbol{w}_x,\boldsymbol{w}_y} \frac{\operatorname{cov}\left(\boldsymbol{w}_x^T\boldsymbol{x},\boldsymbol{w}_y^T\boldsymbol{y}\right)}{\sqrt{\operatorname{var}\left(\boldsymbol{w}_x^T\boldsymbol{x}\right)}\sqrt{\operatorname{var}\left(\boldsymbol{w}_y^T\boldsymbol{y}\right)}}$$

$$=\max_{\boldsymbol{w}_x,\boldsymbol{w}_y} \frac{\sum_{i=1}^{n}\boldsymbol{w}_x^T\boldsymbol{x}_i\boldsymbol{y}_i^T\boldsymbol{w}_y}{\sqrt{\sum_{i=1}^{n}\boldsymbol{w}_x^T\boldsymbol{x}_i\boldsymbol{x}_i^T\boldsymbol{w}_x}\sqrt{\sum_{i=1}^{n}\boldsymbol{w}_y^T\boldsymbol{y}_i\boldsymbol{y}_i^T\boldsymbol{w}_y}}$$

$$\Rightarrow\begin{pmatrix} & \boldsymbol{X}\boldsymbol{Y}^T \\ \boldsymbol{Y}\boldsymbol{X}^T & \end{pmatrix}\begin{pmatrix}\boldsymbol{w}_x\\ \boldsymbol{w}_y\end{pmatrix}=\lambda\begin{pmatrix}\boldsymbol{X}\boldsymbol{X}^T & \\ & \boldsymbol{Y}\boldsymbol{Y}^T\end{pmatrix}\begin{pmatrix}\boldsymbol{w}_x\\ \boldsymbol{w}_y\end{pmatrix}$$

where $\boldsymbol{X}=[\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n],\boldsymbol{Y}=[\boldsymbol{y}_1,\ldots,\boldsymbol{y}_n]$

CCA Alternative formulation

$$\min_{\boldsymbol{w}_x, \boldsymbol{w}_y} \sum_{i=1}^{n} \left\| \boldsymbol{w}_x^T (\boldsymbol{x}_i - \overline{\boldsymbol{x}}) - \boldsymbol{w}_y^T (\boldsymbol{y}_i - \overline{\boldsymbol{y}}) \right\|^2$$

$$\text{s.t.} \quad \sum_{i=1}^{n} \left\| \boldsymbol{w}_x^T (\boldsymbol{x}_i - \overline{\boldsymbol{x}}) \right\|^2 = 1, \ \sum_{i=1}^{n} \left\| \boldsymbol{w}_y^T (\boldsymbol{y}_i - \overline{\boldsymbol{y}}) \right\|^2 = 1$$

$$\Updownarrow$$

$$\max \ \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\boldsymbol{x}_i - \boldsymbol{x}_j \right) \left(\boldsymbol{y}_i - \boldsymbol{y}_j \right)^T \cdot \boldsymbol{w}_y$$

$$\text{s.t.} \quad \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\boldsymbol{x}_i - \boldsymbol{x}_j \right) \left(\boldsymbol{x}_i - \boldsymbol{x}_j \right)^T \cdot \boldsymbol{w}_x = 1$$

$$\boldsymbol{w}_y^T \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\boldsymbol{y}_i - \boldsymbol{y}_j \right) \left(\boldsymbol{y}_i - \boldsymbol{y}_j \right)^T \cdot \boldsymbol{w}_y = 1$$

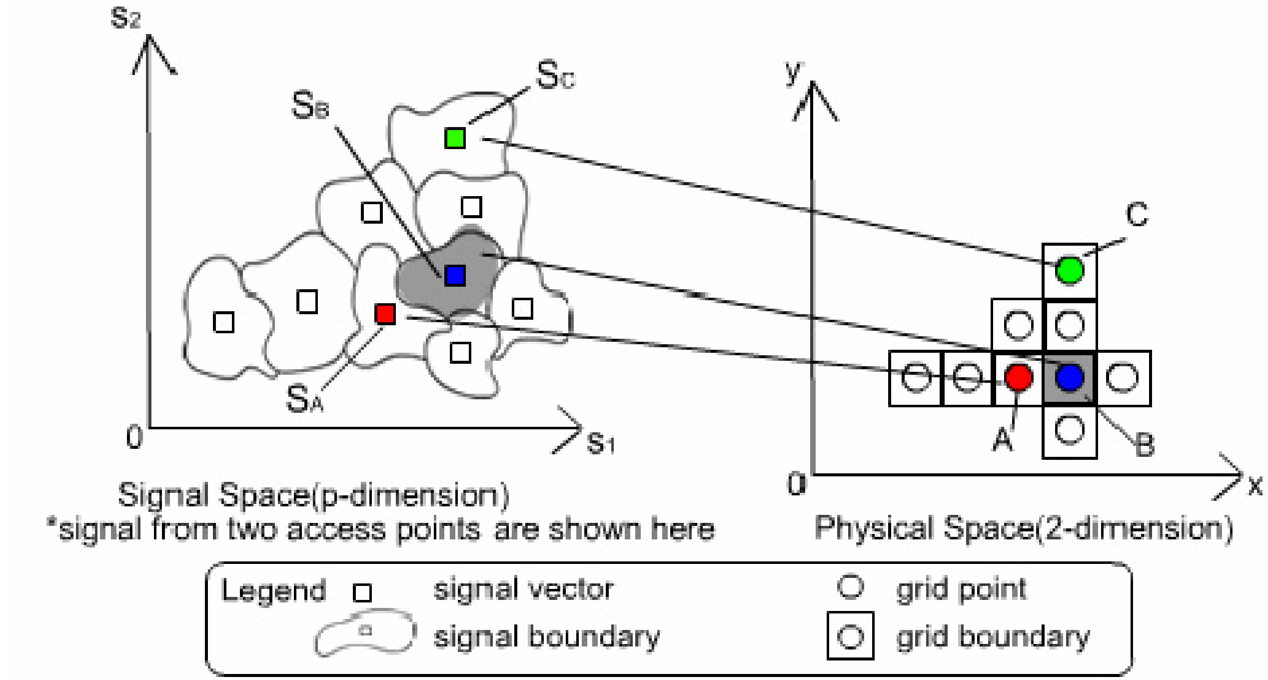
Basis of our LPCCA

Applications

- image retrieval [HSS04], image segmentation [LGD05], image de-noising [He04] , image analysis [Nie02], feature detection of image [And00]
- text categorization, text mining, text retrieval, image-text retrieval , machine translation [For04]
- computer vision [KSE05]
- pattern recognition [SZL05]
- biometrics [YVN03]
- signal processing [SS06]
- location estimation in wireless sensor networks [PKY06]
- facial expression recognition [ZZZ06]
- Multi-view SVM design [FHM05]

New Applications

- **location estimation in wireless sensor networks**

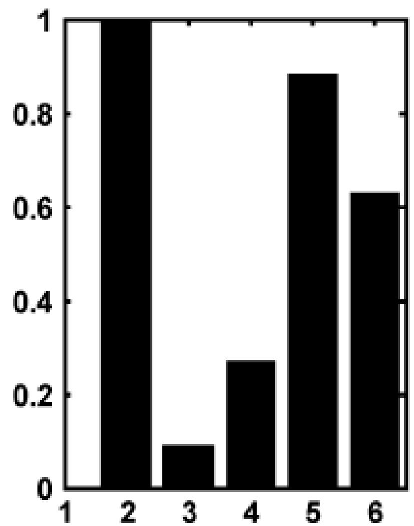


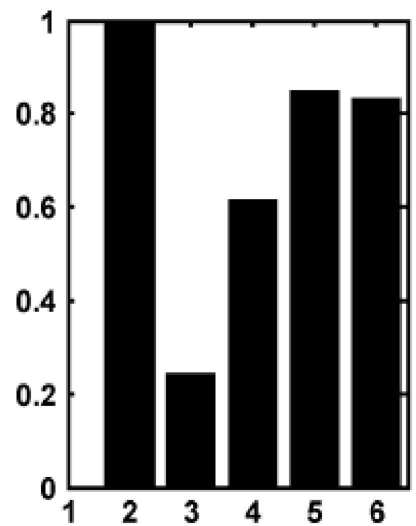
Sensor signal vs. location

[PKY06] J.J. Pan, J.T. Kwok, Q. Yang, Y. Chen, Multidimensional Vector Regression for Accurate and Low-Cost Location Estimation in Pervasive Computing, *IEEE Transaction on Knowledge and Data Engineering*, 18(2006):1181-1193.

New Applications

- **facial expression recognition** **(1:happiness, sadness, surprise, angry, disgust, 6:fear)**





Landmark points vs. Semantic ratings (estimation)

(JAFFE database)

[ZZZ06] W. Zheng, X. Zhou, C. Zou, L. Zhao, Facial Expression Recognition Using Kernel Canonical Correlation Analysis (KCCA), *IEEE Transaction on Neural Networks*, 17(2006), 233-238.

New Applications

❖ **Multi-view SVM design**

$$
\begin{aligned}
\min\ L \quad &= \quad \frac{1}{2}\|\mathbf{w}_A\|^2 + \frac{1}{2}\|\mathbf{w}_B\|^2 + C^A \sum_{i=1}^{\ell} \xi_i^A + C^B \sum_{i=1}^{\ell} \xi_i^B + D \sum_{i=1}^{\ell} \eta_i \\
\text{such that} \quad & \quad |\langle \mathbf{w}_A, \phi_A(\mathbf{x}_i)\rangle + b_A - \langle \mathbf{w}_B, \phi_B(\mathbf{x}_i)\rangle - b_B| \le \eta_i + \epsilon \\
& \quad y_i \left(\langle \mathbf{w}_A, \phi_A(\mathbf{x}_i)\rangle + b_A\right) \ge 1 - \xi_i^A \\
& \quad y_i \left(\langle \mathbf{w}_B, \phi_B(\mathbf{x}_i)\rangle + b_B\right) \ge 1 - \xi_i^B \\
& \quad \xi_i^A \ge 0, \quad \xi_i^B \ge 0, \quad \eta_i \ge 0 \quad \text{all for} \quad 1 \le i \le \ell.
\end{aligned}
$$

CCA alternative formulation

	Motorbike	Bicycle	People	Car
SVM 1	94.05	91.58	91.58	87.95
SVM 2	91.15	91.15	90.57	86.21
KCCA + SVM	94.19	90.28	90.57	88.68
SVM 2K	**94.34**	**93.47**	**92.74**	**90.13**

[FHM05] J. D.R. Farquhar, D. R. Hardoon, H. Meng, J. Shawe-Taylor, S. Szedmak, Two view learning: SVM-2K, Theory and Practice, *NIPS* 2005.

Our previous work

- Locality preserving CCA (LPCCA)
- Kernelized LPCCA
- Their applications in pose estimation and data visualization

Our previous work--LPCCA

- **Motive: CCA is linear, and is insufficient to reveal the nonlinear correlation between two sets X and Y of real-world variables**
- **Objective:**

$$\max_{\boldsymbol{w}_x,\boldsymbol{w}_y} \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^{x}\left(\boldsymbol{x}_i-\boldsymbol{x}_j\right) S_{ij}^{y}\left(\boldsymbol{y}_i-\boldsymbol{y}_j\right)^T \cdot \boldsymbol{w}_y$$

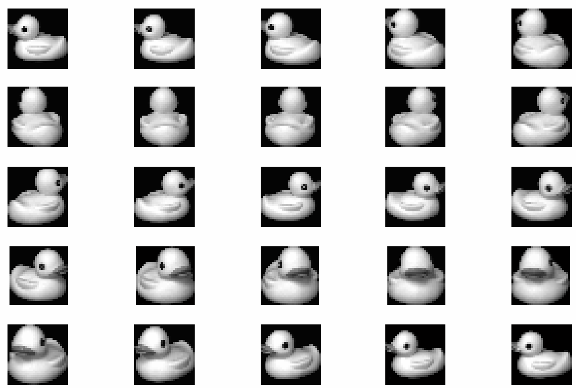
$$\text{s.t.}\quad \boldsymbol{w}_x^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^{x2}\left(\boldsymbol{x}_i-\boldsymbol{x}_j\right)\left(\boldsymbol{x}_i-\boldsymbol{x}_j\right)^T \cdot \boldsymbol{w}_x = 1$$

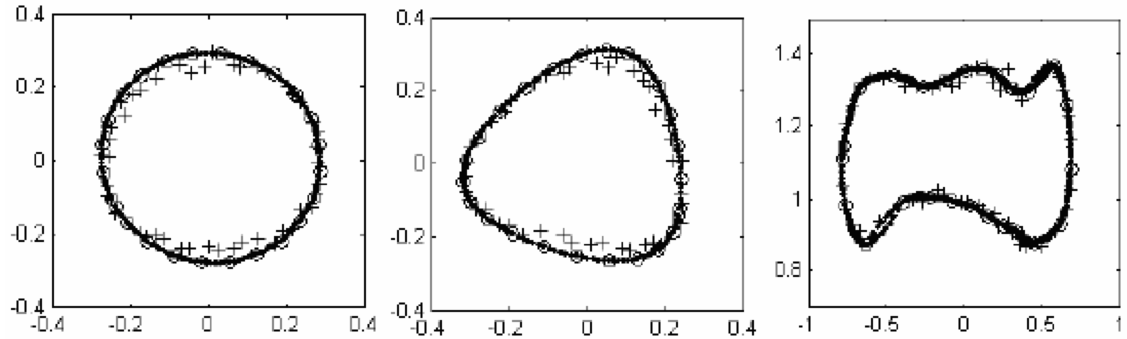
$$\boldsymbol{w}_y^T \cdot \sum_{i=1}^{n}\sum_{j=1}^{n} S_{ij}^{y2}\left(\boldsymbol{y}_i-\boldsymbol{y}_j\right)\left(\boldsymbol{y}_i-\boldsymbol{y}_j\right)^T \cdot \boldsymbol{w}_y = 1$$

$$\Rightarrow \begin{pmatrix} & \boldsymbol{X}\boldsymbol{S}_{xy}\boldsymbol{Y}^T \\ \boldsymbol{Y}\boldsymbol{S}_{yx}\boldsymbol{X}^T & \end{pmatrix}\begin{pmatrix}\boldsymbol{w}_x \\ \boldsymbol{w}_y\end{pmatrix} = \lambda_{LP}\begin{pmatrix} \boldsymbol{X}\boldsymbol{S}_{xx}\boldsymbol{X}^T & \\ & \boldsymbol{Y}\boldsymbol{S}_{yy}\boldsymbol{Y}^T \end{pmatrix}\begin{pmatrix}\boldsymbol{w}_x \\ \boldsymbol{w}_y\end{pmatrix}$$

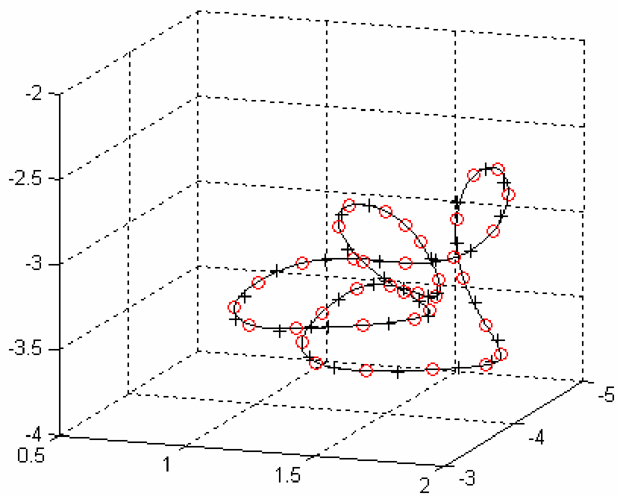
[SC06] Tingkai Sun, Songcan Chen, Locality preserving CCA with applications to data visualization and pose estimation, Image and Vision Computing, in press, 2006

Our previous work

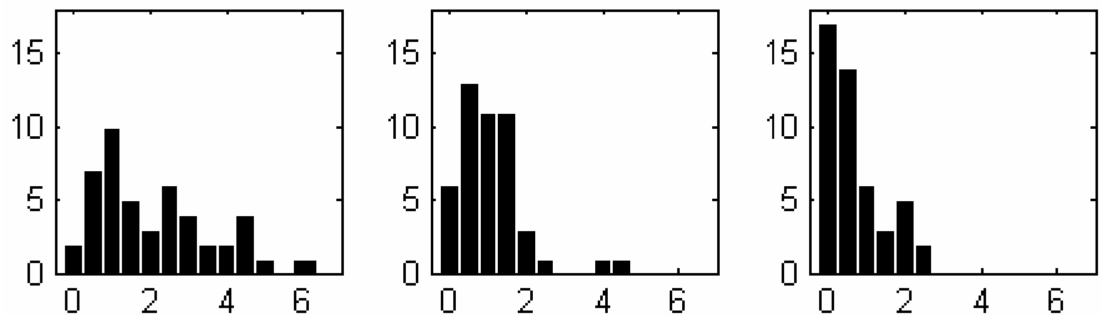




CCA KCCA LPCCA



KLPCCA, TNN in revision



CCA KCCA LPCCA

Error distribution histograms

Our new work

- CIPCA--Class-information-incorporated PCA
- DCCA-- Discriminant CCA
- DCCAM– DCCA with missing samples
- Multi-Kernel learning
- Beyond KCCA

CIPCA

- **Motive**

  1) utilize the class information for feature extraction

  2) CIPCA and CCA can be unified into a framework

$$\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^T \boldsymbol{C}_{zz} \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{D} \boldsymbol{w}}, \quad \text{where} \quad \boldsymbol{D} = \begin{bmatrix} \boldsymbol{X}\boldsymbol{X}^T & \\ & \boldsymbol{Y}\boldsymbol{Y}^T \end{bmatrix} \text{ for CCA}, \quad \boldsymbol{D} = \boldsymbol{I} \text{ for CIPCA}$$

- **Objective:**

$$\max_{\boldsymbol{W}} tr\left(\boldsymbol{W}^T \boldsymbol{C}_{zz} \boldsymbol{W}\right) \Rightarrow \hat{\boldsymbol{y}} = \boldsymbol{W}_Y \boldsymbol{W}_X^+ \boldsymbol{x}$$

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \in \Re^{D+c}$$

[CS05] Songcan Chen, Tingkai Sun, Class-information-incorporated principal component analysis, Neurocomputing, 69 (2005) 216–223

CIPCA

Results

Dataset	PCA	S1	S2	S3	S4	LDA
Image	95.89	*96.02*	77.34	92.51	89.62	77.91
Splice	71.38	70.38	*83.37*	70.94	73.64	79.08
Banana	*86.36*	*86.36*	54.64	71.83	71.83	61.66
B.-Cancer	67.04	*67.13*	54.84	60.71	60.43	64.95
Diabetis	69.45	69.87	*75.57*	72.32	73.93	68.70
F.-Solar	60.91	60.94	62.75	*62.81*	*62.81*	61.10
German	70.07	70.03	*75.64*	70.75	73.60	67.77
Heart	77.02	76.76	*82.59*	77.20	79.83	76.88
Ringnorm	65.63	65.02	*74.78*	65.74	68.69	68.22
Thyroid	*95.68*	95.67	85.76	94.13	92.44	82.55
Titanic	67.00	67.00	67.69	*73.66*	*73.66*	67.04
Twonorm	93.45	93.11	*97.32*	93.30	94.64	96.47
Waveform	84.43	84.25	80.34	84.48	*85.24*	81.43

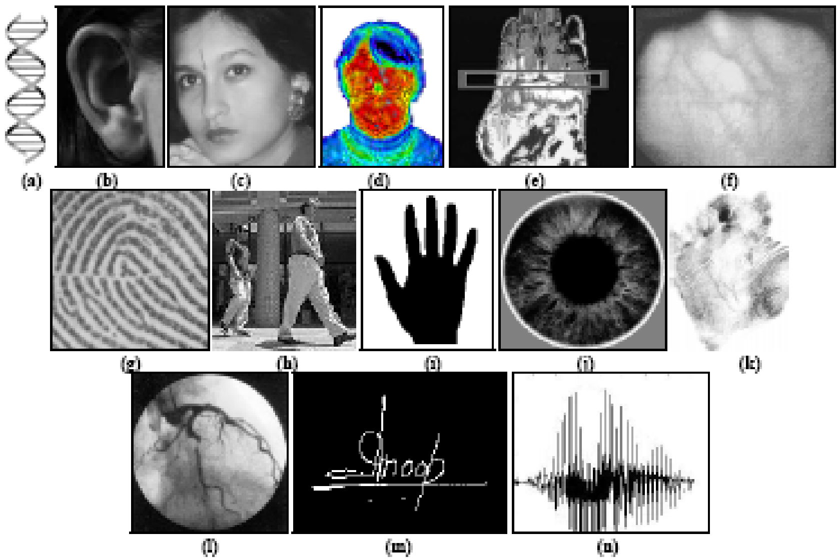
Our new work

- CIPCA
- DCCA
- DCCAM
- Multi-Kernel learning
- Beyond KCCA

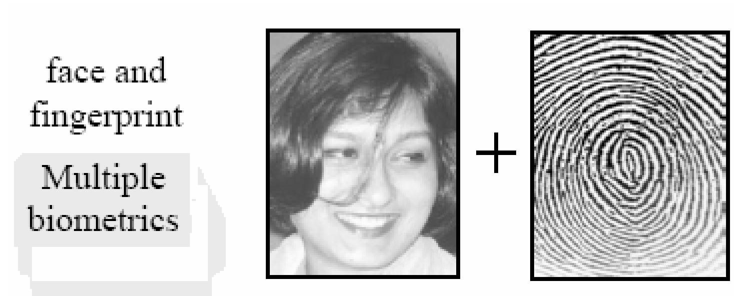
DCCA- modified feature extractor

- **Motive**

  1) for feature fusion and multimodal recognition



Multiple Biometrics [JRS04]



Applications

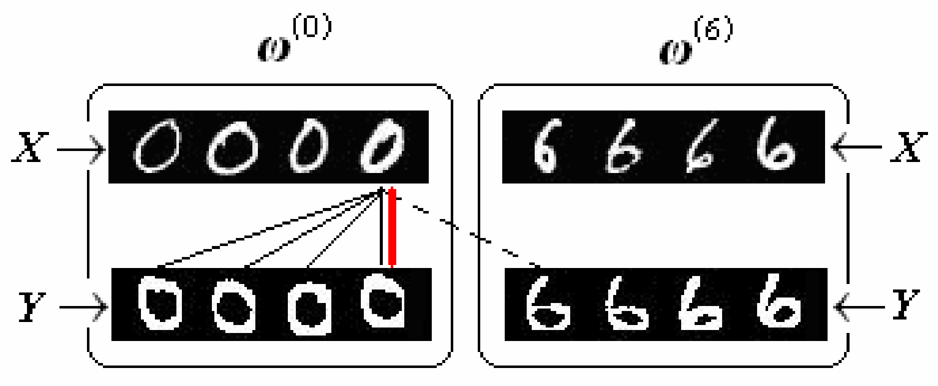
DCCA

- **Motive**

  2) in CCA, the correlation between (x_i, y_i) is insufficient to discriminate between classes

  3) in DCCA, we take correlation as similarity metric, aims to maximize the within-class correlation, minimize the between-class correlation

- **Consider the correlation between classes**



DCCA

- maximize the within-class correlation
  minimize the between-class correlation
- Objective function

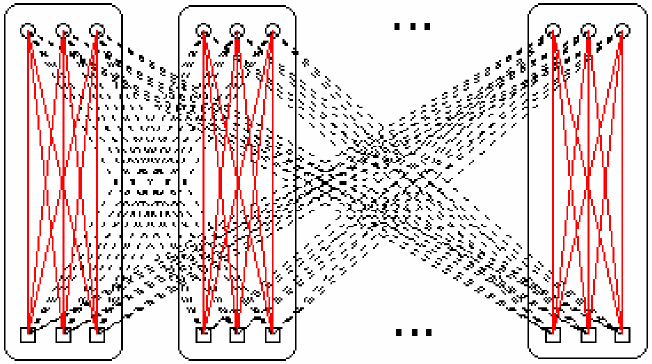
$$\max_{w_x, w_y} \frac{w_x^T (C_w - \eta C_b) w_y}{(w_x^T C_{xx} w_x \cdot w_y^T C_{yy} w_y)^{1/2}}$$

where η is a tunable parameter.

DCCA

- **within-class correlation**

$$C_w = \sum_{i=1}^{c}\sum_{k=1}^{n_i}\sum_{l=1}^{n_i} \boldsymbol{x}_k^{(i)} \boldsymbol{y}_l^{(i)T} = \boldsymbol{XAY}^T$$



- **between-class correlation**

$$\begin{aligned} C_b &= \sum_{i=1}^{c}\sum_{\substack{j=1\\ j\neq i}}^{c}\sum_{k=1}^{n_i}\sum_{l=1}^{n_j} \boldsymbol{x}_k^{(i)} \boldsymbol{y}_l^{(j)T} \\ &= \left(\boldsymbol{X1}_n\right)\left(\boldsymbol{Y1}_n\right)^T - \boldsymbol{XAY}^T \\ &= -\boldsymbol{XAY}^T \\ &= -C_w \end{aligned}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{1}_{n_1\times n_1} & & & & \\ & \ddots & & & \\ & & \boldsymbol{1}_{n_i\times n_i} & & \\ & & & \ddots & \\ & & & & \boldsymbol{1}_{n_c\times n_c} \end{bmatrix} \in \Re^{n\times n}$$

is a block, diagonal matrix of size *n*-by-*n*.

DCCA

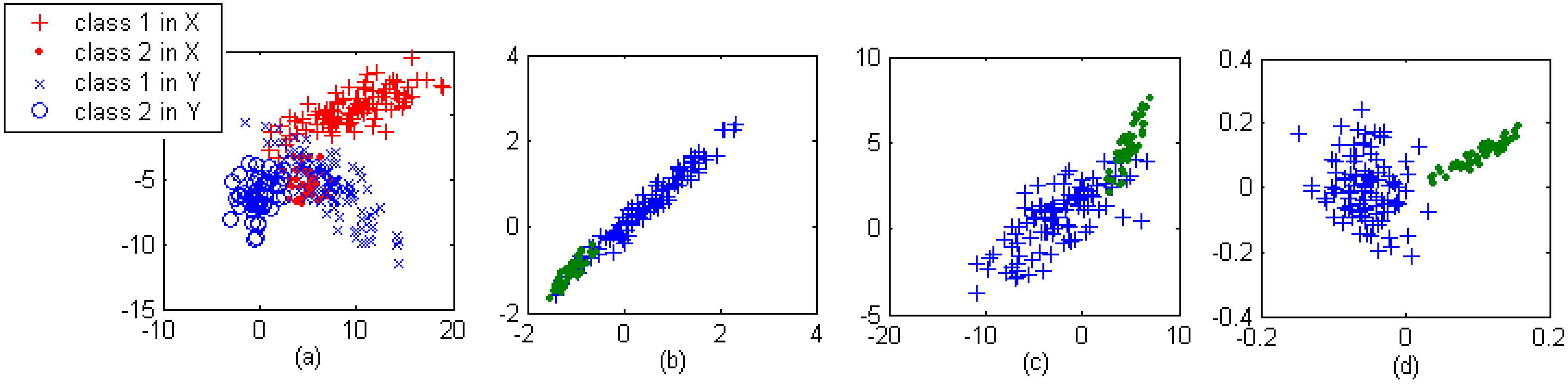
- **Objective function**

$$\max_{\boldsymbol{w}_x,\boldsymbol{w}_y} \frac{(1+\eta)\cdot \boldsymbol{w}_x^T \boldsymbol{C}_w \boldsymbol{w}_y}{(\boldsymbol{w}_x^T \boldsymbol{C}_{xx} \boldsymbol{w}_x \cdot \boldsymbol{w}_y^T \boldsymbol{C}_{yy} \boldsymbol{w}_y)^{1/2}}$$

$$= \max_{\boldsymbol{w}_x,\boldsymbol{w}_y} \frac{\boldsymbol{w}_x^T \boldsymbol{C}_w \boldsymbol{w}_y}{(\boldsymbol{w}_x^T \boldsymbol{C}_{xx} \boldsymbol{w}_x \cdot \boldsymbol{w}_y^T \boldsymbol{C}_{yy} \boldsymbol{w}_y)^{1/2}}$$

$$\Rightarrow \begin{pmatrix} & \boldsymbol{XAY}^T \\ \boldsymbol{YAX}^T & \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{pmatrix} = \lambda \begin{pmatrix} \boldsymbol{XX}^T & \\ & \boldsymbol{YY}^T \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{pmatrix}$$

DCCA Results on toy problem



Data CCA PLS DCCA

$$\boldsymbol{y}_i = \boldsymbol{W}^T \boldsymbol{x}_i + \boldsymbol{b} + \varepsilon_i$$

$\boldsymbol{W} = \begin{bmatrix} 0.6 & -\sqrt{2}/2 \\ 0.8 & \sqrt{2}/2 \end{bmatrix}$ $\boldsymbol{b} = [1,1]^T$ ε_i the imposed Gaussian noise

DCCA Results on real datasets

Hypertext categorization

(WebKB dataset)

Unimodal Recognition		
Method	Recognition Accuracy	
	fulltext	inlinks
Naïve Bayes	0.9083	0.8753
k-NN	**0.9448**	**0.9467**
CMV	0.9098	0.8881

Multimodal Recognition		
Method	Recognition Accuracy	
	PR1	PR2
DCCA	**0.9522**	**0.9574**
CCA	0.9235	0.9213
PLS	0.9215	0.9203

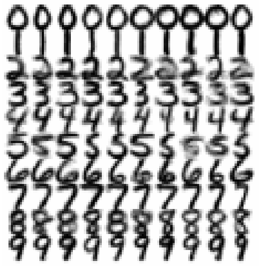
Note: PR1 and PR2 correspond to features in parallel and in serial, respectively.

DCCA Results

Handwritten numeral recognition (Multiple Feature database)

#	X	Y	Recognition accuracy using different methods					
			DCCA		CCA		PLS	
			PR1	PR2	PR1	PR2	PR1	PR2
1	mfeat_fac	mfeat_fou	**0.9560**	**0.9813**	0.8673	0.8785	0.9394	0.9394
2	mfeat_fac	mfeat_kar	**0.9752**	**0.9789**	0.9603	0.9598	0.9410	0.9397
3	mfeat_fac	mfeat_mor	**0.9077**	**0.9302**	0.7596	0.7656	0.8716	0.8789
4	mfeat_fac	mfeat_pix	**0.9718**	**0.9752**	0.9472	0.9476	0.9433	0.9396
5	mfeat_fac	mfeat_zer	**0.9589**	**0.9772**	0.8542	0.8623	0.9521	0.9570
6	mfeat_fou	mfeat_kar	0.9393	0.9687	0.8969	0.9195	**0.9714**	**0.9698**
7	mfeat_fou	mfeat_mor	**0.8089**	**0.8278**	0.7567	0.7633	0.4398	0.4389
8	mfeat_fou	mfeat_pix	0.9373	0.9662	0.8270	0.8431	**0.9761**	**0.9756**
9	mfeat_fou	mfeat_zer	**0.8367**	**0.8543**	0.8239	0.8351	0.8110	0.8119
10	mfeat_kar	mfeat_mor	**0.8928**	**0.9253**	0.7857	0.8158	0.6314	0.6234
11	mfeat_kar	mfeat_pix	0.9493	0.9497	0.9643	0.9641	**0.9751**	**0.9753**
12	mfeat_kar	mfeat_zer	**0.9383**	**0.9638**	0.9081	0.9211	0.8245	0.8289
13	mfeat_mor	mfeat_pix	**0.8799**	**0.9100**	0.7263	0.7602	0.7071	0.7078
14	mfeat_mor	mfeat_zer	**0.7943**	**0.8097**	0.7258	0.7452	0.6983	0.7154
15	mfeat_pix	mfeat_zer	**0.9310**	**0.9544**	0.8232	0.8398	0.8375	0.8401

Note: PR1 and PR2 correspond to features in parallel and in serial, respectively.



http://www.ics.uci.edu/~mlearn/MLSummary.html

The works about DCCA have been submitted.

Our new work

- CIPCA
- DCCA
- DCCAM
- Multi-Kernel learning
- Beyond KCCA

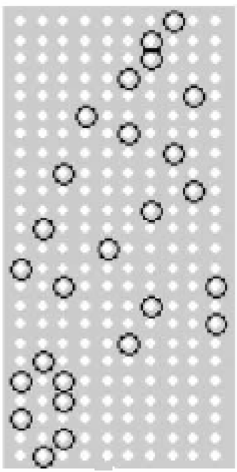
DCCAM

- **Motive**

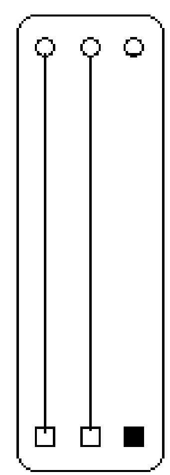
  1) deal with missing samples due to sensor failure, high cost, etc.

  2) "missing samples" means "not pairwise" for X and Y

  3) differ from "missing data" (or "incomplete data")



Missing data



Missing sample

DCCAM

- **Usual strategies for missing data**

  1) expectation maximization [DHS00]

  2) substitution of class mean for missing data

  3) iterative estimation of the missing data [WM01]

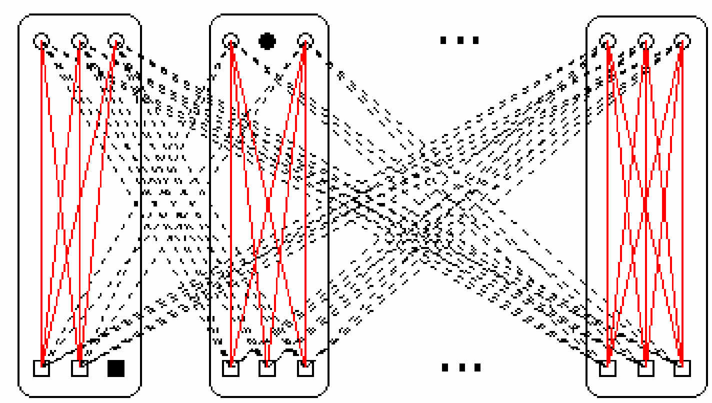
  4) local least squares imputation (估算) [KGP05]

- **Our solution**

  directly deal with the training set with missing samples

DCCAM

- **Objective**



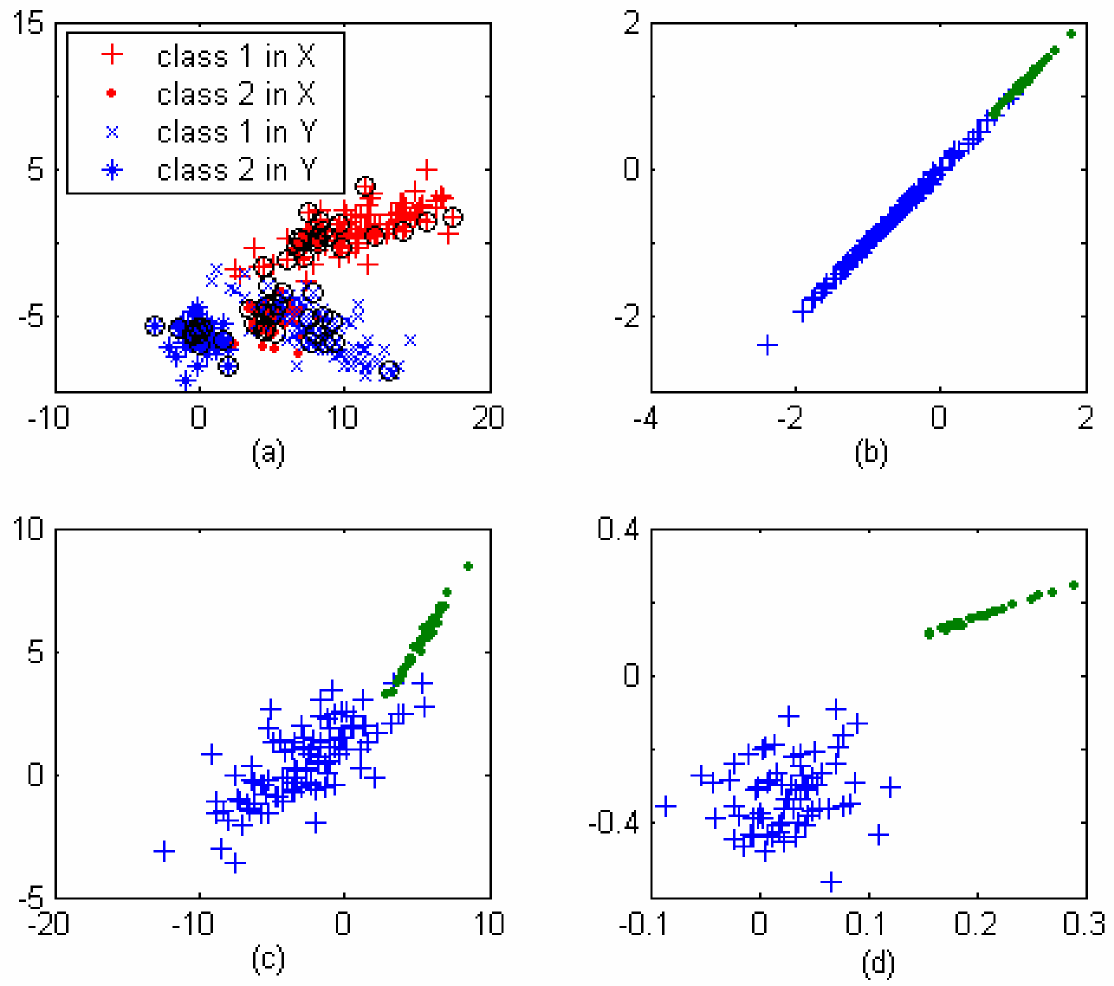
$$\max_{\boldsymbol{w}_x, \boldsymbol{w}_y} \frac{\boldsymbol{w}_x^T (\boldsymbol{C}_w - \eta \boldsymbol{C}_b) \boldsymbol{w}_y}{(\boldsymbol{w}_x^T \boldsymbol{C}_{xx} \boldsymbol{w}_x \cdot \boldsymbol{w}_y^T \boldsymbol{C}_{yy} \boldsymbol{w}_y)^{1/2}}$$

$$\Rightarrow \begin{pmatrix} & \boldsymbol{XAY}^T \\ \boldsymbol{YAX}^T & \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{pmatrix} = \lambda \begin{pmatrix} \boldsymbol{XX}^T & \\ & \boldsymbol{YY}^T \end{pmatrix} \begin{pmatrix} \boldsymbol{w}_x \\ \boldsymbol{w}_y \end{pmatrix}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{1}_{m_1 \times n_1} & & & & \\ & \ddots & & & \\ & & \boldsymbol{1}_{m_i \times n_i} & & \\ & & & \ddots & \\ & & & & \boldsymbol{1}_{m_c \times n_c} \end{bmatrix} \in \Re^{m \times n}$$

is a block, diagonal matrix of size *m*-by-*n*.

DCCAM Results on Toy problem



(a) Data

(b) CCA

(c) PLS

(d) DCCAM

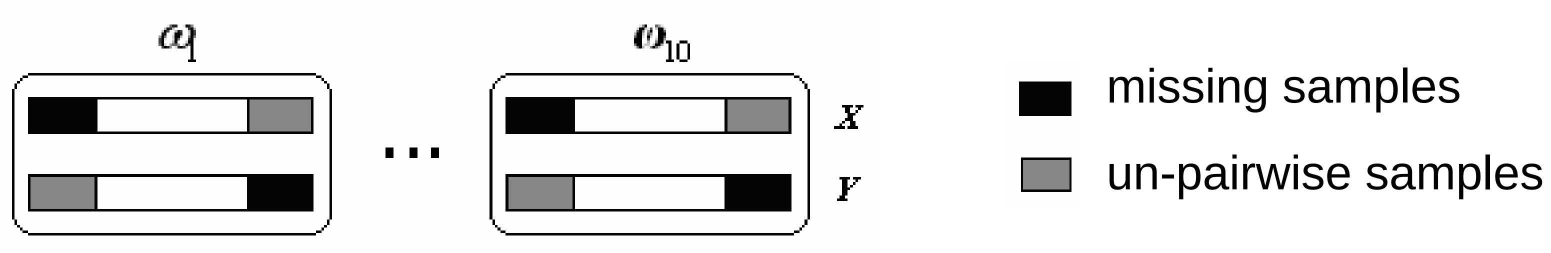
Circled samples are missing ones. For (b) CCA and (c) PLS, the un-pairwise samples are deleted.

DCCAM Results on Real datasets

Multiple Feature Database (handwritten numerals), 10 classes, 100 pairs per class. For each class, we

1) delete 10 samples from X, and delete 10 samples from Y.
2) delete 20 samples from X, and delete 20 samples from Y.
3) delete 30 samples from X, and delete 30 samples from Y.
4) delete 40 samples from X, and delete 40 samples from Y.
5) delete 50 samples from X, and delete 50 samples from Y.
6) delete 60 samples from X, and delete 60 samples from Y.

Note: the deleted samples are NOT pairwise.



DCCAM Results on Real datasets

- **Adjustment of the comparative algorithms**

For CCA and PLS, the following algorithm is adjusted to match the case of "missing samples"

1) delete the rest un-pairwise samples

(named as CCA-I, PLS-I) and DCCA-I.

2) substitution of class mean vector for missing samples

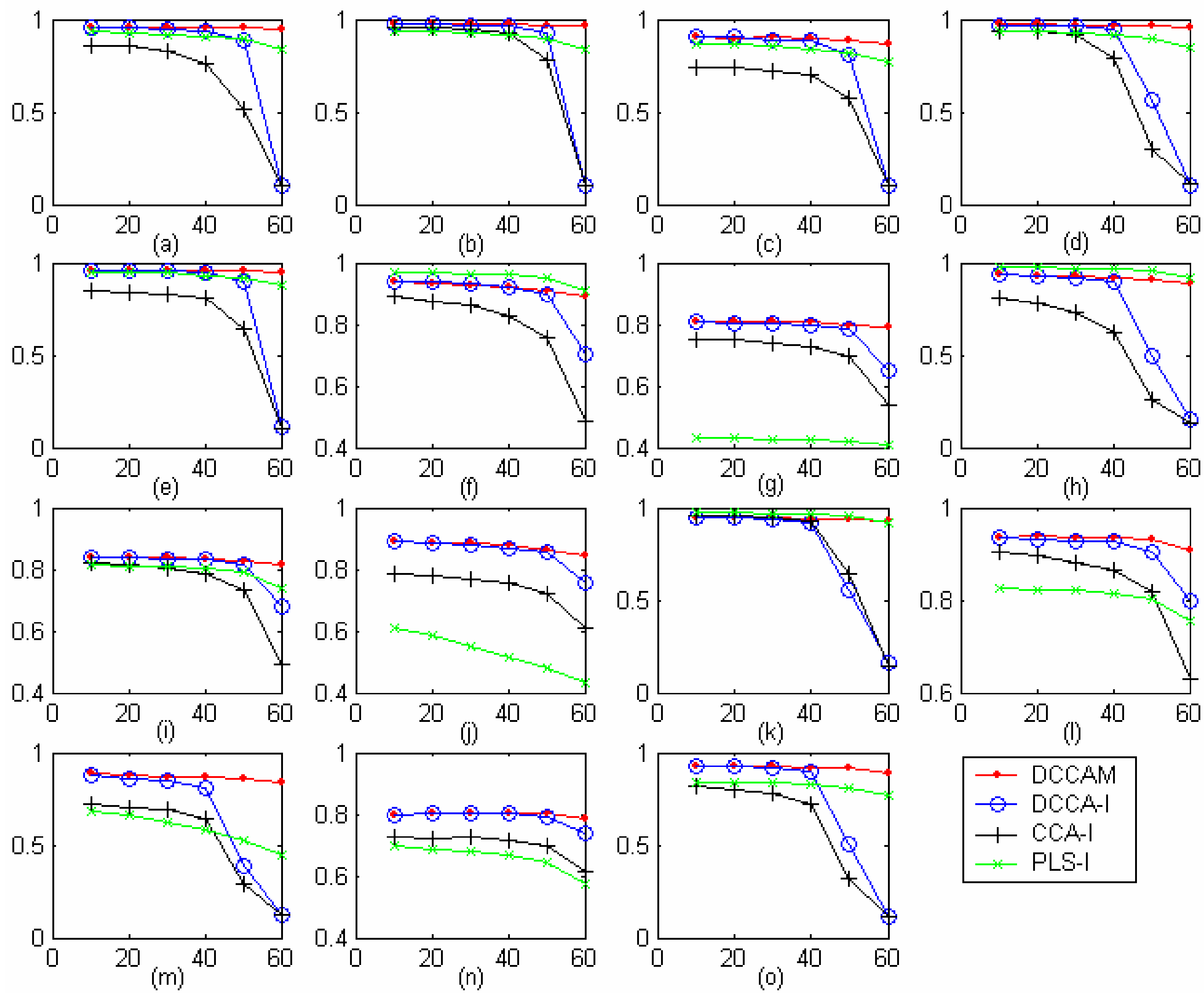
(named as CCA-II, PLS-II) and DCCA-II.

3) iterative estimation of the missing samples

(named as CCA-III, PLS-III).

Note that DCCA-I and –II are just for comparison, and no DCCA-III is introduced.

DCCAM Results on Real datasets



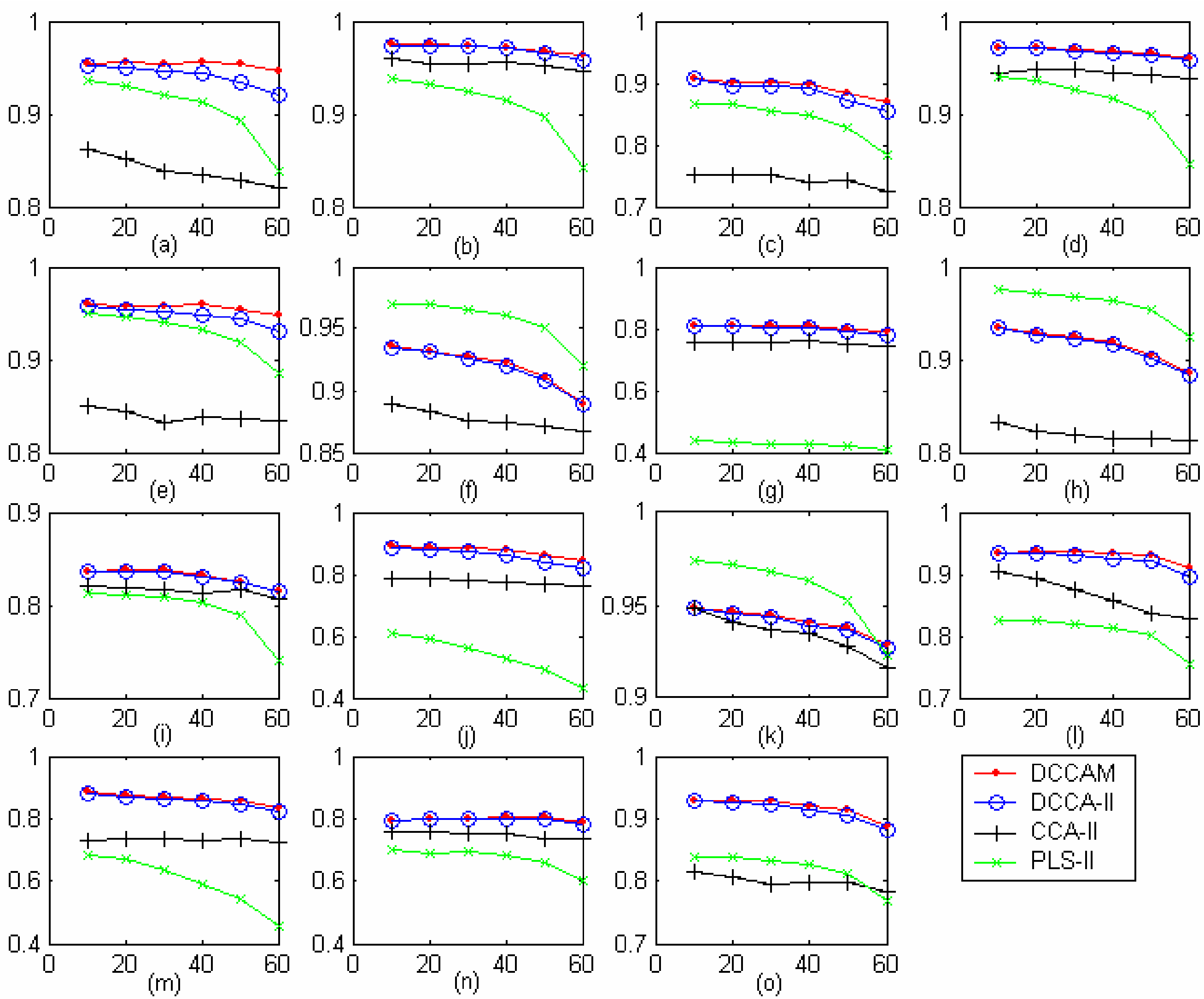
X-axis:

the # of missing samples per class. i.e., 10, 20,…50, 60.

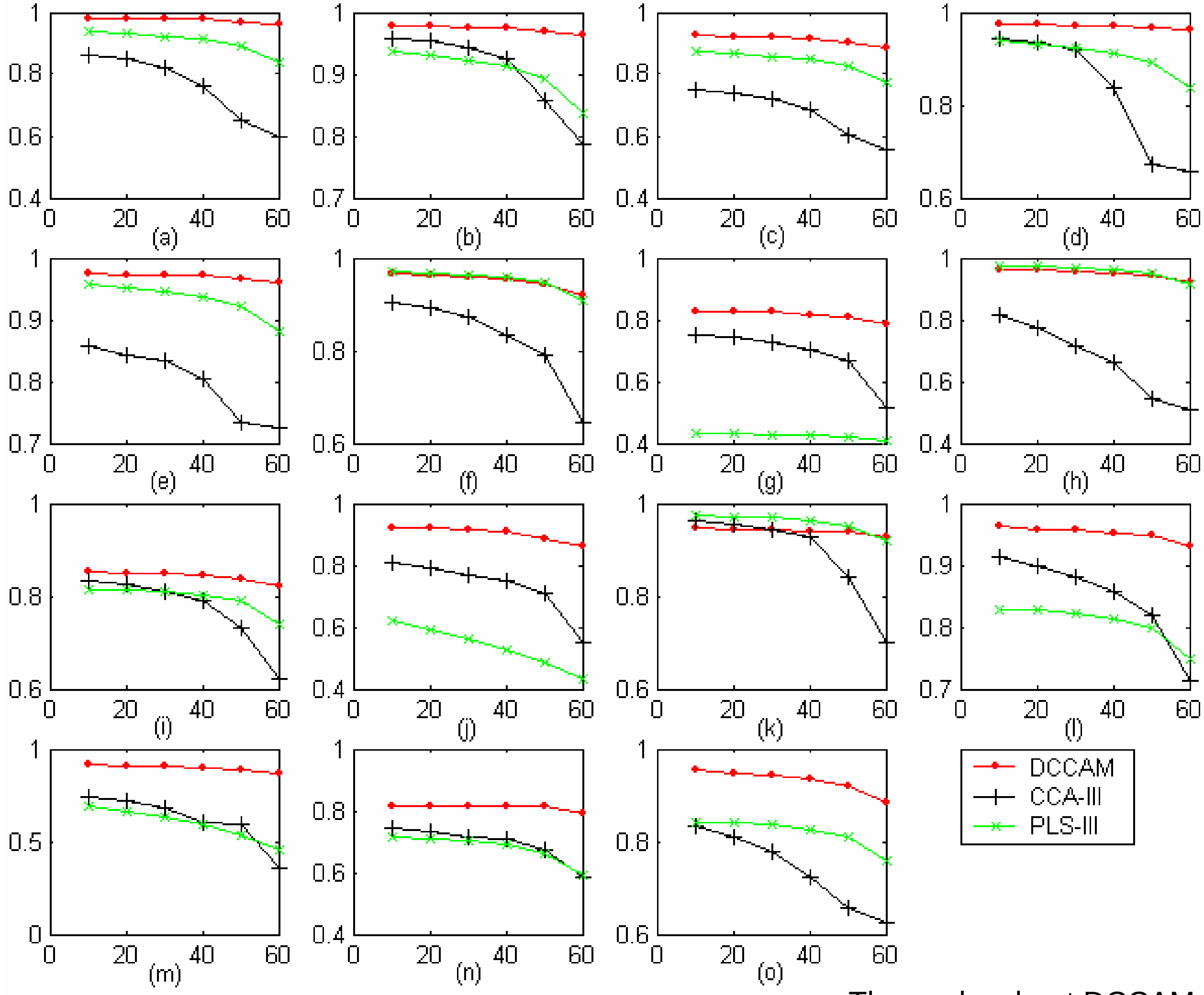
Y-axis:

the recognition accuracy.

DCCAM Results on Real datasets



DCCAM Results on Real datasets



The works about DCCAM have been submitted.

DCCAM advantages

- Supervised learning method due to the incorporation of the class information
- Tolerance to the missing of samples
- Superior performance to some multimodal recognition algorithms
- Direct processing ability of missing samples and without resorting to artificially compensate the missing samples
- Relative insensitiveness to the number of missing samples
- Only involvement of one tunable parameter, *d*, the final dimensionality of the extracted features. This makes it easy to be manipulated in real applications.

Our new work

- CIPCA
- DCCA
- DCCAM
- Multi-Kernel learning
- Beyond KCCA

Multi-Kernel Learning (MKL)

- **Motive:**

  1) approximate heterogeneous (异质) data sources;

  2) avoid model or kernel parameter selection to some extent;

- **The related work [BLJ04, SRS05]**

  The conic combinations of multiple kernels

$$K = \sum_i \alpha_i K_i$$

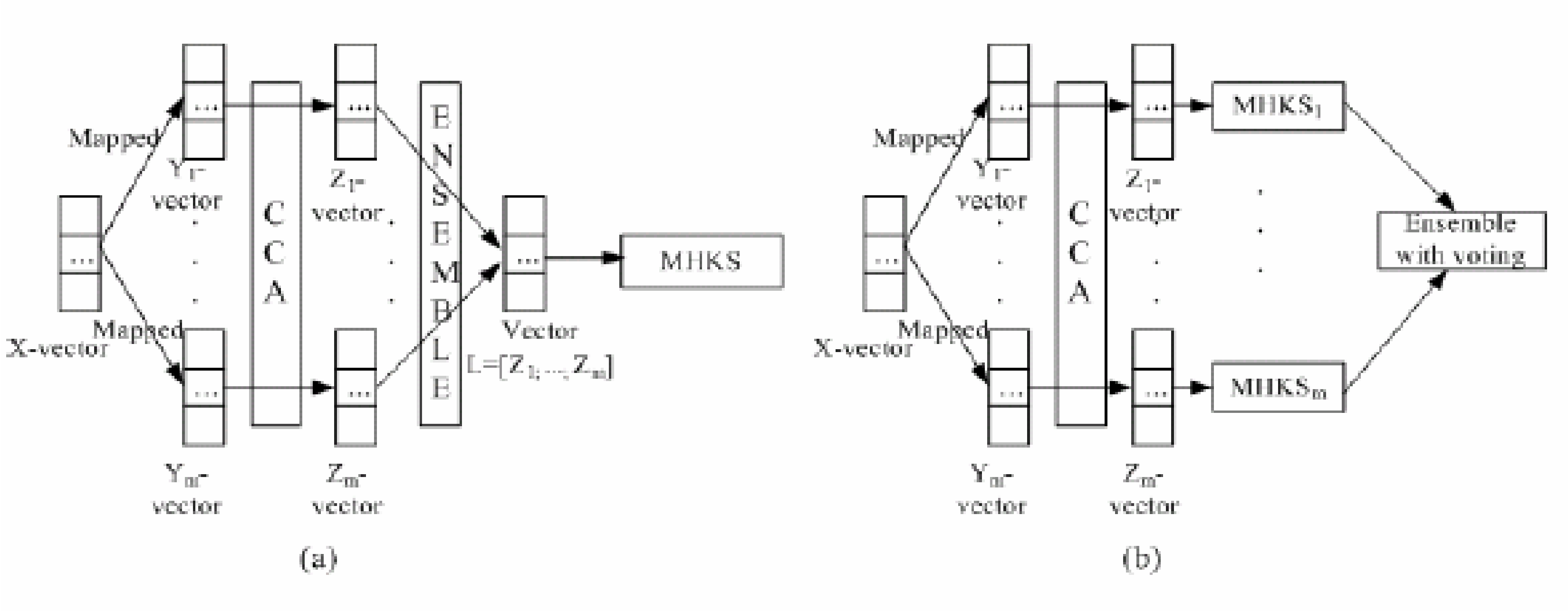
F. Bach, G.R.G. Lanckriet, and M.I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the 21st International Conference on Machine Learning*, 2004.

S. Sonnenburg, G. Ratsch, and C. Schafer. A general and efficient multiple kernel learning algorithm. In *Neural Information Processing Systems*, 2005.

Multi-Kernel Learning (MKL)

- **Our Method**

  borrow multiple CCA (mCCA) to fuse multiple kernels



Essence: Single dataset → Multi-Kernel mapping

+ Multi-CCA (NmCCA)→ MKL

Multi-Kernel Learning (MKL)

mCCA

$$\min_{W^{(1)},\ldots,W^{(m)}} \quad J = \sum_{k,l=1;k\neq l}^{m} \|S^{(k)}W^{(k)} - S^{(l)}W^{(l)}\|_F$$

$$s.t. \quad W^{(k)T}C_{kk}W^{(k)} = I;$$

$$w_i^{(k)T}C_{kl}w_j^{(l)} = 0;$$

$$k,l = 1,\ldots,m, l \neq k; i,j = 1,\ldots,q, j \neq i;$$

NmCCA

$$\min_{\substack{\omega_l \in \mathbb{R}^{n_l+1}, b_l>0;\\ l=1,\ldots,m}} L = \sum_{l=1}^{m}((Y_l\omega_l - I_{N\times 1} - b_l)^T(Y_l\omega_l - I_{N\times 1} - b_l) + c_l\tilde{\omega}_l^T\tilde{\omega}_l)$$

$$+c_{m+1}\sum_{l=1}^{m}(Y_l\omega_l - \frac{1}{m}\sum_{j=1}^{m}Y_j\omega_j)^T(Y_l\omega_l - \frac{1}{m}\sum_{j=1}^{m}Y_j\omega_j),$$

Least Squares Criterion

New Multi-CCA (NmCCA)

MKL Results compared to [SRS05]

Classification performance comparison

Datasets	MultiK-MHKS	CCA+MHKS1	CCA+MHKS2	MHKS(best)	MHKS(worst)	MKL[21]
Soybean-small	**99.57**	90.00	88.26	96.52	94.35	*97.83*
Balance	**93.94**	87.88	87.88	90.16	90.10	*90.19*
Water	**96.21**	57.27	63.79	*94.85*	94.70	93.03
Sonar	**80.65**	72.78	72.78	*75.83*	75.74	75.28
Wdbc	**63.89**	*63.48*	62.96	62.48	59.04	50.81
Pima-diabetes	*71.40*	70.83	70.20	69.54	62.91	**71.43**
Iris	**97.33**	92.67	90.53	**97.33**	97.07	**97.33**
Wine	**96.89**	89.81	93.02	*95.19*	88.96	79.15

MKL Results compared to [SRS05]

t-test comparison on MultiK-MHKS with the following algorithms

Datasets	CCA+MHKS1	CCA+MHKS2	MHKS(best)	MHKS(worst)	MKL[21]
Soybean-small	5.3906	4.5810	2.2778	3.4330	2.0580
Balance	10.6106	10.6106	6.8245	6.2137	6.5939
Water	19.2638	24.3671	1.4443*	1.4128*	2.6877
Sonar	6.8811	6.8811	4.4371	3.2984	2.3520
Wdbc	1.0855*	2.5204	2.7287	4.1504	9.8425
Pima-diabetes	3.0534	2.9210	2.5185	27.5278	1.0000*
Iris	6.2163	8.9938	0*	0.3612*	0*
Wine	4.8455	3.8432	2.6056	8.9000	3.8048

'*' Denotes that the difference between the two corresponding algorithms is not significant at 5% significance level, i.e. *t*-value < 1.7341.

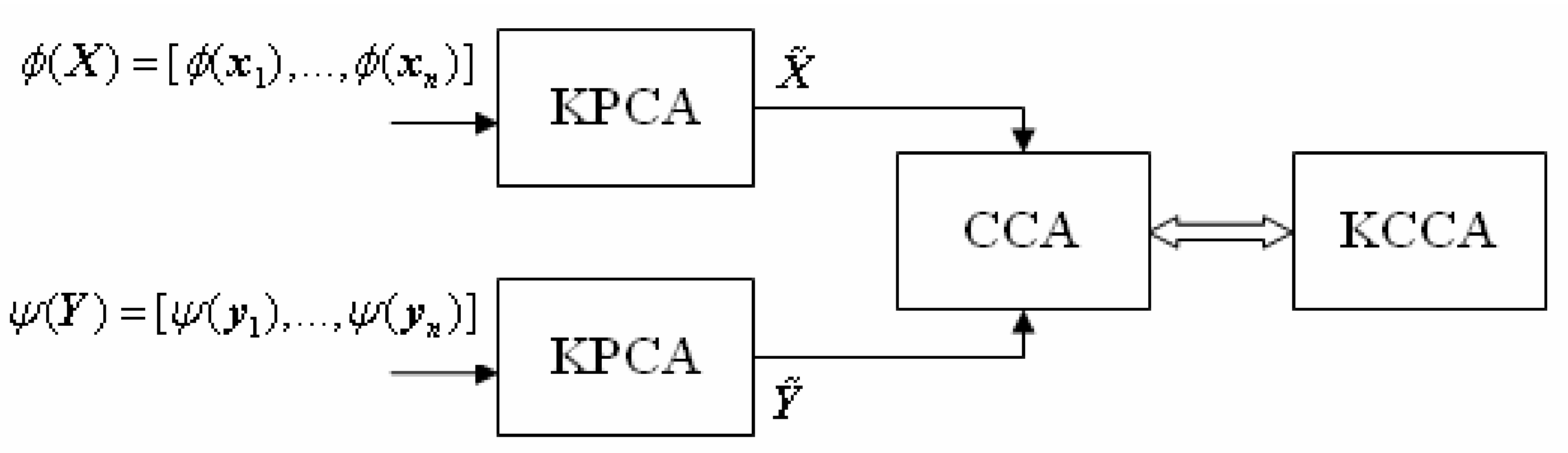
The works about MKL have been submitted.

Our new work

- CIPCA
- DCCA
- DCCAM
- Multi-Kernel learning
- Beyond KCCA

Beyond KCCA

- **Proof of "KCCA = 2-KPCA + CCA"**



Beyond KCCA

Insight from "KCCA = 2-KPCA + CCA"

- **Reveal the essence of KCCA**
- **What is important …**

1) the original kernel mappings can be generalized to empirical kernel mappings, i.e.,

$$\phi : \boldsymbol{x} \mapsto \phi(\boldsymbol{x}) \text{ and } \psi : \boldsymbol{y} \mapsto \psi(\boldsymbol{y})$$
$$\Rightarrow \boldsymbol{x} \mapsto \boldsymbol{K}_X(\boldsymbol{X}, \boldsymbol{x}) \text{ and } \boldsymbol{y} \mapsto \boldsymbol{K}_Y(\boldsymbol{Y}, \boldsymbol{y})$$

Beyond KCCA

2) More generally

$$\boldsymbol{x} \mapsto F_X(\boldsymbol{X}, \boldsymbol{x}) \text{ and } \boldsymbol{y} \mapsto F_Y(\boldsymbol{Y}, \boldsymbol{y})$$

where both F_x and F_y are non-negative real functions.

- **Advantages**

  Any types of kernel can be employed,

  e.g., graph-kernel, string- kernel, tree-kernel, and etc.,

  even non-kernel functions

The works about Beyond KCCA have been submitted.

References

- [And00] S. Ando. Image field categorization and edge/corner detection from gradient covariance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(2):179-190.
- [Bor99] Canonical correlation: a tutorial, at http://people.imt.liu.se/~magnus/cca/tutorial/, 1999
- [DHS00] R.O. Duda, P.E. Hart, D.G. Stock, Pattern Classification, 2nd edition, John Wiley & Sons, New York, 2000.
- [WM01] Walczak, B. and D.L. Massart, Dealing with missing data Part I. Chemometrics & Intelligent Laboratory Systems, vol.58, no.1, pp.15-27. 2001
- [For04] B. Fortuna. Kernel canonical correlation analysis with applications. In: SIKDD 2004 at Multiconference, Ljubljana, Slovenia. 2004.
- [He04] Y. Hel-Or, The canonical correlations of color images and their use for demosaicing, HP Labs Technical Report, HPL-2003-164(R.1), February 2004.

References

- [HSS04] D.R. Hardoon, S. Szedmak, J. Shawe-Taylor. Canonical correlation analysis: an overview with application to learning methods. Neural Computation 2004, 16: 2639-2664.
- [JRS04] Anil K. Jain, Arun Ross and Salil Prabhakar,An Introduction to Biometric Recognition,IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Image- and Video-Based Biometrics, Vol. 14, No. 1, January 2004.
- [KGP05] H. Kim, G.H. Golub and H. Park, Missing value estimation for DNA microarray gene expression data: local least squares imputation, Bioinformatics, vol.21, 187-198, 2005
- [KSE05] E. Kidron, Y.Y. Schechner, M. Elad. Pixels that sound, IEEE Proc. of Computer Vision and Pattern Recognition, 2005: 1, 88-95.
- [LGD05] M. Loog, B. Ginneken, R.P.W. Duin. Dimensionality reduction of image features using the canonical contextual correlation projection. Pattern Recognition, 2005,38(12): 2409-2418

References

- [Nie02] A. Nielsen. Multiset canonical correlations analysis and multispectral, truly multitemporal remote sensing data. IEEE Transactions on Image Processing, 2002,11: 293-305.
- [SS06] P. Schreier, L. Scharf, Canonical coordinates for transform coding of noisy sources, IEEE Transactions on signal processing, 2006, 54(1):235-243.
- [SZL05] Q. Sun, S. Zeng,Y. Liu, P-A. Heng, D-S. Xia. A new method of feature fusion and its application in image recognition. Pattern Recognition, 2005, 38(12): 2437-2448.
- [YVN03] Y. Yamanishi, J.-P. Vert, A. Nakaya1, et al. Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis. Bioinformatics, 2003, 19(1), pp. i323–i330.

Thanks!