Adaptive Similarity Measures for Material Identification in Hyperspectral Imagery

Ph.D. Defense

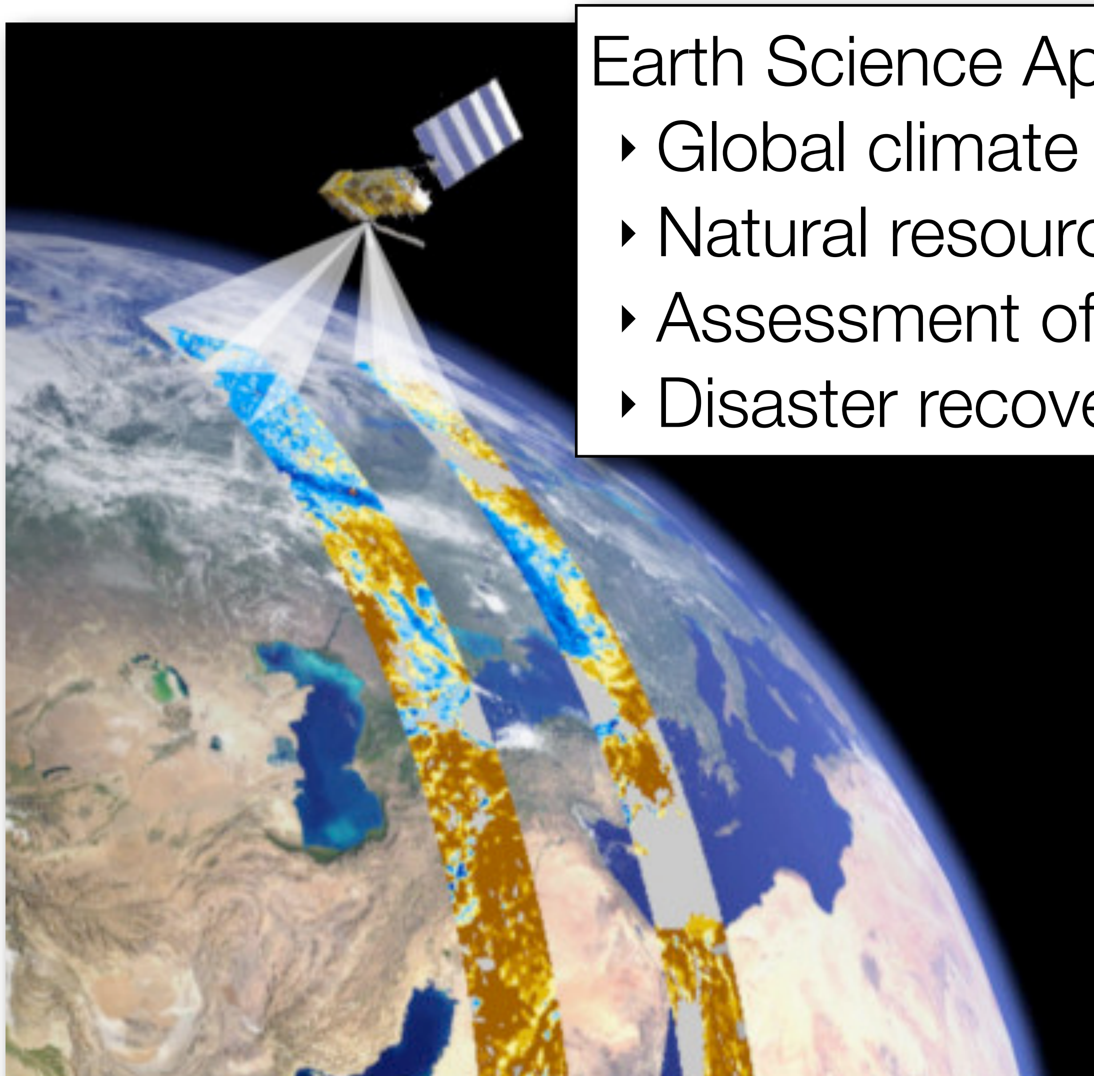
Brian D. Bue (bbue@rice.edu)
Department of Electrical and Computer Engineering

Committee:
Prof. Erzsébet Merényi (chair), Statistics/ECE
Prof. Chris Jermaine, CS
Prof. Devika Subramanian, CS/ECE
Dr. Kiri Wagstaff, Jet Propulsion Laboratory

This work involves research and development carried out at the Jet Propulsion Laboratory, and for the AMMOS MultiMission Operations System and MGSS Instrument Operations Subsystem. AMMOS and U.S. Government Support Acknowledged.

Remote Sensing
Imagery from satellite and airborne sensors

"*The vast majority of our current knowledge on the geology of solar system objects has been derived from remote sensing measurements.*" [Bell et al. 2001]



Earth Science Applications:
- Global climate change monitoring
- Natural resource management
- Assessment of natural hazards
- Disaster recovery efforts

Figure: Univ. Valencia, IPL

Figure: NASA/GSFC

Planetary Science Applications:
- Analysis of geologic processes
- Finding conditions hospitable to life (e.g., liquid water)

Identifying materials from imagery crucial to all of these applications

Hyperspectral Imagery

Captured in hundreds of bands in the visible & near-infrared portion of the electromagnetic spectrum

Pixels ("*spectra*") capture detailed material characteristics

Each image contains millions of **high-dimensional** pixels

Modern missions produce Terabyte-sized datasets

Automated methods necessary to summarize material content and flag interesting observations

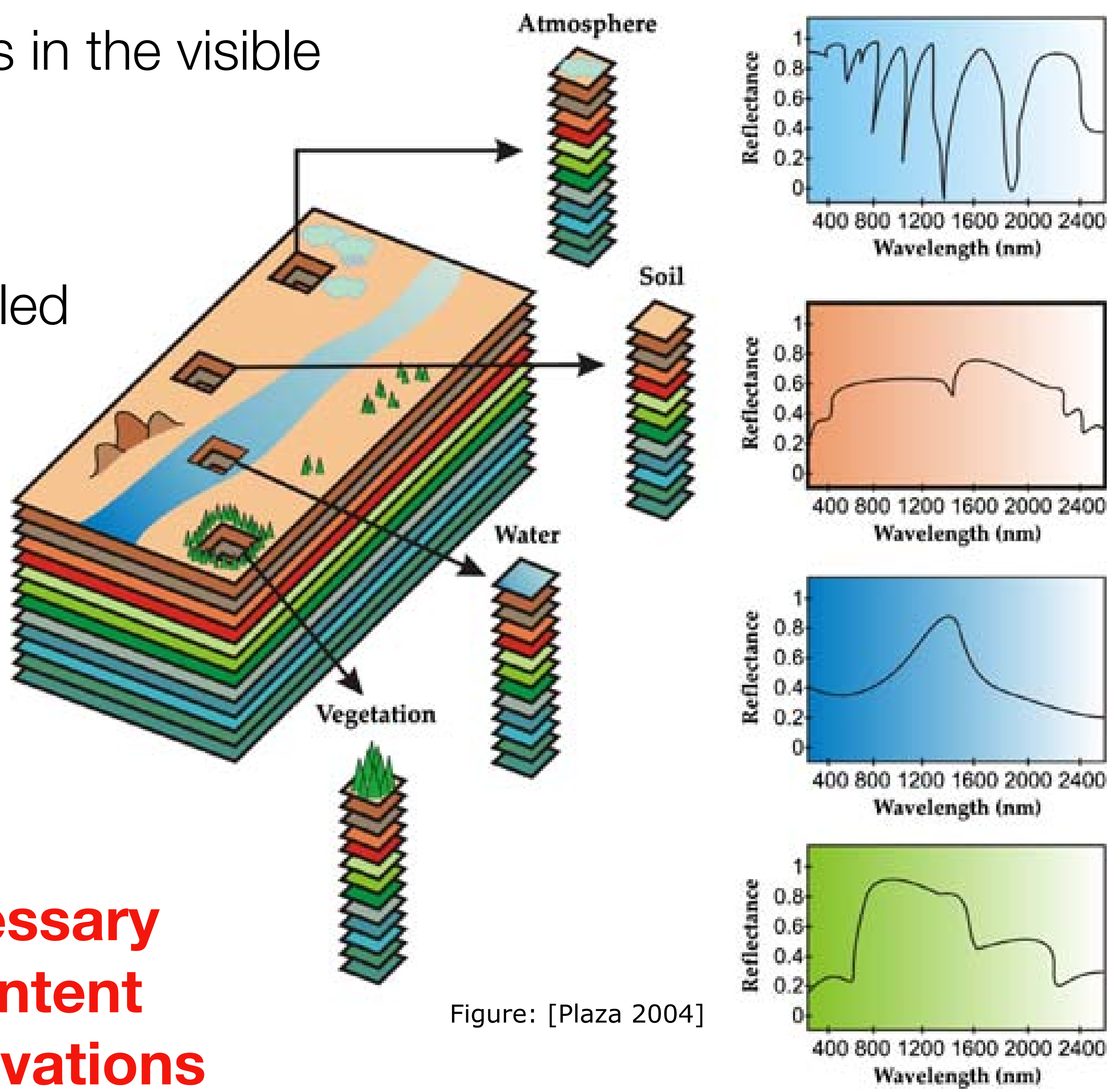


Figure: [Plaza 2004]

Thesis Objective and Contributions

Objective: develop **adaptive, task-specific similarity measures** for automatic material identification in remotely-sensed hyperspectral imagery

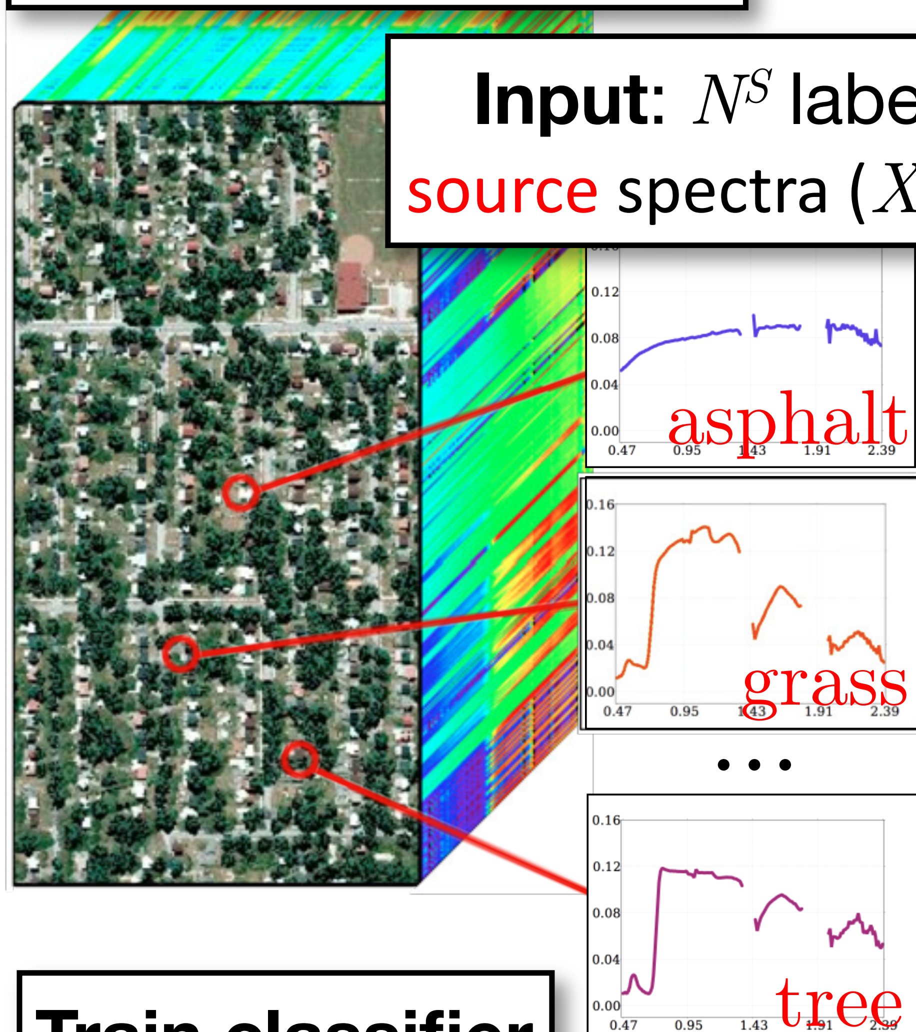
Contributions:

- Material identification with library-based spectral matching
- Intra-domain material identification
  - Hybrid-LDA: method to combine several similarity measures
  - Evaluation of Mahalanobis metric learning techniques
- Inter-domain material identification
  - New framework for supervised/unsupervised domain adaptation
  - Comparisons to multitask learning/manifold alignment methods
- Demonstrated results in practical, real-world classification settings

Material Identification as Multiclass Classification

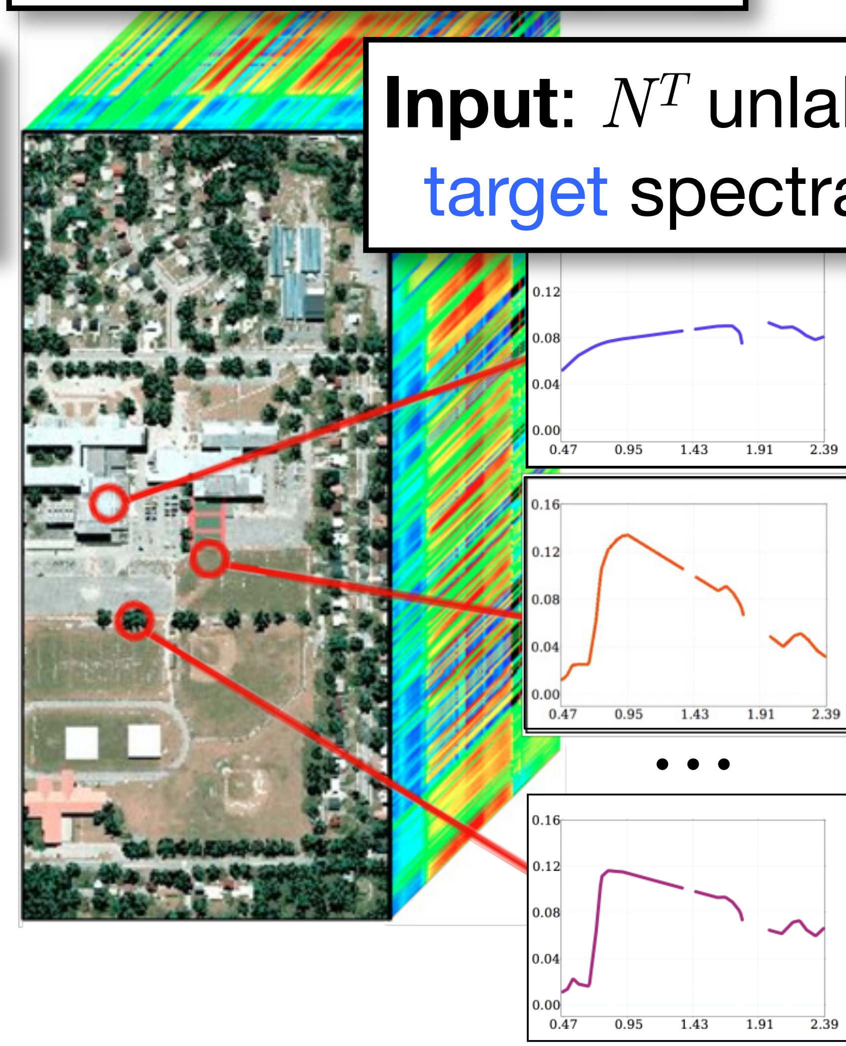
Source domain ($\mathcal{D}^S$)

Input: N^S labeled source spectra (X^S, Y^S)



Target domain ($\mathcal{D}^T$)

Input: N^T unlabeled target spectra X^T



asphalt

grass

tree

Train classifier

$$\left\{\left(\mathbf{x}_i^S, y_i^S\right)\right\}_{i=1}^{N^S} \rightarrow f(\mathbf{x}) = y$$

Output: target predictions Y^T

Intra-domain Material Identification

Source and target spectra captured under identical conditions ⇒ $\mathcal{D}^S = \mathcal{D}^T$

good source classifier ⇒ good target predictions

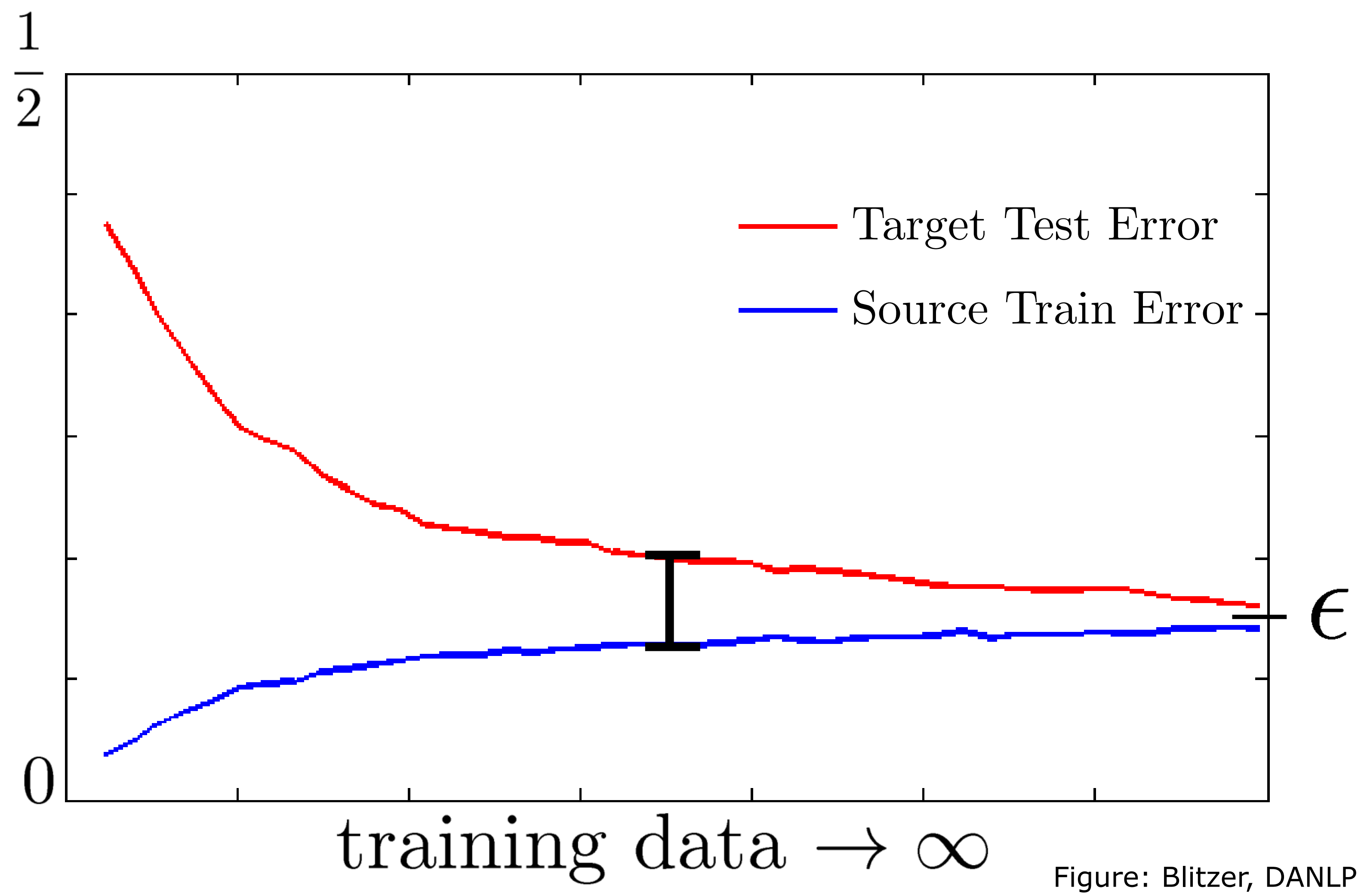


Figure: Blitzer, DANLP '10

Material Identification Fundamentals
Absorption Features and Continua

Can derive information about the chemistry of a material from its spectral signature

Positions and widths of absorption bands typify materials

Continuum removal characterizes absorption bands

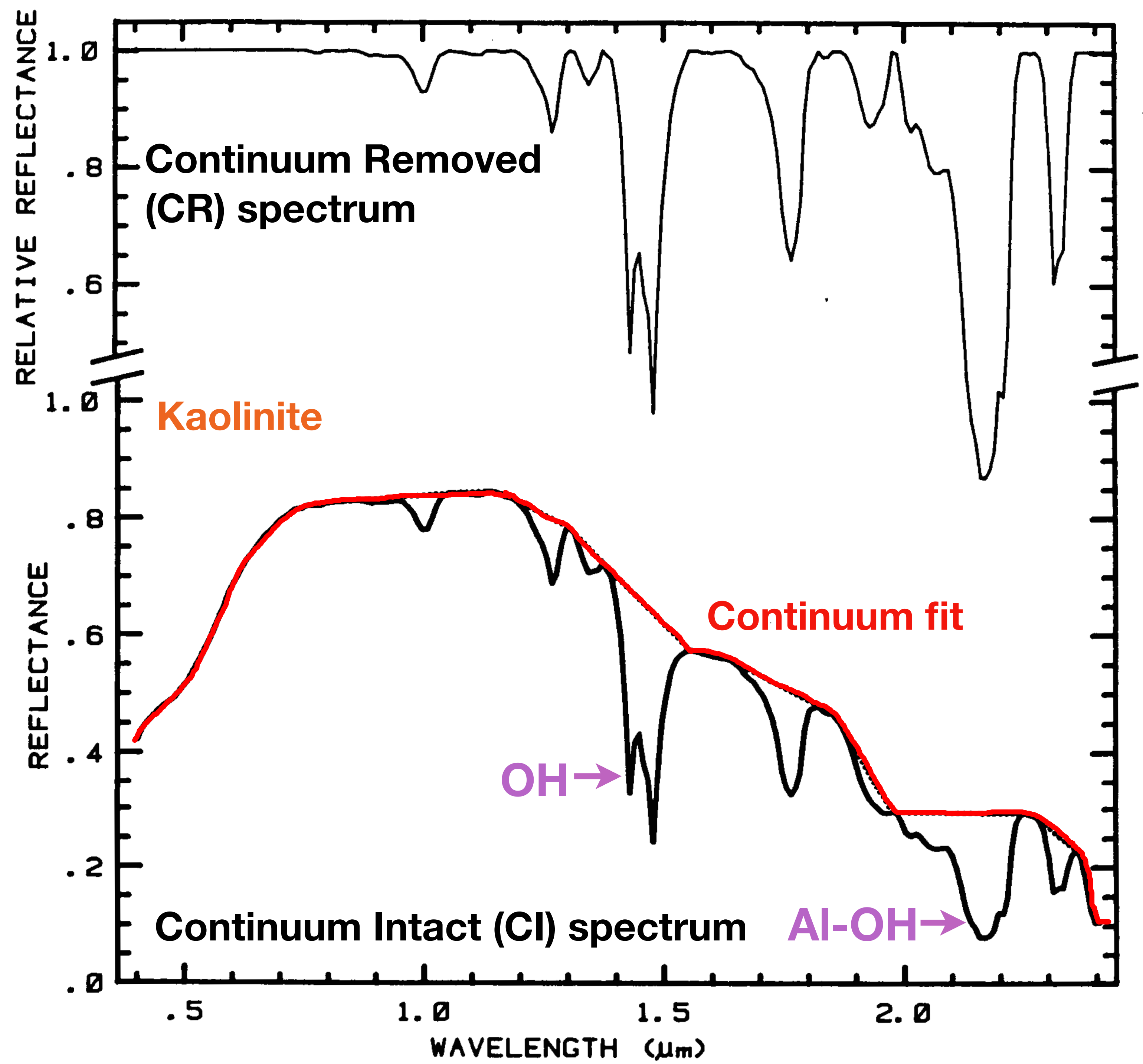


Figure modified after: [Clark 90]

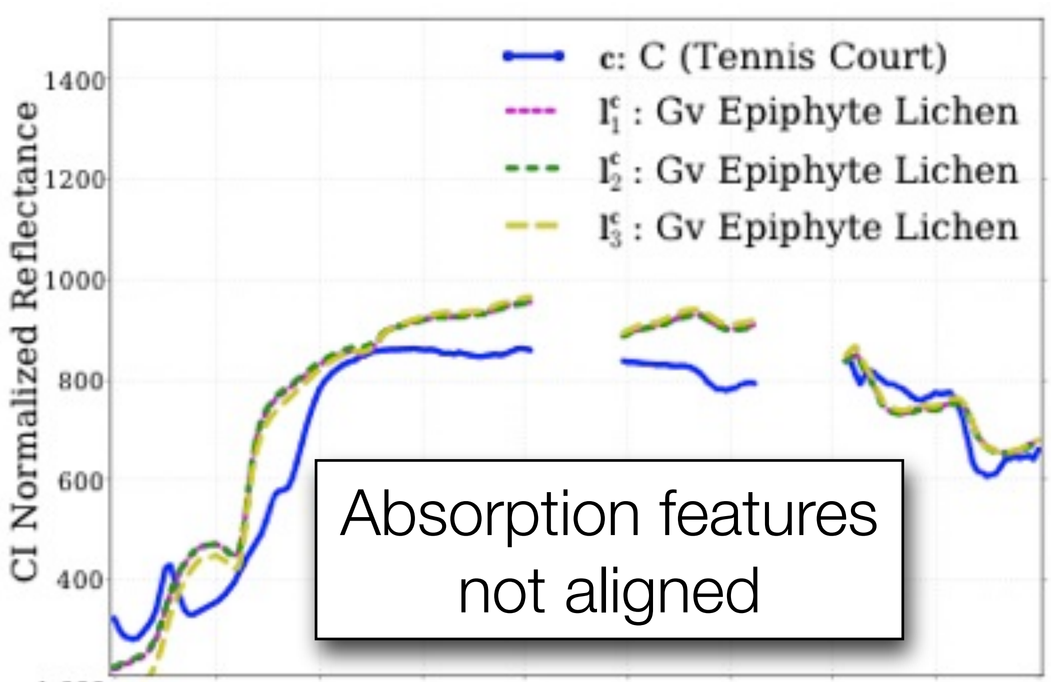
Library-based Spectral Matching
CI vs. CR Distances

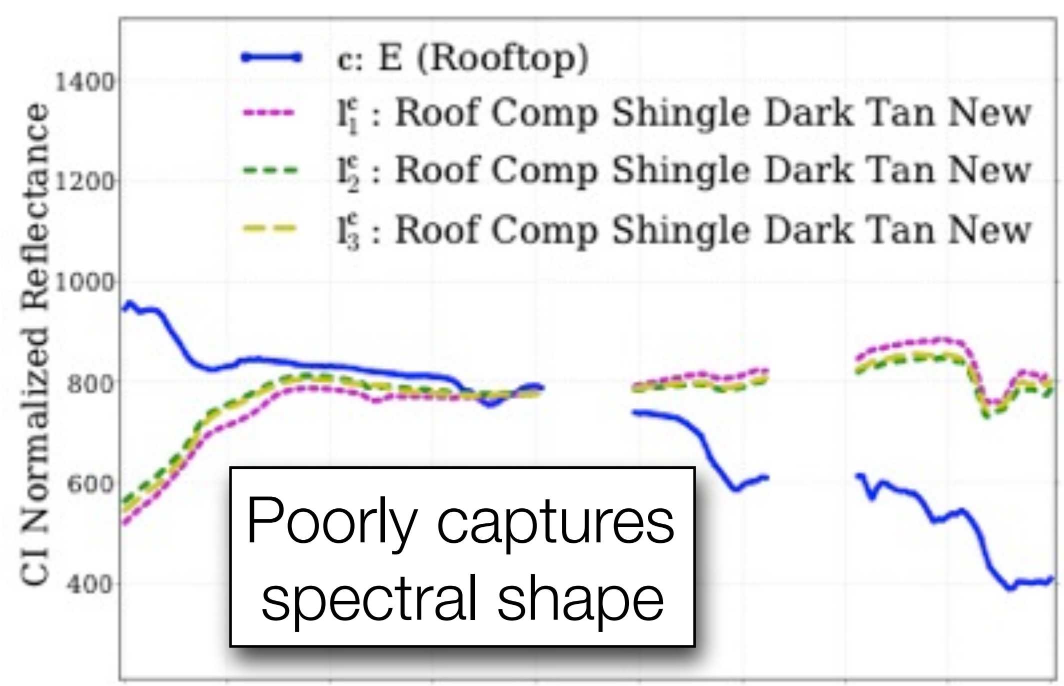
Distance between CI spectra

$$d_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) = \left\| \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|} - \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|} \right\|$$

Distance between CR spectra

$$d_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) = \left\| \frac{\mathrm{CR}(\mathbf{x}_i)}{\|\mathrm{CR}(\mathbf{x}_i)\|} - \frac{\mathrm{CR}(\mathbf{x}_j)}{\|\mathrm{CR}(\mathbf{x}_j)\|} \right\|$$



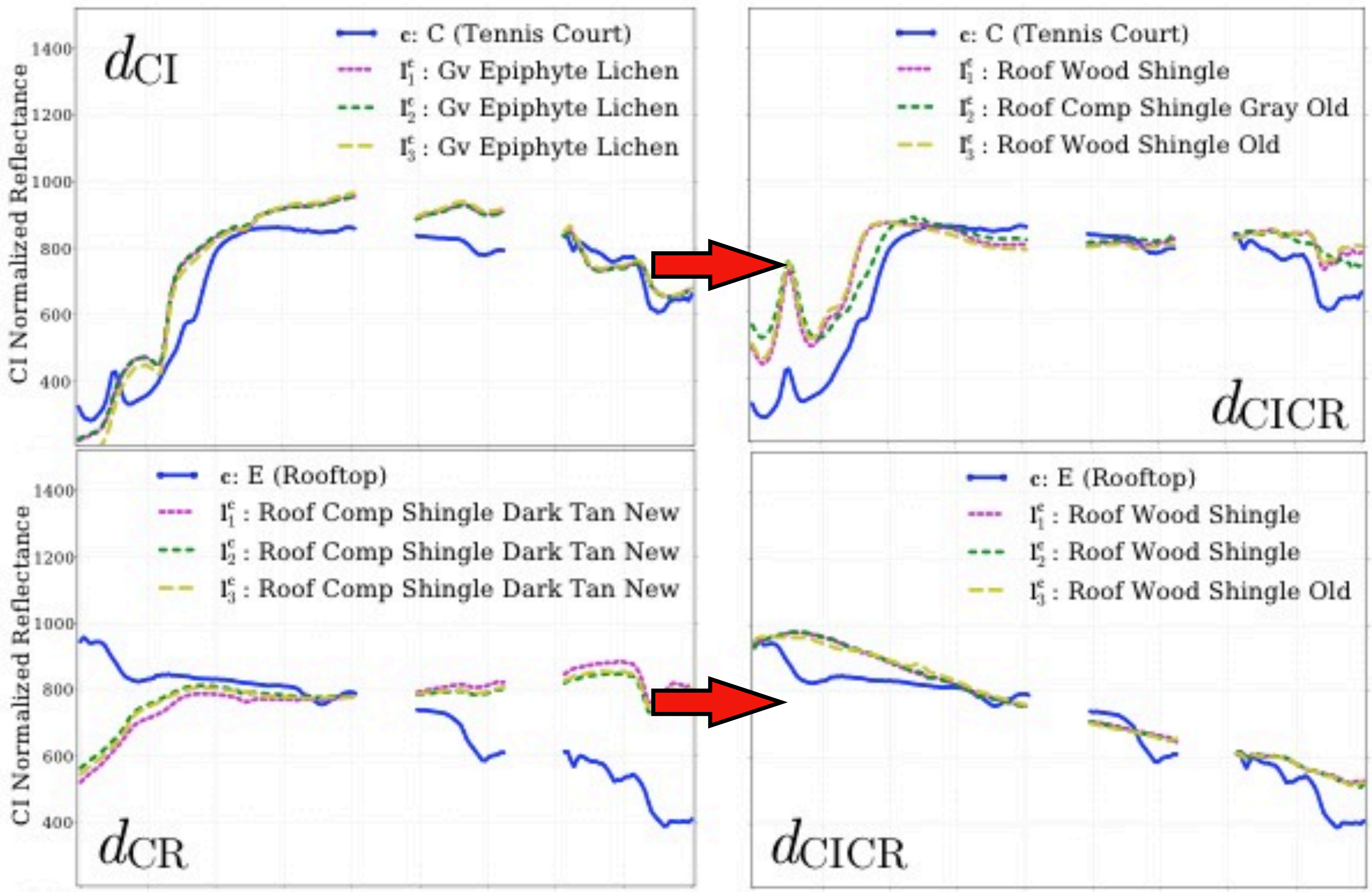


Blue line=target signature, dashed lines=best matchng source signatures

Spectral Representation
Continuum-Intact vs. Continuum Removed

Solution: measure CI shape and CR absorption features

$$d_{\mathrm{CICR}}(\mathbf{x}_i, \mathbf{x}_j, \alpha) = (1-\alpha)d_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) + \alpha d_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j)$$

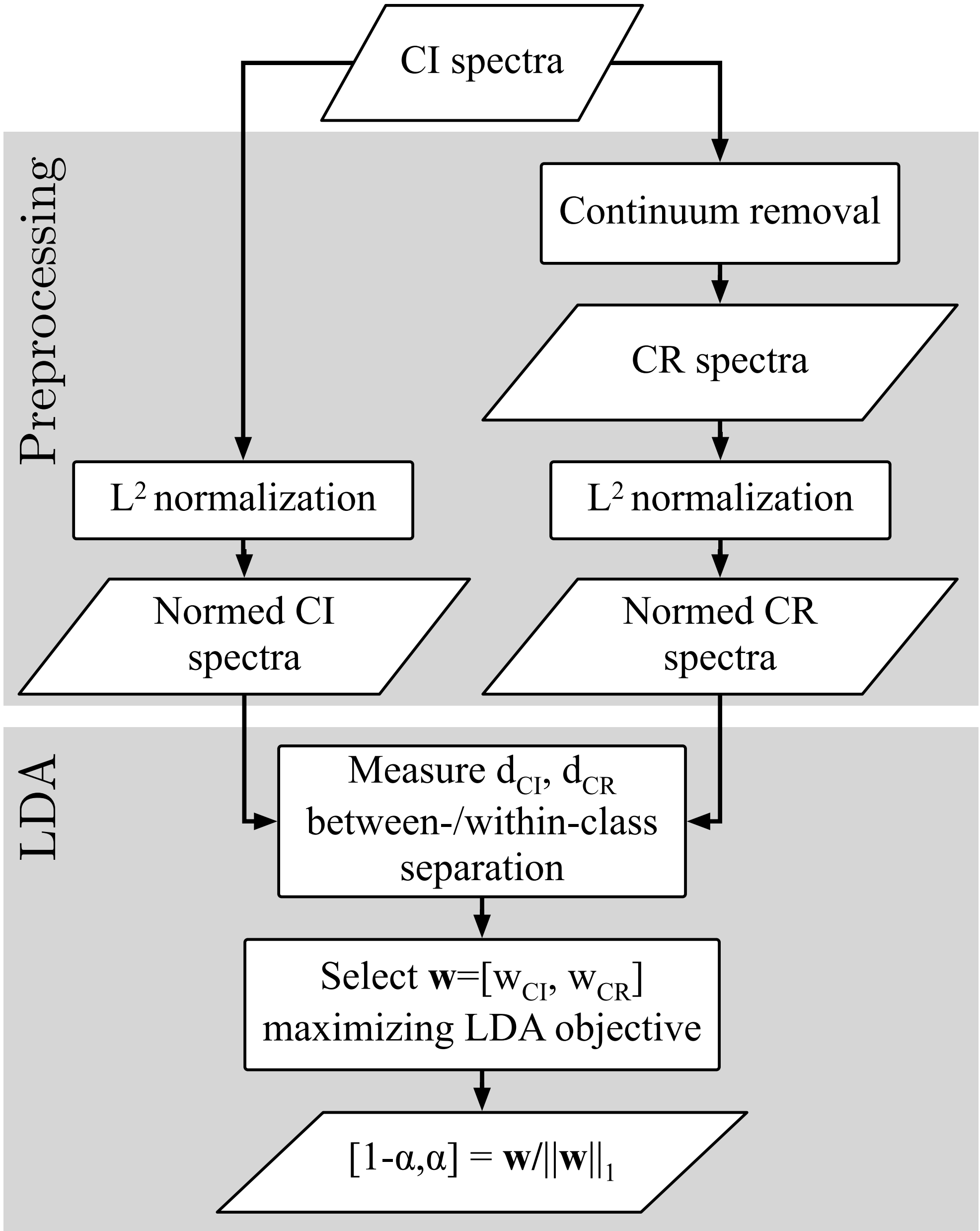


Hybrid-Linear Discriminant Analysis (LDA)

Example: Adaptive CICR [Bue and Merényi, IEEE GRSL 2012]

Goal: Learn scalar weight parameter α from labeled data

$$d_{\mathrm{CICR}}(\mathbf{x}_i, \mathbf{x}_j, \alpha) = (1-\alpha)d_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) + \alpha d_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j)$$



LDA objective: calculate projection matrix $\mathbf{W}^*$ maximizing

Between-class separation

$$\mathbf{W}^* = \underset{\mathbf{W}}{\operatorname{argmax}} \left\{ \frac{|\mathbf{W}^T \mathbf{M}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{M}_W \mathbf{W}|} \right\}$$

Within-class scatter

[Fisher, 1936]

Solution: $[1-\alpha, \alpha] = \mathbf{w}/||\mathbf{w}||_1$

$\mathbf{w}$ = largest eigenvector of $\mathbf{M}_W^{-1}\mathbf{M}_B$

Hybrid-LDA
Within and Between Class Separation

For distance measures $d_1(\mathbf{x}_i, \mathbf{x}_j)$ and $d_2(\mathbf{x}_i, \mathbf{x}_j)$:

$$[\mathbf{M}_W]_{1,2} = [\mathbf{M}_W]_{2,1} = \frac{1}{N}\sum_{j=1}^{K}\sum_{i:y_i=j} d_1(\mathbf{x}_i, \boldsymbol{\mu}_j) d_2(\mathbf{x}_i, \boldsymbol{\mu}_j)$$

$\Rightarrow$ d_1 vs. d_2 ***compactness***

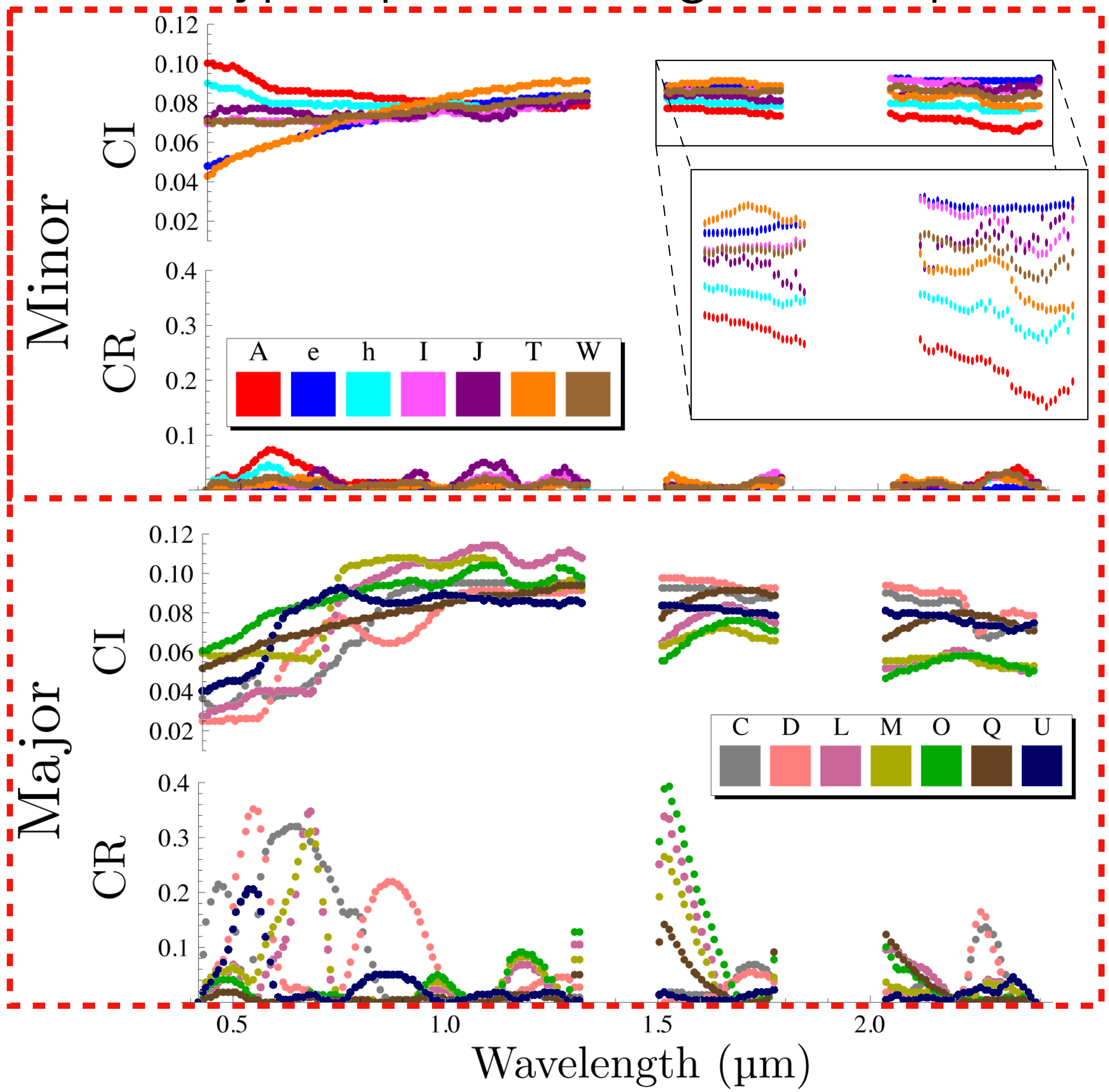
$$[\mathbf{M}_B]_{1,2} = [\mathbf{M}_B]_{2,1} = \frac{1}{N}\sum_{j=1}^{K} N_j d_1(\boldsymbol{\mu}_j, \overline{\boldsymbol{\mu}}) d_2(\boldsymbol{\mu}_j, \overline{\boldsymbol{\mu}})$$

$\Rightarrow$ d_1 vs. d_2 ***separability***

K = # classes, N = # samples, N_j = # samples in j^{th} class,
$\boldsymbol{\mu}_j$ = j^{th} class mean, $\overline{\boldsymbol{\mu}}$ = mean($\boldsymbol{\mu}_j$)

Evaluation on Ocean City AVIRIS Image
Minor/Major/Combined Absorption Scenarios

AVIRIS hyperspectral image, 158 spectral bands



Minor Absorption Scenario:

- 7 classes of flat spectra with few absorptions
- CR representation uninformative ($\alpha \rightarrow 0$)

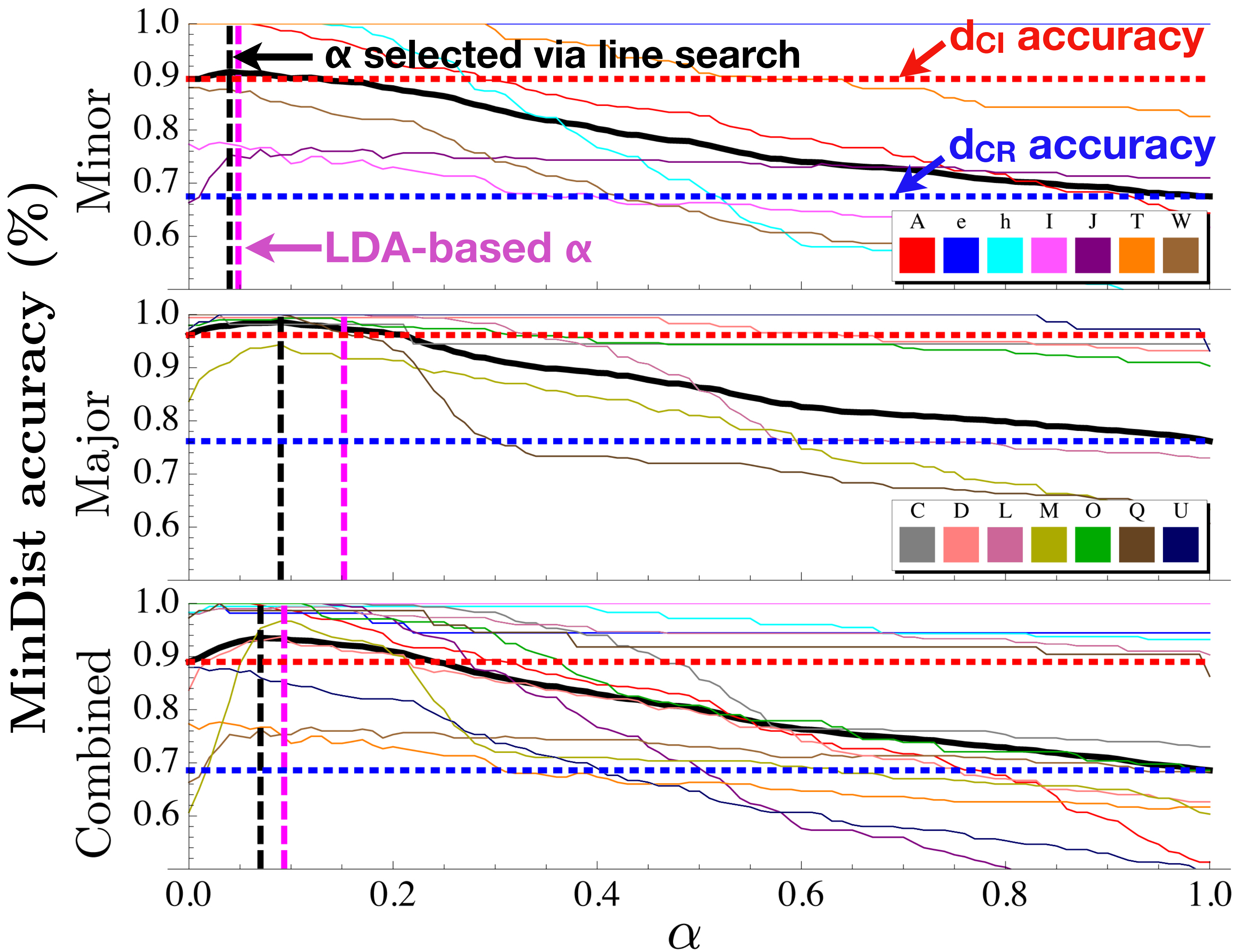
Major Absorption Scenario:

- 7 classes with several deep absorptions
- CR representation discriminative ($\alpha \rightarrow 1$)

Combined Scenario:

- All Major+Minor classes
- $\alpha_{Minor} \leq \alpha_{Combined} \leq \alpha_{Major}$

Evaluation on Ocean City AVIRIS Image
LDA vs. Line Search α



Curves: average (**black**) and per-class (colored) accuracy using CICR

Adaptive Sobolev Metric

Spectral signatures are **functional** data: $\mathbf{REFLECTANCE_b}(\boldsymbol{\lambda})$

$(\mathbf{b}$: band, $\boldsymbol{\lambda}$: wavelength$)$

- High correlation between adjacent spectral bands
- Exploit functional characteristics to improve classification accuracy

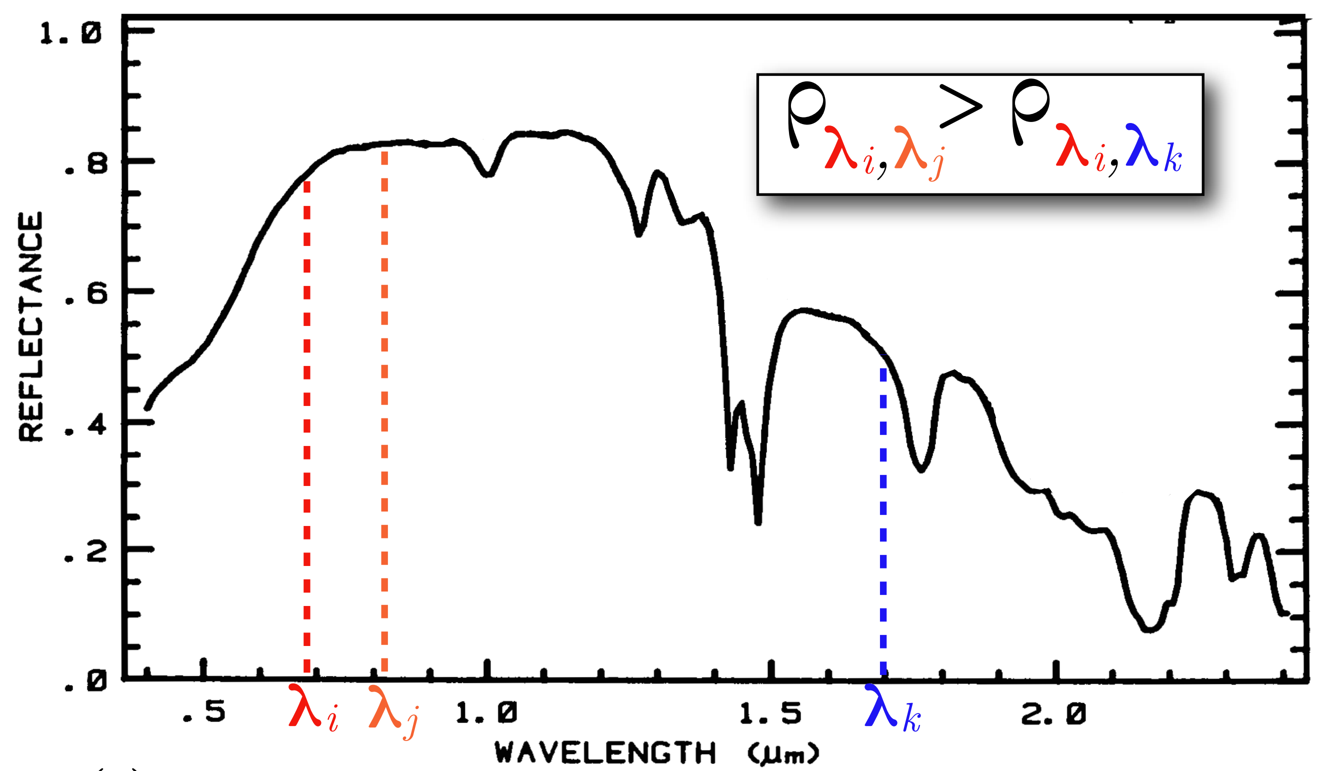


Figure modified after: [Clark 90]

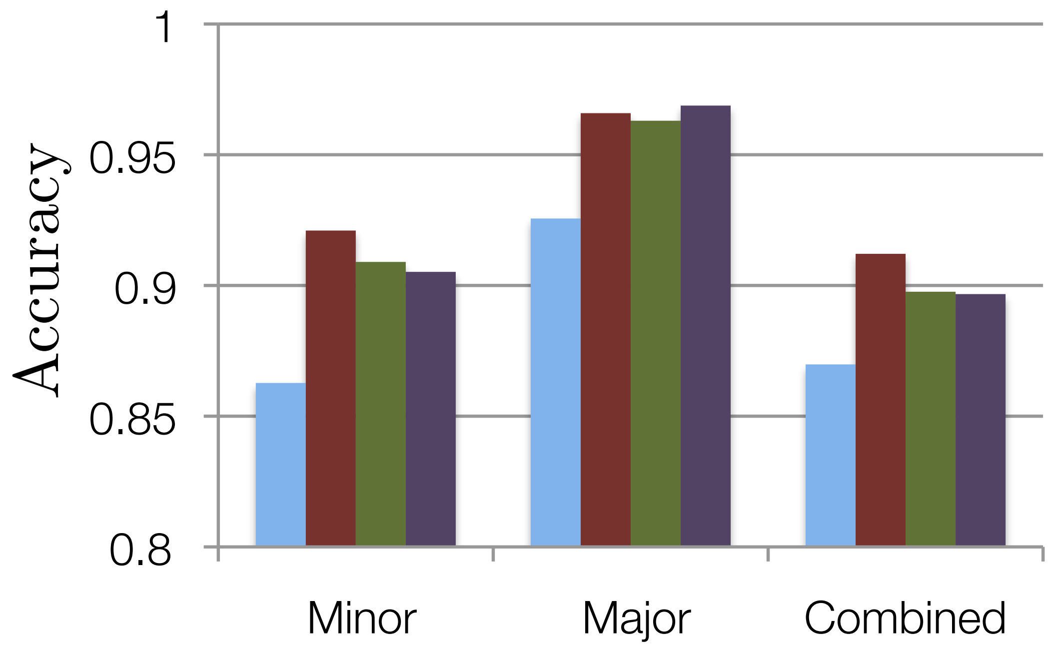
Approach: classify using Sobolev metric with Hybrid-LDA weights α_l

$$\mathrm{d}_{\mathrm{Sobolev}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{l=0}^{\kappa} \gamma_l \alpha_l \, \mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j)$$

- $\mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)}\|$, $\mathbf{x}_i^{(l)} = l^{\text{th}}$ derivative of $\mathbf{x}_i$
- $\gamma_l = 1/\mathrm{stddev}(\mathrm{d}^{(l)})$

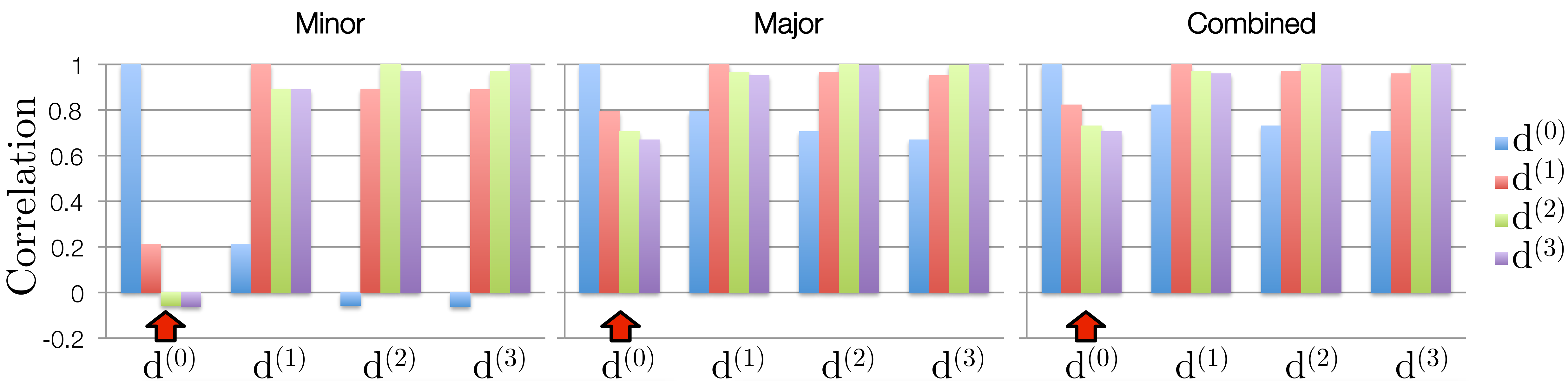
Adaptive Sobolev Metric

Accuracy vs. Per-derivate Correlation



$d^{(0)}$ (Euclidean)
d_{Sobolev} (κ=1)
d_{Sobolev} (κ=2)
d_{Sobolev} (κ=3)
(κ = **# derivatives**)

5-8% improvement for $\kappa = 1$
No improvement for $\kappa > 1$



Only $d^{(0)}$ substantially uncorrelated with other $d^{(l)}$

Distances must capture complementary (i.e., uncorrelated) notions of similarity!

Feature-weighted (Mahalanobis) Metric Learning

Goal: Learn task-specific Mahalanobis metric

$$\begin{aligned} \mathrm{d}_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) &= (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \\ &= \mathrm{d}_{\mathbf{Euc}}(\mathbf{A}^T \mathbf{x}_i, \mathbf{A}^T \mathbf{x}_j), \end{aligned}$$

$\mathbf{M} = \mathbf{A}\mathbf{A}^T = n$ x n positive semidefinite matrix

$\mathbf{A} = n$ x r matrix

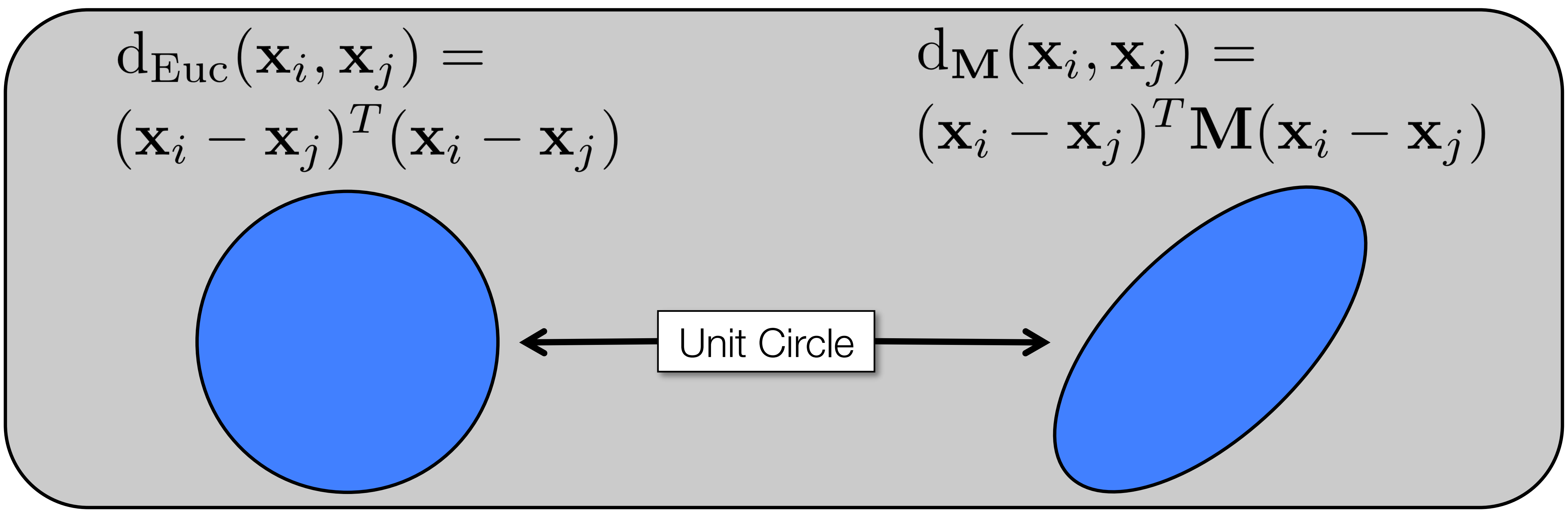


Figure modified after: [Weinberger 07]

Feature-weighted Metric Learning
Multiclass LDA for Mahalanobis Metric Learning

Feature-weighted, multiclass LDA

$$\mathbf{M} = \mathbf{A}\mathbf{A}^T, \quad \mathbf{A} = \underset{\mathbf{W}}{\operatorname{argmax}} \left\{ \frac{|\mathbf{W}^T \mathbf{M}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{M}_W \mathbf{W}|} \right\}$$

where

$$\mathbf{M}_B = \frac{1}{K} \sum_{j=1}^{K} (\boldsymbol{\mu}_j - \boldsymbol{\mu})(\boldsymbol{\mu}_j - \boldsymbol{\mu})^T$$ **(between-class separation)**

$$\mathbf{M}_W = \frac{1}{NK} \sum_{j=1}^{K} \sum_{i:y_i=j} (\mathbf{x}_i - \boldsymbol{\mu}_j)(\mathbf{x}_i - \boldsymbol{\mu}_j)^T$$ **(within-class scatter)**

Regularization:

$\mathbf{M}^*_W = (1\text{-}\gamma)\mathbf{I} + \gamma\mathbf{M}_W$, $\gamma \in [0,1]$, $\mathbf{I}$ = identity matrix

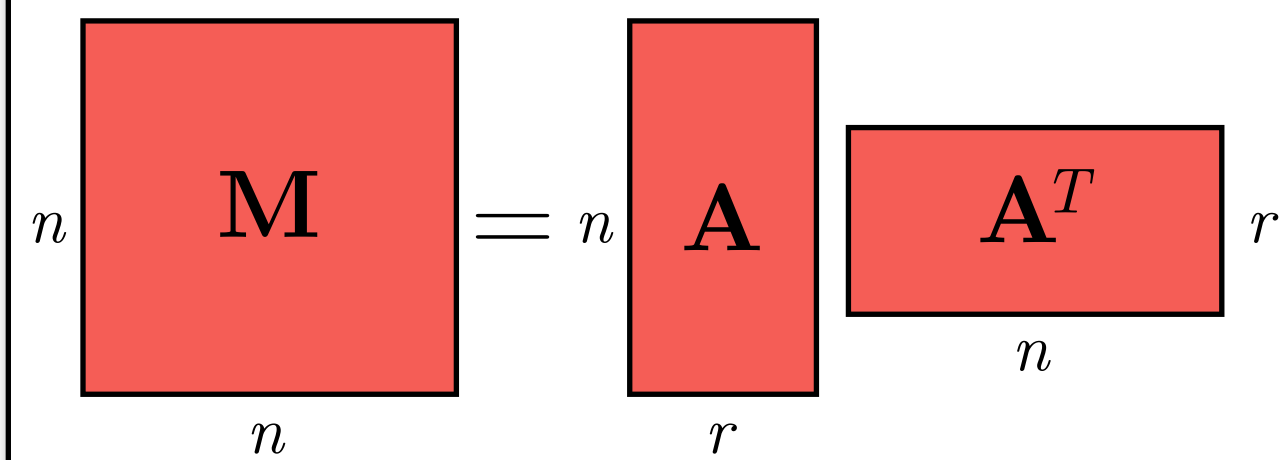
$\mathbf{A}$ = top K-1 eigenvectors of $(\mathbf{M}^*_W)^{-1}\mathbf{M}_B$

Feature-weighted Metric Learning
So...why LDA?

- Closed form solution = **FAST**
- Performs particularly well for **low-rank** Mahalanobis metric learning
- Often more accurate with low-rank metrics than...
  - Discriminative Components Analysis (DCA, Hoi et al. 2006)
  - Information Theoretic Metric Learning (ITML, Davis et al. 2007)
  - Large-Margin Nearest Neighbor (LMNN, Weinberger et al. 2006)
  - Local Fisher Discriminant Analysis (LFDA, Sugiyama 2007)
  - Maximally-Collapsing Metric Learning (MCML, Globerson et al. 2006)
  - Neighborhood Components Analysis (NCA, Globerson et al. 2005)

Low-rank Mahalanobis metric learning:

$\mathbf{A} \in \mathbb{R}^{n \times r}$ **s.t.** $\boldsymbol{r} << \boldsymbol{n}$

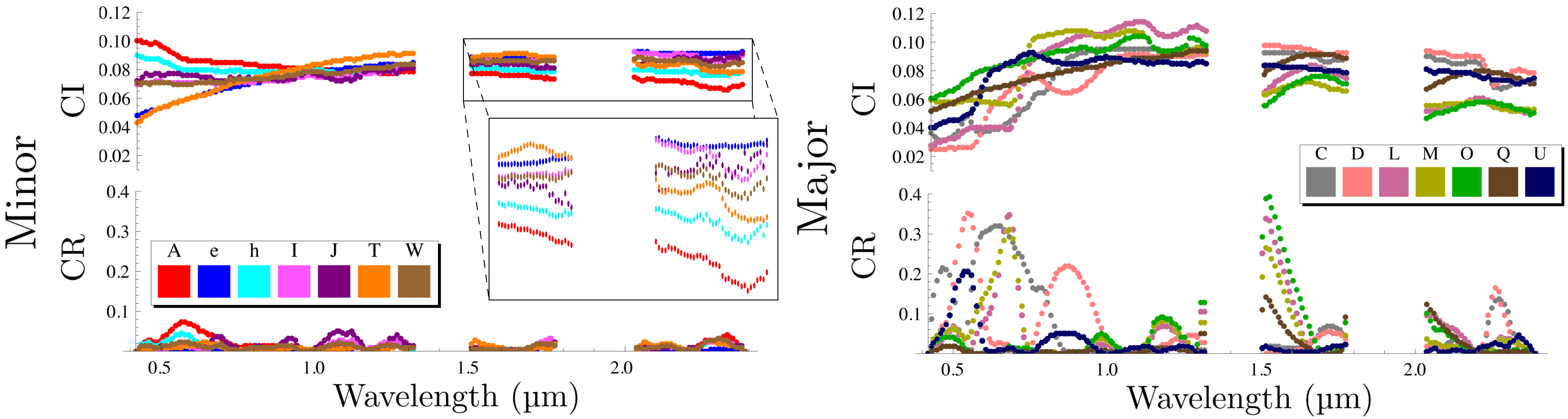


Applications:

- Dimensionality reduction
- Visualization (e.g., r=2 or r=3)
- Hybrid-LDA

Feature-weighted Metric Learning

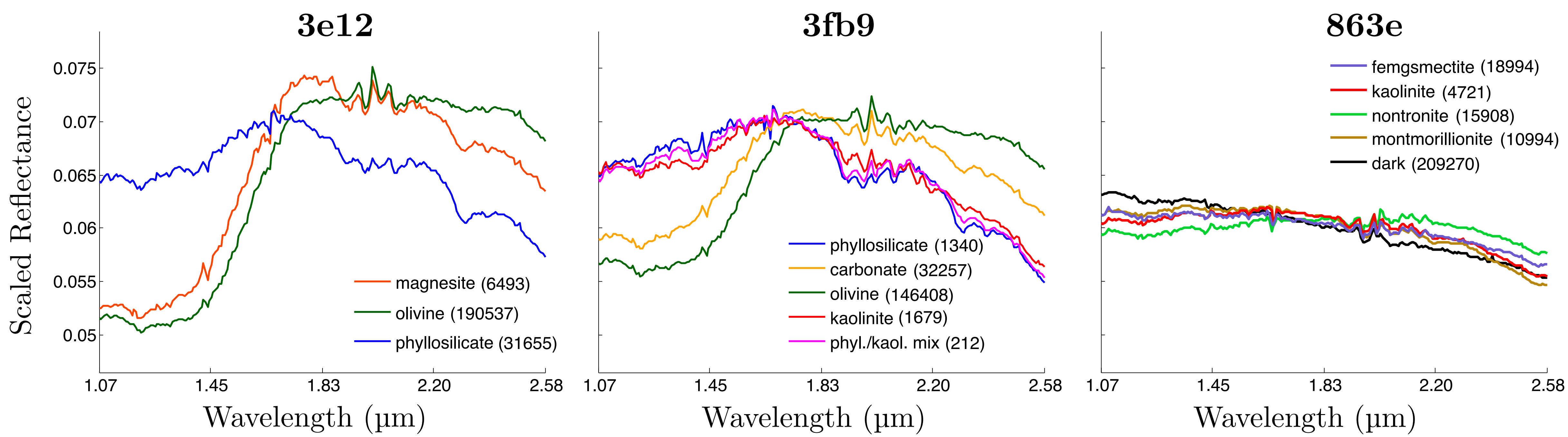
Ocean City Image Results ($n = 158$, $r = K$-1)



	EUC	LDA	LFDA	DCA	ITML	NCA	LMNN	MCML	Mean
Minor (mean)	94.6630	*95.7437*	94.9773	95.0654	95.1356	91.7682	95.6077	*95.7893*	94.8438
(std. dev.)	1.5033	0.8797	1.7054	0.9147	1.1968	2.0639	1.1710	1.3797	1.3518
Major (mean)	98.9563	99.0247	*99.3011*	98.7627	*99.3596*	97.8212	98.6416	99.0467	98.8642
(std. dev.)	0.6659	0.6980	0.1782	0.7308	0.4975	0.5267	0.6737	0.7047	0.5844
Combined (mean)	97.2308	*97.4018*	96.7522	94.6506	97.1351	94.9593	96.9959	*97.3196*	96.5557
(std. dev.)	0.2166	0.2945	0.4008	0.4513	0.5952	0.4230	0.3406	0.4147	0.3921
Overall (mean)	96.9500	*97.3901*	97.0102	96.1596	97.2101	94.8496	97.0817	*97.3852*	96.7546
(std. dev.)	0.7953	0.6241	0.7615	0.6989	0.7632	1.0045	0.7284	0.8330	0.7761

kNN (k=3) accuracy averaged over training sets of {25,50,100,150,250} samples / class for rank = K-1 Mahalanobis matrices (red=best, blue=second best)

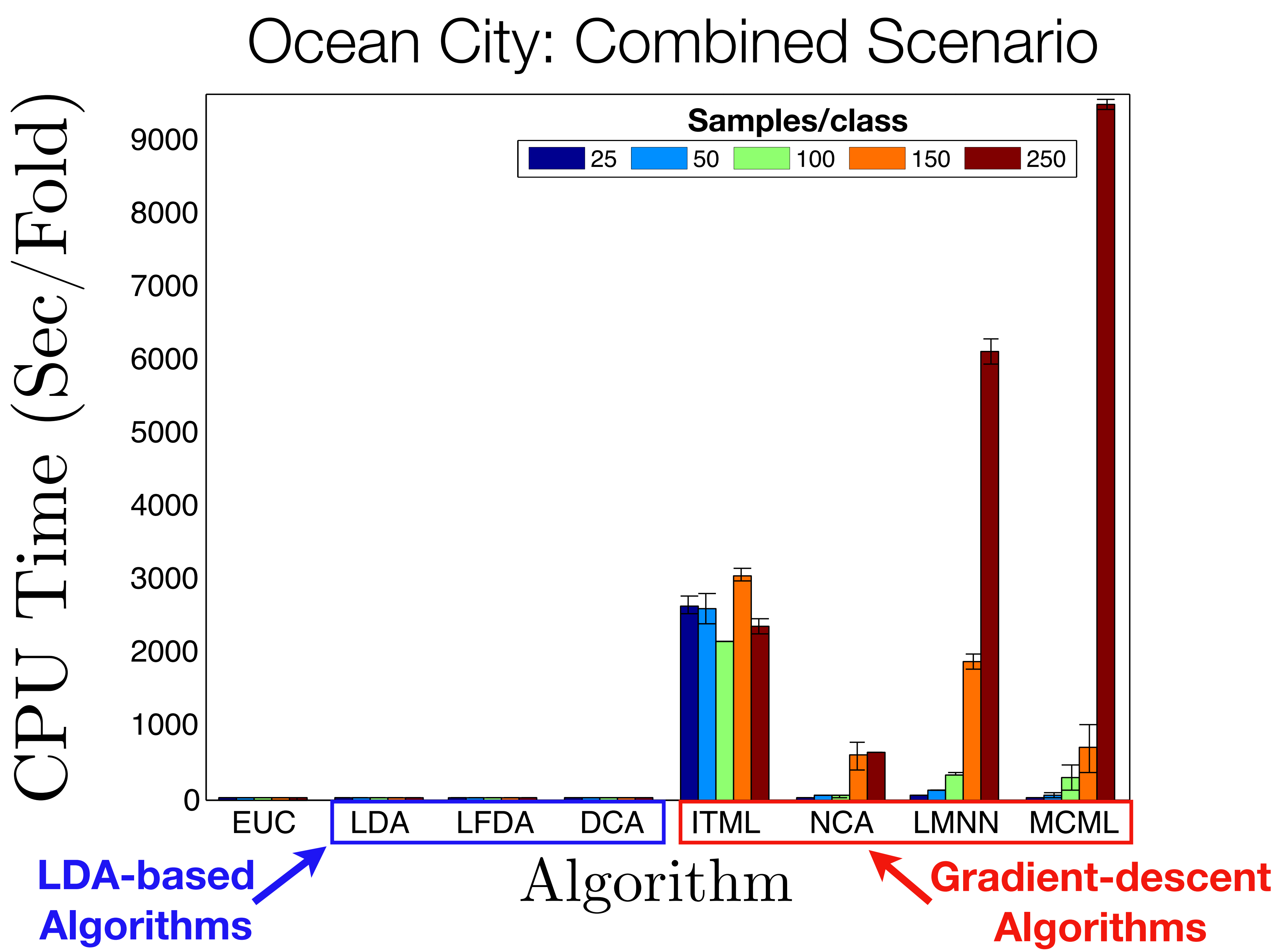
Feature-weighted Metric Learning
CRISM Imagery Results ($n = 231$, $r = K$-1)



	EUC	LDA	LFDA	DCA	ITML	NCA	LMNN	MCML	Mean
3e12 (mean)	98.3581	98.6337	98.4915	*99.0391*	98.5892	98.2425	98.6089	*98.8337*	98.5996
(std. dev.)	1.3917	1.6557	1.5803	1.1578	1.5929	1.2057	1.2634	1.3729	1.4026
3fb9 (mean)	88.1268	*92.0010*	88.6068	88.0427	*90.5458*	90.3243	90.2225	89.7439	89.7017
(std. dev.)	1.2670	1.6550	1.4933	0.5077	1.0319	2.0527	0.9556	1.3094	1.2841
863e (mean)	97.1800	*97.5700*	96.7900	96.1567	97.2100	*98.3667*	97.3000	97.3900	97.2454
(std. dev.)	1.0842	0.8344	1.4661	1.6358	1.2115	0.4997	1.1785	1.1361	1.1308
Overall (mean)	94.5550	*96.0682*	94.6294	94.4128	95.4483	*95.6445*	95.3771	95.3225	95.1822
(std. dev.)	1.2476	1.3817	1.5132	1.1004	1.2788	1.2527	1.1325	1.2728	1.2725

kNN (k=3) accuracy averaged over training sets of {25,50,100,150,250} samples / class for rank = K-1 Mahalanobis matrices (red=best, blue=second best)

Feature-weighted Metric Learning Computation Time / Fold



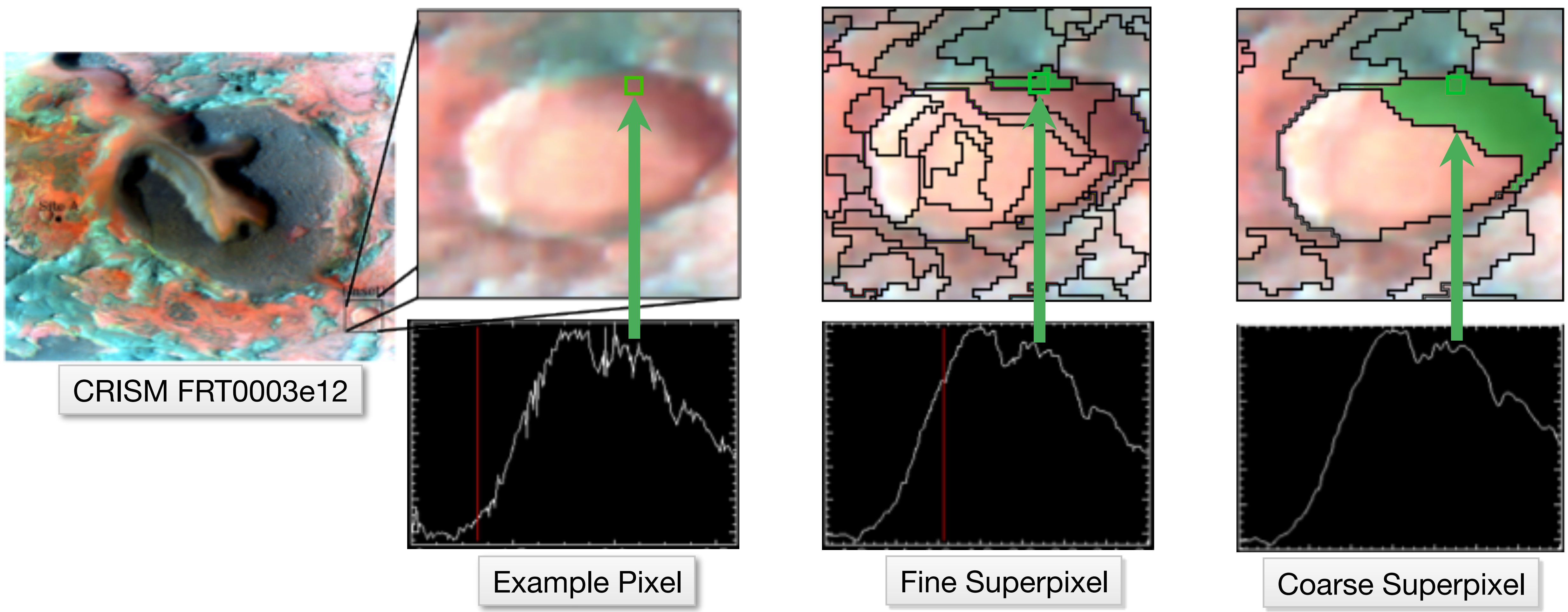
Feature-weighted Metric Learning
Application: Superpixel Segmentation

[Bue and Thompson, WHISPERS 2010]

Goal: segment image into groups of spatially contiguous spectra (*superpixels*) representing similar materials (via the Felzenszwalb segmentation algorithm)

[Felzenszwalb 2004]

- Reduces processing time of subsequent analyses
- Noise reduction of order $n^{1/2}$ for an n-pixel superpixel

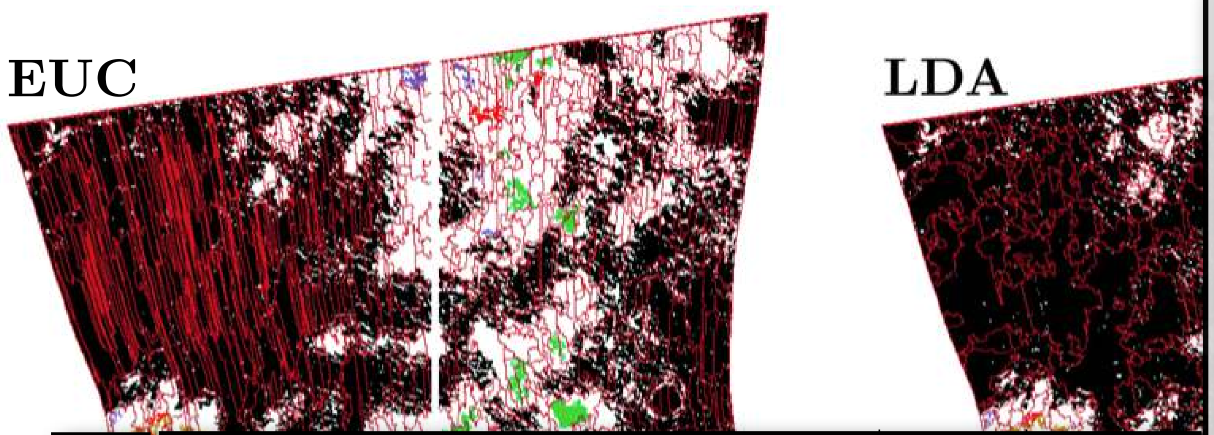


More accurate similarity measure => more pure* superpixels

*pure = constituent spectra represent the same material(s)

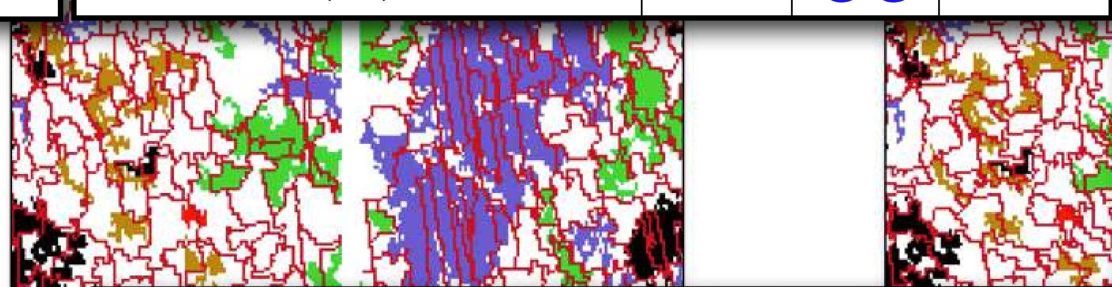
Segmentation Results

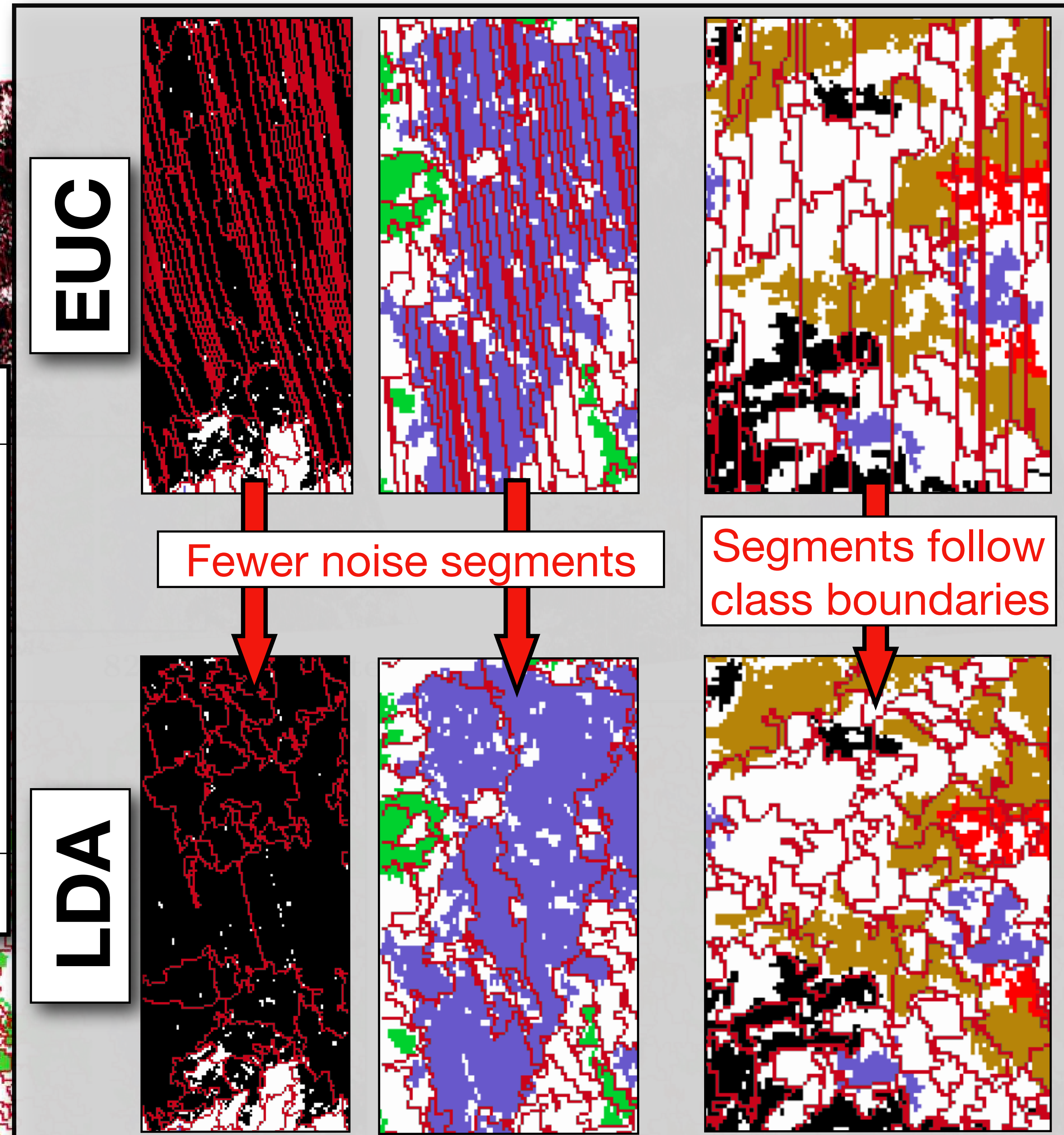
CRISM Image 863e: Euclidean vs. LDA vs. ITML



Superpixel Purity

Class (# pixels)	EUC	LDA	ITML
FeMg Smectite (6443)	26	49	48
Kaolinite (4051)	98	99	99
Montmorillonite (10901)	11	31	17
Nontronite (4753)	37	52	40
Neutral Region (115225)	97	99	98
Average (%)	53	**66**	60





red lines=segment boundaries | colored regions=class ROIs | white regions=unlabeled pixels

Inter-domain Material Identification

Source and target spectra captured under similar (but not identical) conditions ⇒ $\mathcal{D}^S \approx \mathcal{D}^T$

good source classifier ⇏ good target predictions

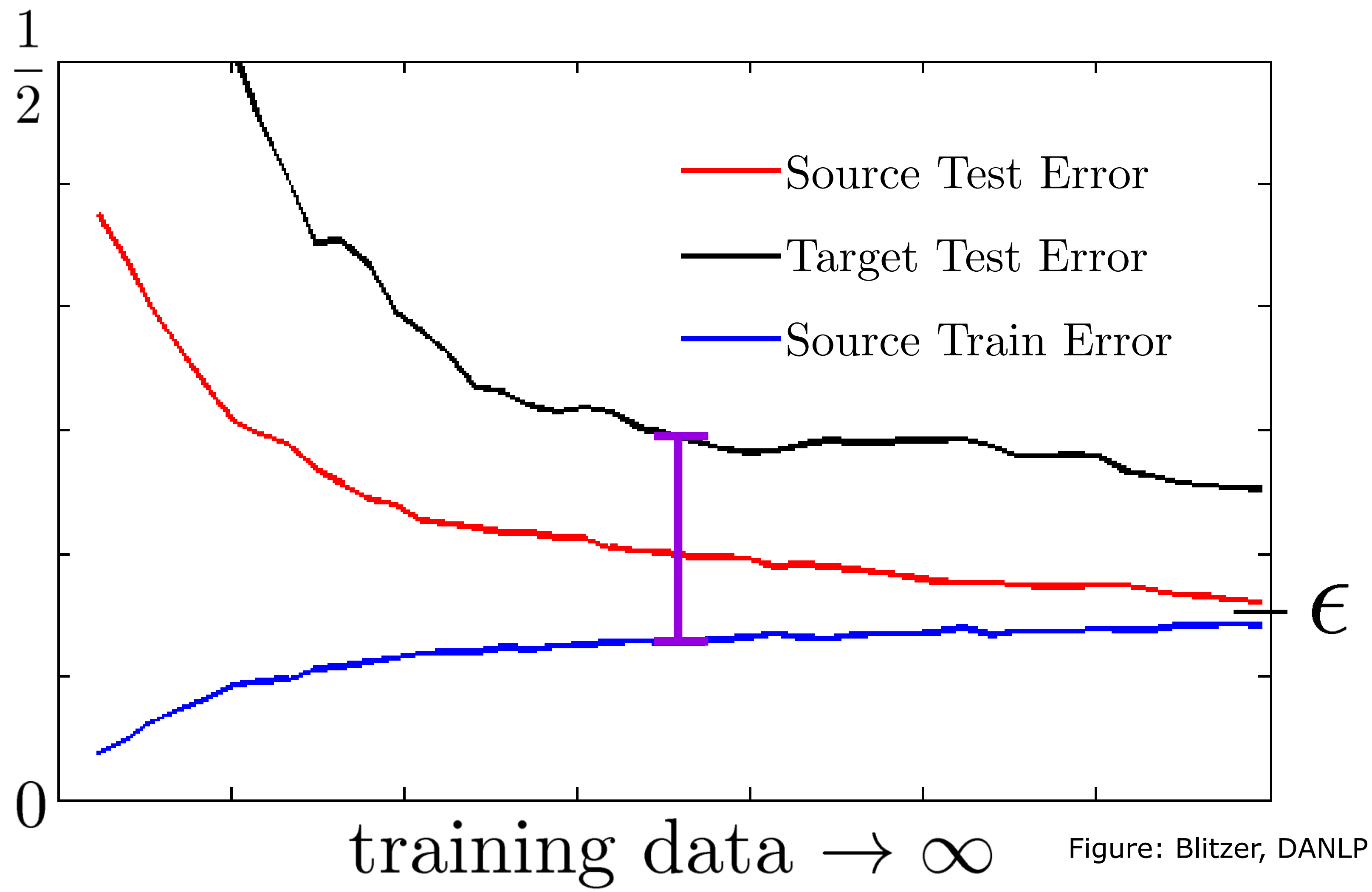
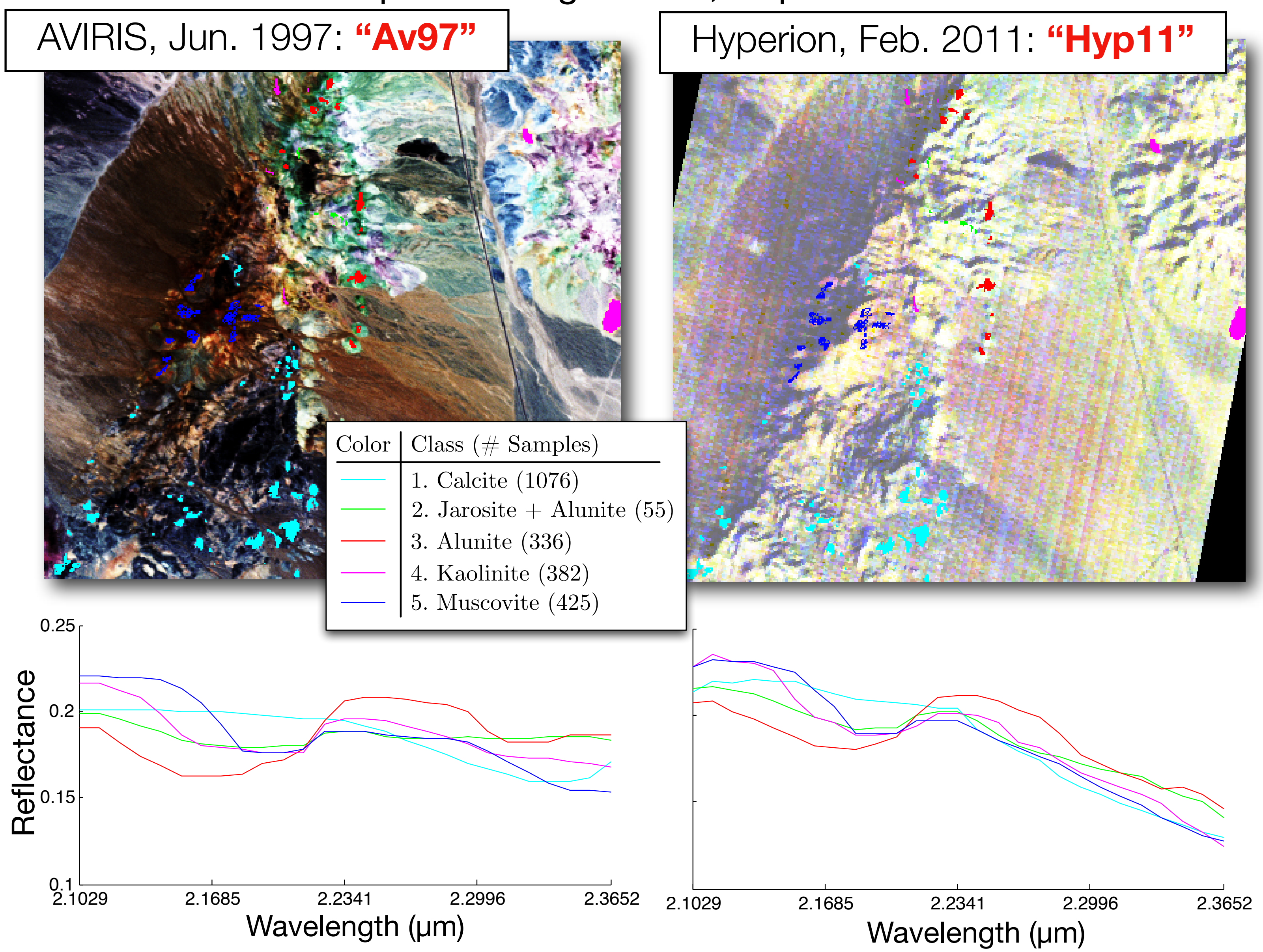


Figure: Blitzer, DANLP '10

Inter-domain Material Identification

Do not always have representative samples to classify target materials
⇒ can we train a classifier using samples from similar imagery?

Cuprite Mining District, Cuprite NV



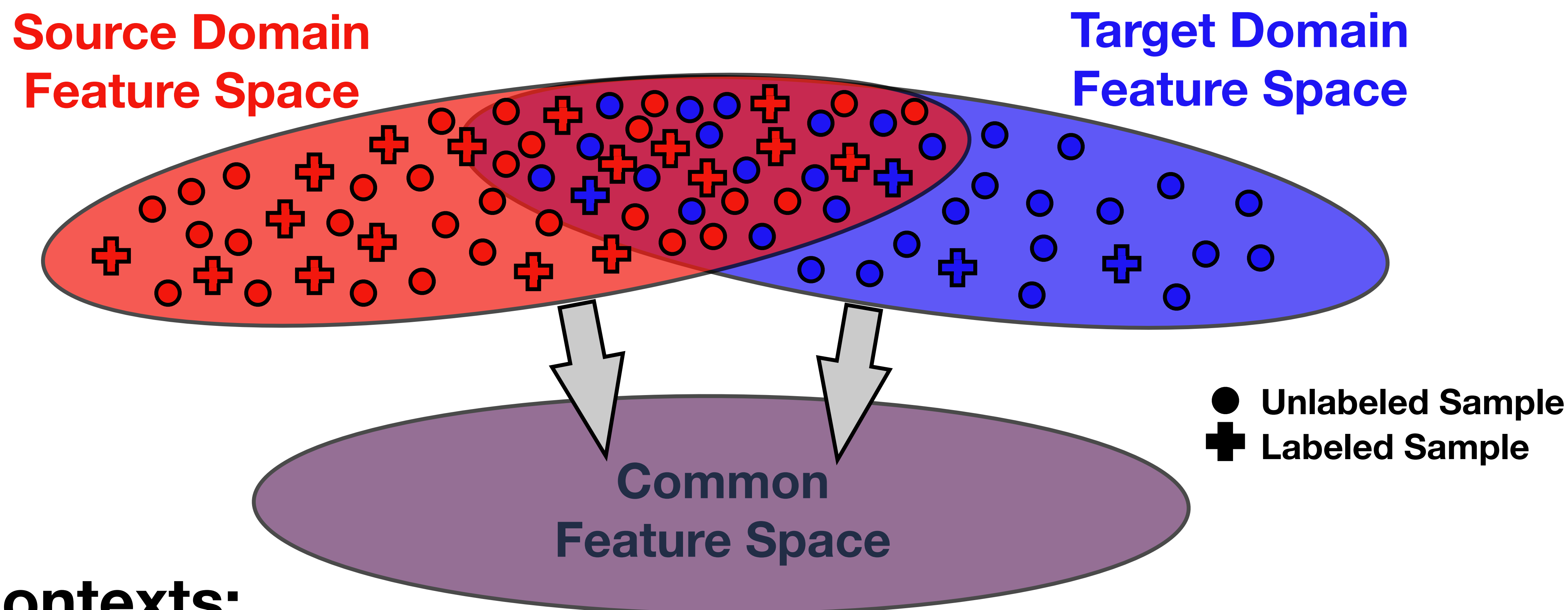
Spectra reflect their capture conditions

- **atmosphere**
- **seasonal effects**
- **sensor resolution (spatial + spectral)**
- **viewing geometry**
- **...**

⇒ Classifier trained with Av97 spectra will not generalize to Hyp11 spectra (and vice-versa)

Domain Adaptation Problems
Supervised vs. Unsupervised

Domain Adaptation: improve source to target generalization by reconciling domain-specific differences



Two contexts:

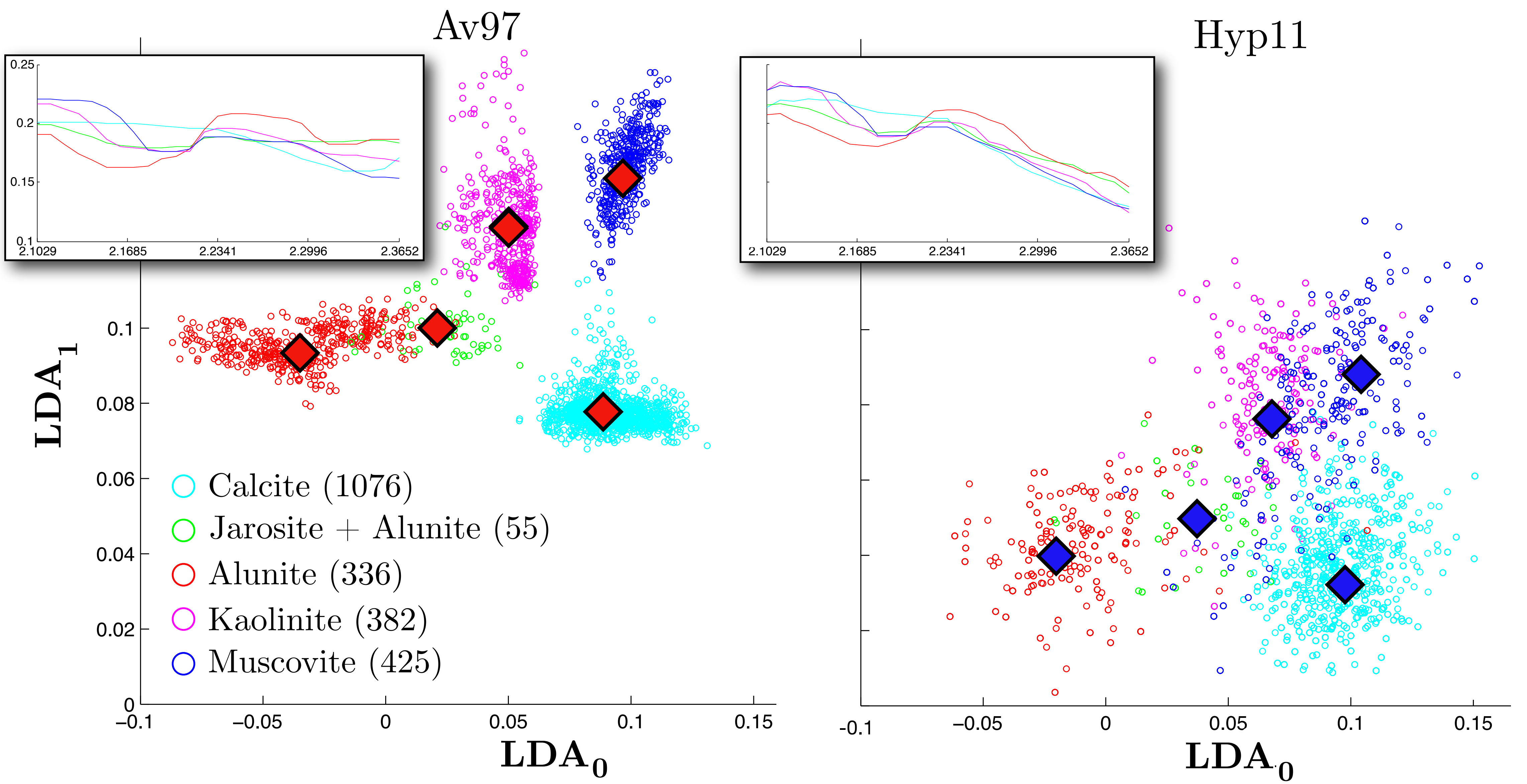
I. **Supervised:** small amount* of labeled target data available

II. **Unsupervised:** no labeled target data available

Related topics: *multitask/transfer learning*, *manifold alignment*

*small amount=insufficient to train a target domain classifier

Domain Adaptation
Separability and Interclass Distances

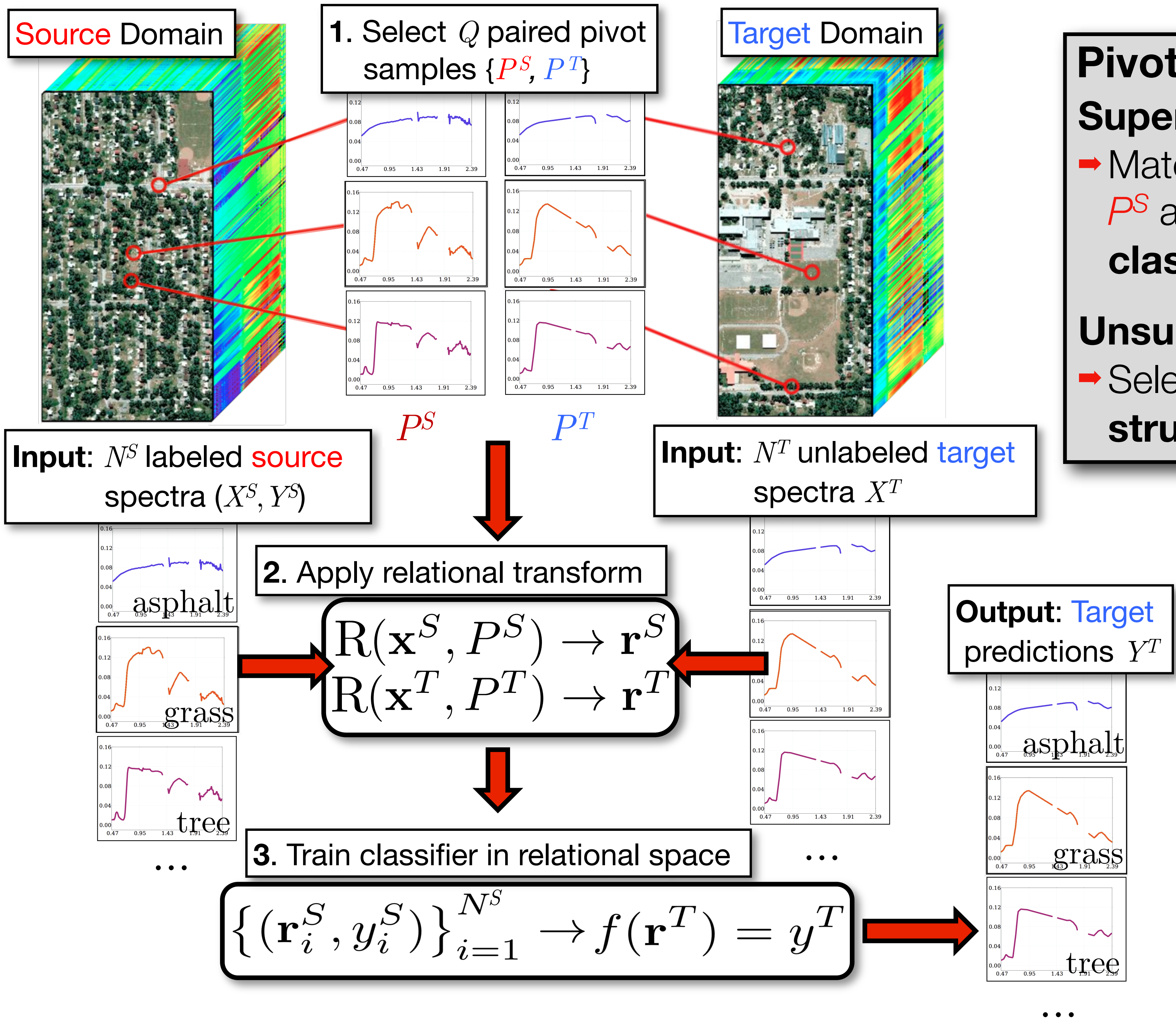


Above: class means and first two dimensions of LDA-projected Av97, Hyp11 samples

- Classes better separated in Av97 image than in Hyp11 image
- Relative distances between classes similar in each images

Multiclass Domain Adaptation
[Bue et al., WHISPERS 2010, 2011]

Relational Class Knowledge Transfer (RelTrans)



Pivot Selection

Supervised:

- Match labeled samples to define P^S and P^T (e.g., **samples near class means**)

Unsupervised:

- Select P^T that reflect **inter-class structure** of P^S

Relational ("R-space") Transformation

For sample $\mathbf{x}^D$ and pivot set P^D from domain $D \in \{S, T\}$, map $\mathbf{x}^D$ from $\mathcal{R}^n \rightarrow \mathcal{R}^Q$ according to

$$R(\mathbf{x}^D, P^D) = \left(\frac{d(\mathbf{x}^D, \mathbf{p}_1^D)}{\sum_{\ell=1}^{Q} d(\mathbf{x}^D, \mathbf{p}_\ell^D)}, \ldots, \frac{d(\mathbf{x}^D, \mathbf{p}_Q^D)}{\sum_{\ell=1}^{Q} d(\mathbf{x}^D, \mathbf{p}_\ell^D)} \right)$$

where

$$d(\mathbf{x}^D, \mathbf{p}^D) = ||\mathbf{x}^D - \mathbf{p}^D||_2{}^{*}$$

$$Q = \sum_{k=1}^{K} Q_k$$

$$Q_k = \# \text{ of pivots for class } k$$

➡ l^{th} entry of $R(\mathbf{x}^D, \mathbf{p}^D)$ = likelihood of distinguishing $\mathbf{x}^D$ and $\mathbf{p}_l^D$ from other pivots P^D

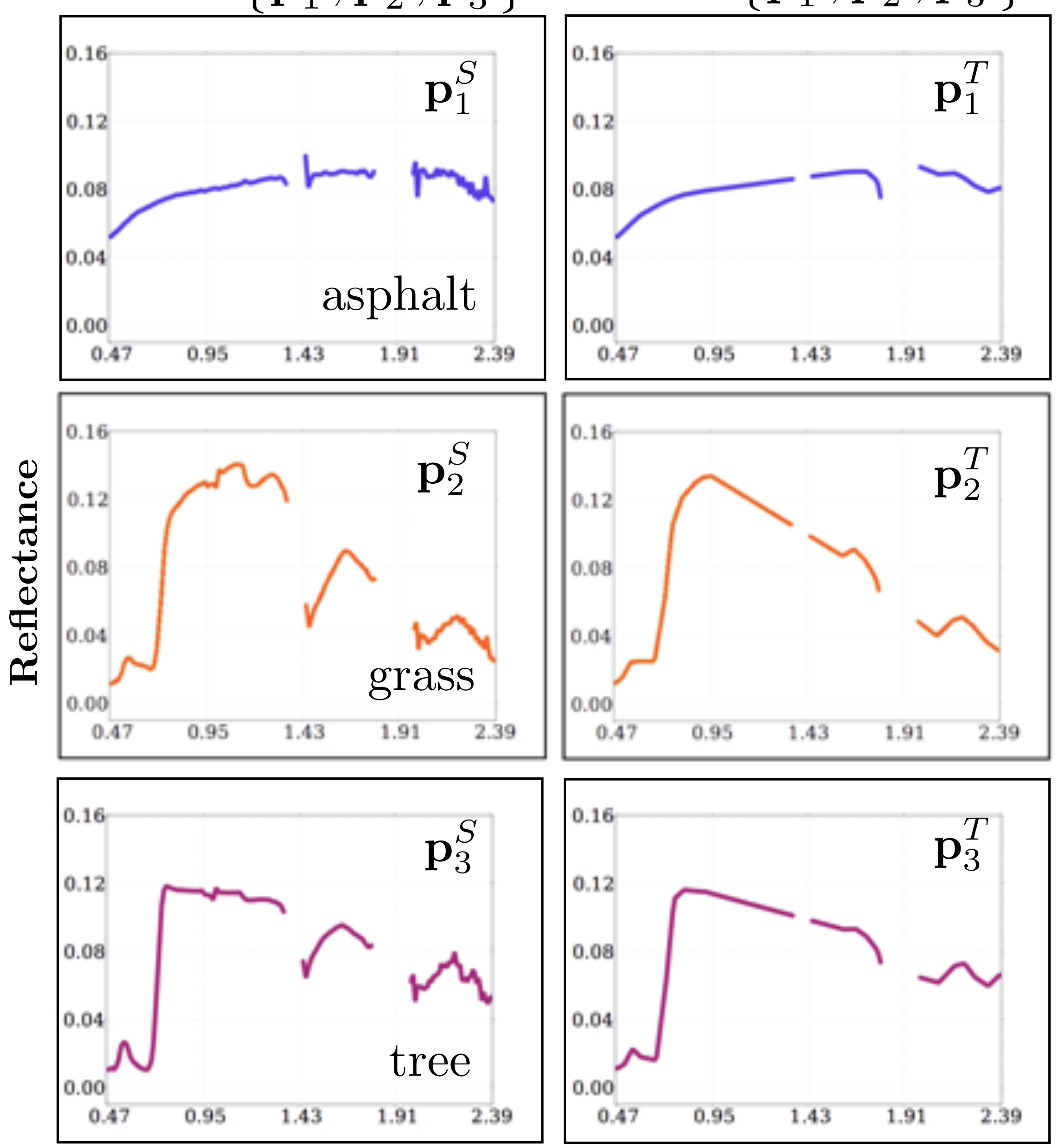
* $\mathbf{x}^D = \mathbf{x}^D / ||\mathbf{x}^D||_2 \Rightarrow d(\mathbf{x}^D, \mathbf{p}^D) \Rightarrow$ angle between $\mathbf{x}^D$ and $\mathbf{p}^D$

Relational ("R-space") Transformation

Example $(K{=}3,\ Q_k{=}1)$

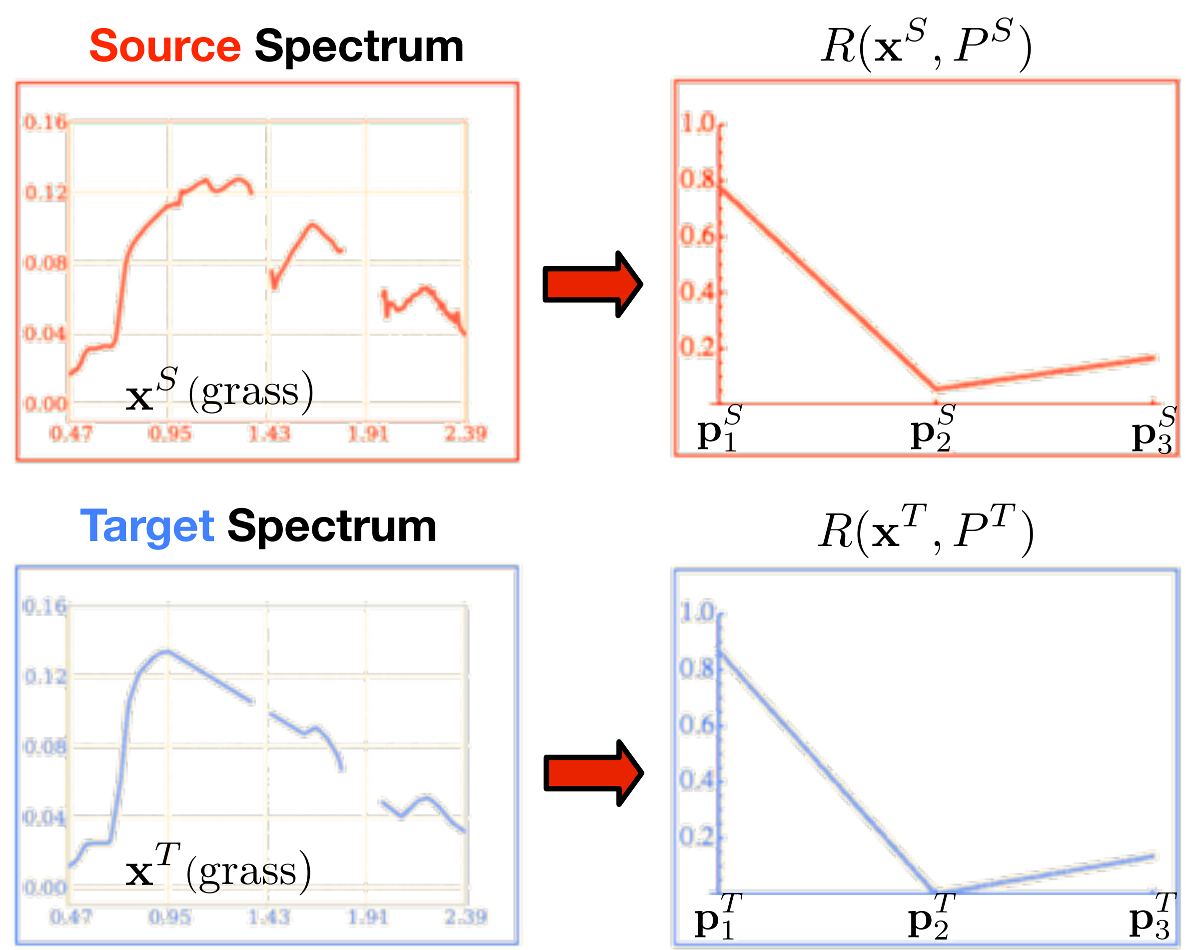
Source Pivots $\qquad$ **Target Pivots**

$$P^S = \{\mathbf{p}_1^S, \mathbf{p}_2^S, \mathbf{p}_3^S\} \qquad P^T = \{\mathbf{p}_1^T, \mathbf{p}_2^T, \mathbf{p}_3^T\}$$



R-Transform

$$R(\mathbf{x}^D, P^D) = \left(\frac{d(\mathbf{x}^D, \mathbf{p}_1^D)}{\sum_{\ell=1}^{Q} d(\mathbf{x}^D, \mathbf{p}_\ell^D)}, \ldots, \frac{d(\mathbf{x}^D, \mathbf{p}_Q^D)}{\sum_{\ell=1}^{Q} d(\mathbf{x}^D, \mathbf{p}_\ell^D)} \right)$$

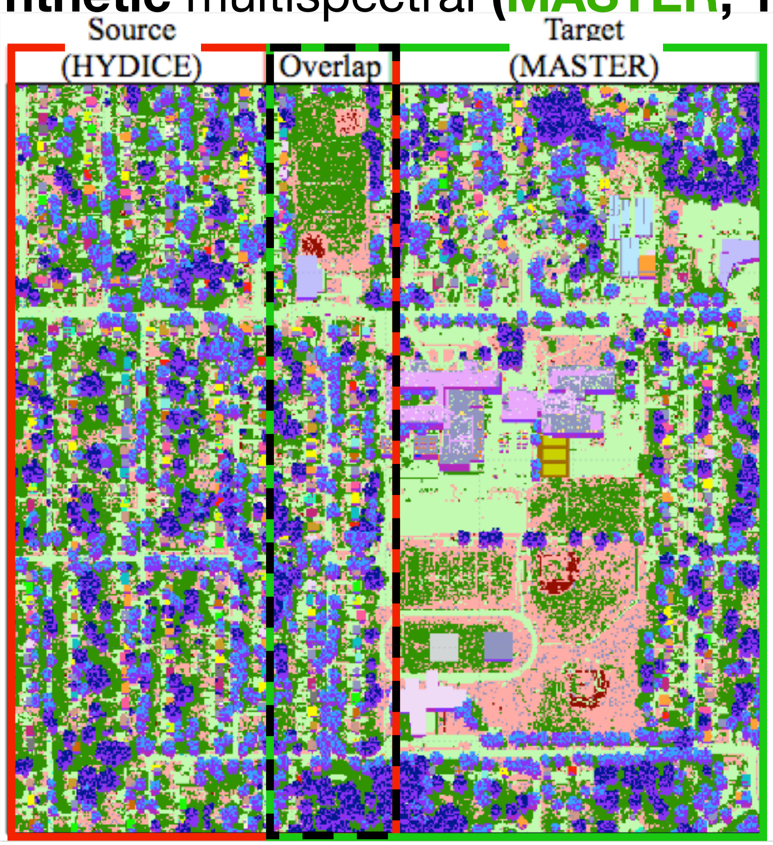


Domain-specific Feature Space ➡ **Common Feature Space**

30

Supervised Domain Adaptation
MinDist vs. MinDist in the R-space

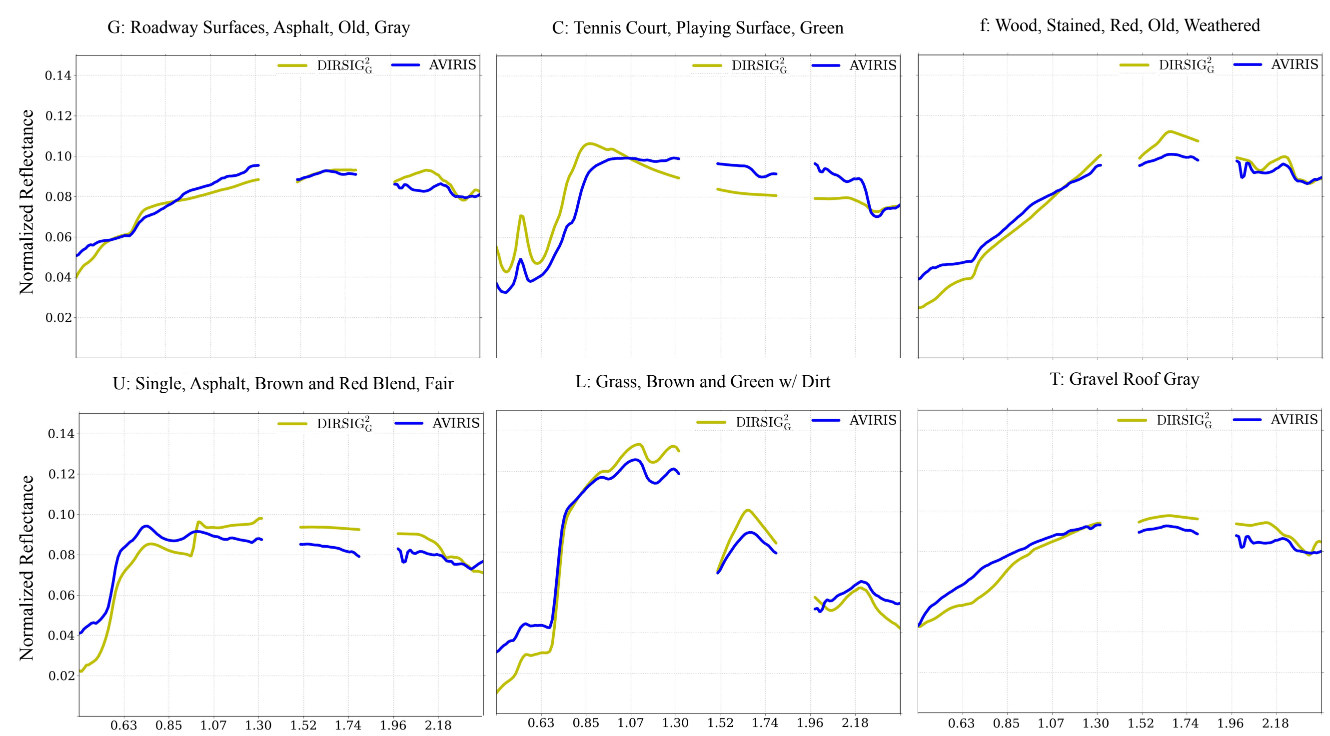
Synthetic hyperspectral (**HYDICE, 210 bands) => Synthetic** multispectral (**MASTER, 12 bands)**



Above: synthetic DIRSIG HYDICE and MASTER Images

	MinDist Source=>Target	R-space MinDist Source=>Target
$K^S = K^T$	86%	99%
$K^S < K^T$ (+outlier detection)	58 (86)%	66 (97)%

Synthetic hyperspectral (**HYDICE, 210 bands) =>**
Real hyperspectral (**AVIRIS, 224 bands)**



Above: class means of pivots from DIRSIG (P^S) and Ocean City AVIRIS (P^T) images

	MinDist Source=>Target	R-space MinDist Source=>Target
$K^S = K^T$	45%	72%
$K^S < K^T$ (+outlier detection)	26 (31)%	43 (74)%

Supervised Domain Adaptation
Radiance to Reflectance Classification

Remotely-sensed spectra contaminated by atmospheric particles

Atmospheric Calibration: converts contaminated at-sensor radiance ($\mathbf{RAD}$) measurements to surface reflectance ($\mathbf{RFL}$)

Two Approaches:

I. Atmospheric modeling (i.e., radiative transfer)
- Problem: *computationally expensive*

II. Linear approximation (ELM): regression using field-measured reflectance spectra

$$\mathbf{RAD_b(\lambda) \approx A_b RFL_b(\lambda) + B_b}$$

$\mathbf{b}$: band

$\boldsymbol{\lambda}$: wavelength

$\mathbf{A_b}$: gain at band $\mathbf{b}$

$\mathbf{B_b}$: offset at band $\mathbf{b}$

- Simple model, often yields comparable/ better results than radiative transfer methods
- Problem: *field spectra may not be available*

Empirical Line Method (ELM)

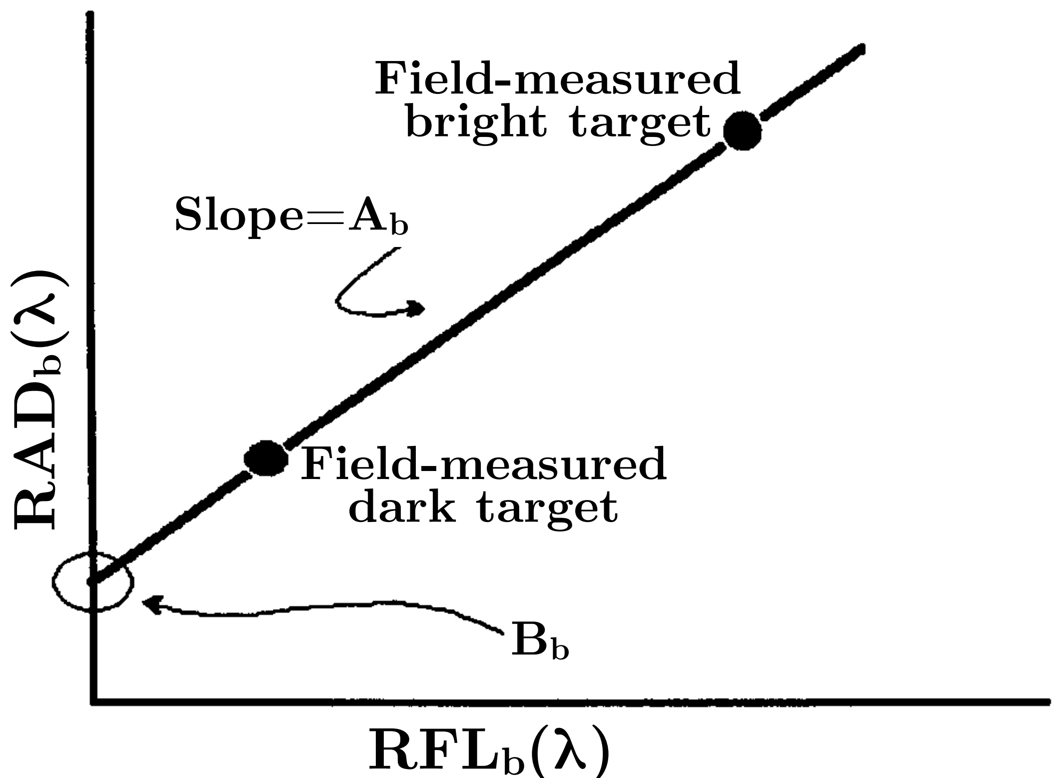


Figure modified after: [Farrand et al. 1994]

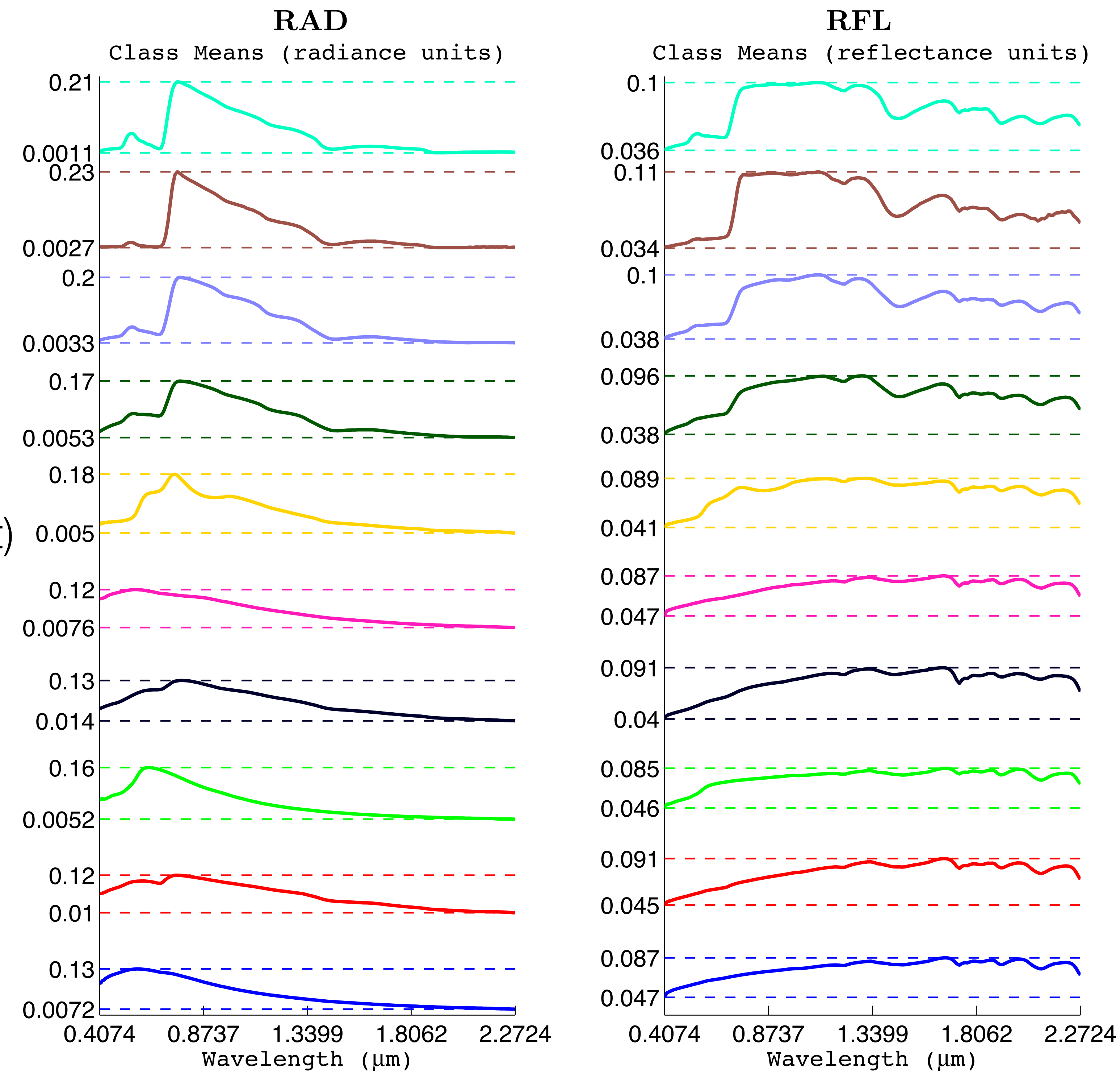
Supervised Domain Adaptation
Radiance to Reflectance Classification

Goal: classify reflectance spectra using radiance spectra as training data (**RAD2RFL**) and vice-versa (**RFL2RAD**)

- Evaluate R-space accuracy with different classifiers
- Compare to multi-task learning techniques

Data: Synthetic HYDICE spectra, 10 classes, 200 samples/class (right)

Tree, Silver Maple, Leaf
Tree, Black Oak, Leaf
Grass, Green, Healthy
Grass, Brown and Green with Dirt
Tennis Court, Playing Surface, Red
Rooftop, Gravel, Gray
Siding, Brick, Mix Brown, Fair
Shingle, Asphalt, Mix Brown, Good
Wood, Stained, Red, Old, Weathered
Asphalt, Black, New



Intra-domain accuracies:

- **RAD** = 100.0%, **RFL** = 99.4%

Supervised Domain Adaptation
R-space Classification with Different Classifiers

		RAD2RFL Mean	RAD2RFL Std	RFL2RAD Mean	RFL2RAD Std	Overall Mean	Overall Std
Mindist	Orig	0.1123	0.0087	0.1123	0.0008	0.1123	0.0048
	RT	*0.8904*	0.0058	0.8634	0.0139	0.8769	0.0099
GLVQ	Orig	0.1123	0.0000	0.1123	0.0000	0.1123	0.0000
	RT	0.8798	0.0174	0.8575	0.0296	0.8687	0.0235
GRLVQ	Orig	0.1123	0.0000	0.1123	0.0000	0.1123	0.0000
	RT	0.8788	0.0161	0.8467	0.0132	0.8628	0.0146
SVM-lin	Orig	0.2024	0.0076	0.1123	0.0000	0.1574	0.0038
	RT	0.8742	0.0040	*0.8806*	0.0047	*0.8774*	0.0044
SVM-rbf	Orig	0.2063	0.0011	0.1123	0.0000	0.1409	0.0266
	RT	*0.8823*	0.0099	*0.8894*	0.0194	*0.8858*	0.0147

Accuracy in original feature space vs. average RelTrans accuracy for $Q_k \in \{5,10,20,50,100\}$ pivots / class (red=best, blue=second best)

Mindist: Minimum distance to class means
GLVQ: Generalized Learning Vector Quantization
GRLVQ: Generalized Relevance Learning Vector Quantization
SVM-lin: Support Vector Machine (linear kernel)
SVM-rbf: Support Vector Machine (radial basis function kernel)

- Original feature space: all samples classified into one (0.1123) or two (0.2024) class(es)
- All classifiers above give good performance in R-space

Domain Adaptation
Related Work: Multitask Learning

Per-task (*task=domain*) training data **Jointly-learned, per-task linear models**

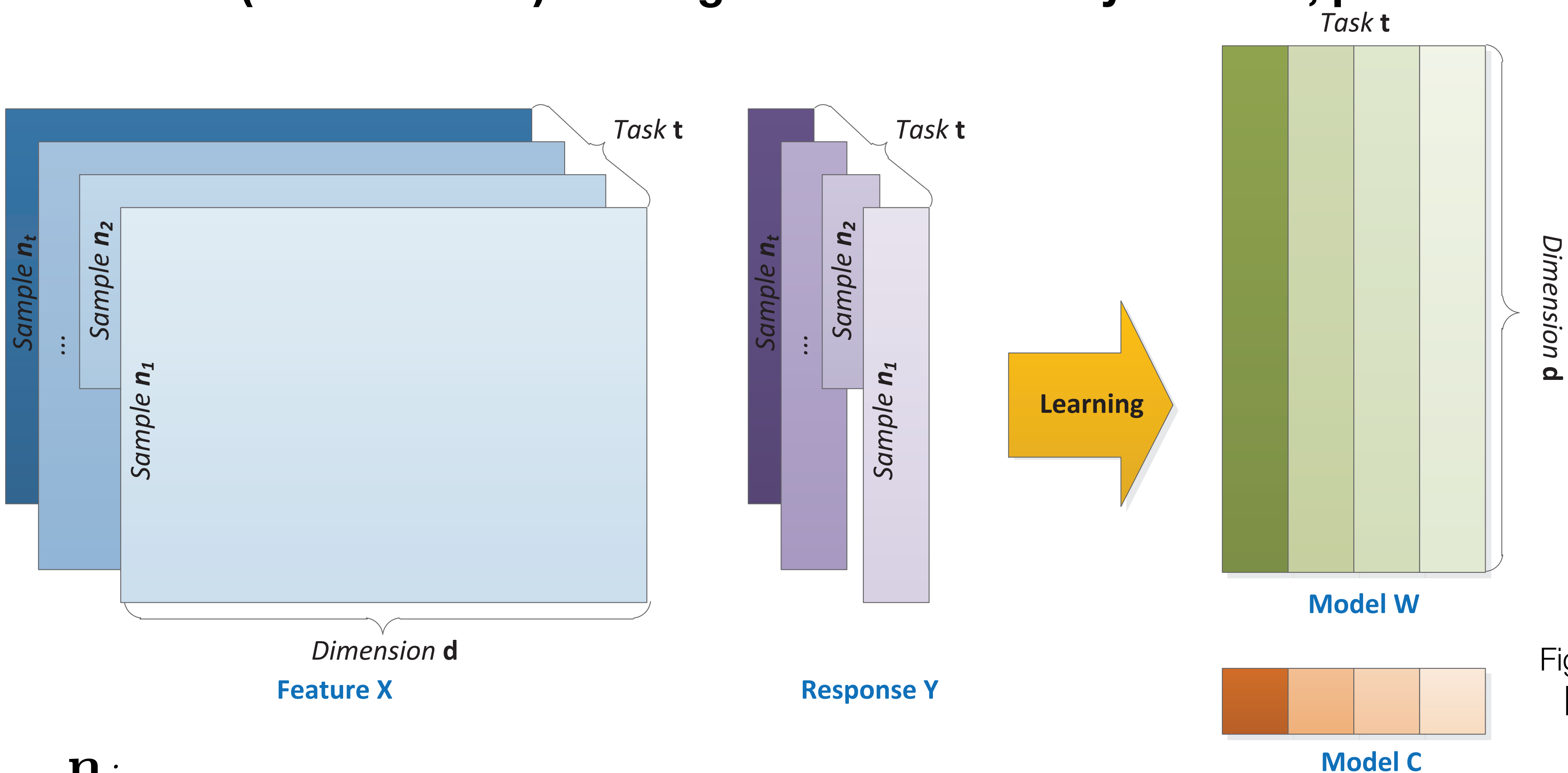


Figure modified after: [Zhou et al. 2012]

$$\min_{\mathbf{W},\mathbf{c}} \sum_{i=1}^{\mathbf{t}} \sum_{j=1}^{\mathbf{n}_i} \underbrace{\log\left(1+\exp\left(\boxed{-Y_{i,j}}\left(\langle \mathbf{W}_{i,\cdot}, X_{i,j}\rangle + \mathbf{c}_i\right)\right)\right)}_{\text{Logistic Loss}} + \underbrace{\Omega(\mathbf{W},\gamma)}_{\text{Regularizer}}$$

Problem: **Two-class assumption!** $Y_{i,j} \in \{-1,1\}$

Approach: Learn $K(K\text{-}1)/2$ binary multitask classifiers, predict via majority vote

Supervised Domain Adaptation
Radiance to Reflectance Multitask Learning Results

MTL Method	$\Omega(\mathbf{W},\gamma)$
JFS [Obozinski et al., 2009]	$\gamma_1\|\mathbf{W}\|_1 + \gamma_2\|\mathbf{W}\|_2^2$
MTFL [Argyriou et al., 2007]	$\gamma_1\|\mathbf{W}\|_{1,2} + \gamma_2\|\mathbf{W}\|_2^2$
Trace [Ji and Ye, 2009]	$\gamma_1\|\mathbf{W}\|_*$

Source task: N^S labeled source domain samples

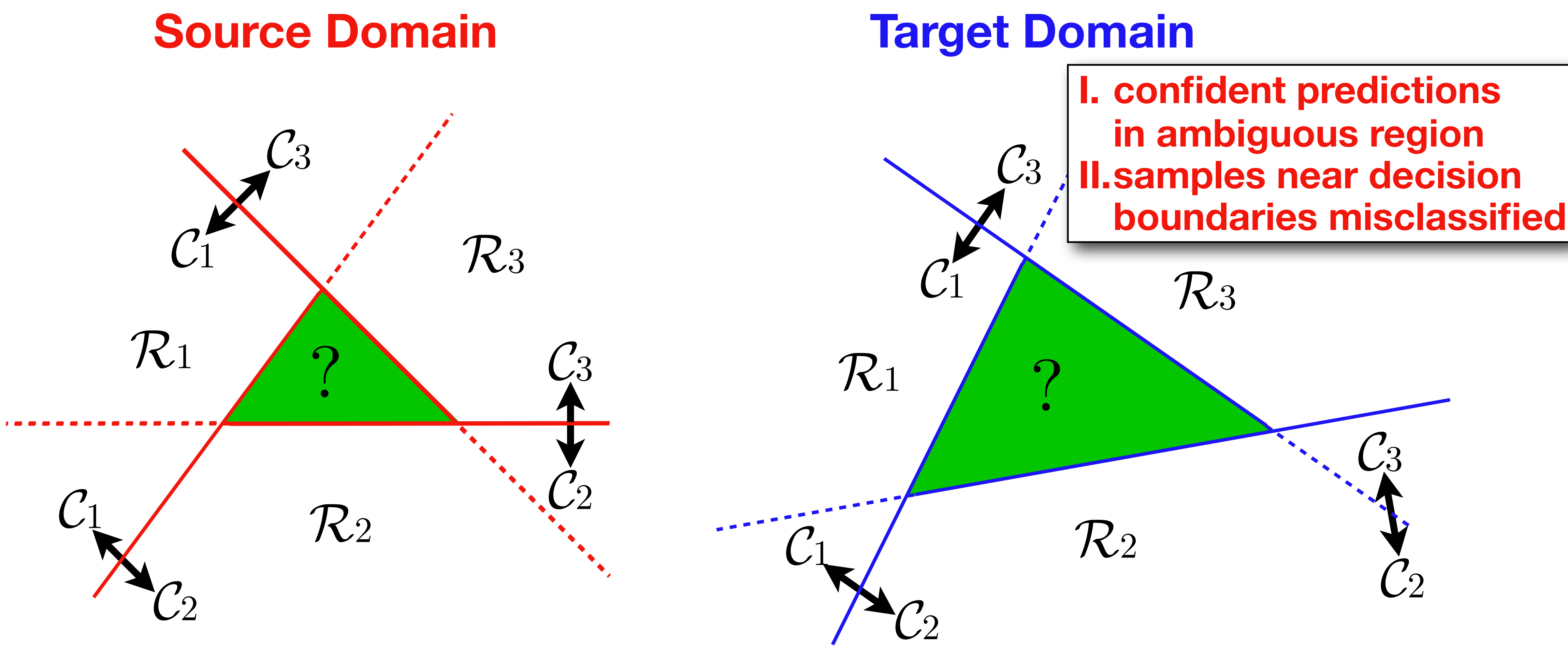
Target task: Q labeled target pivot samples

		RAD2RFL		RFL2RAD		Overall	
		Mean	Std	Mean	Std	Mean	Std
MTL	JFS	0.8444	0.0521	0.7953	0.0103	0.8198	0.0312
MTL	MTFL	0.8785	0.0126	0.8015	0.0058	0.8400	0.0092
MTL	Trace	*0.8921*	0.0015	0.8004	0.0108	0.8463	0.0062
RT	SVM-lin	0.8742	0.0040	*0.8806*	0.0047	*0.8774*	0.0044
RT	SVM-rbf	*0.8823*	0.0099	*0.8894*	0.0194	*0.8858*	0.0147

Average accuracy using MTL techniques vs. RelTrans with $Q_k \in \{5,10,20,50,100\}$ labeled target samples (pivots) / class (red=best, blue=second best)

Binary decomposition ignores multiclass structure of the learning problem

Domain Adaptation
Issues with Multiclass to Binary Decomposition



Figures modified after: [Bishop 06]

Issue: separability of classes in source vs. target domain*

I. Target classes *more* separated than source classes ⇒ D.A. easier

II. Target classes *less* separated than source classes ⇒ D.A. harder

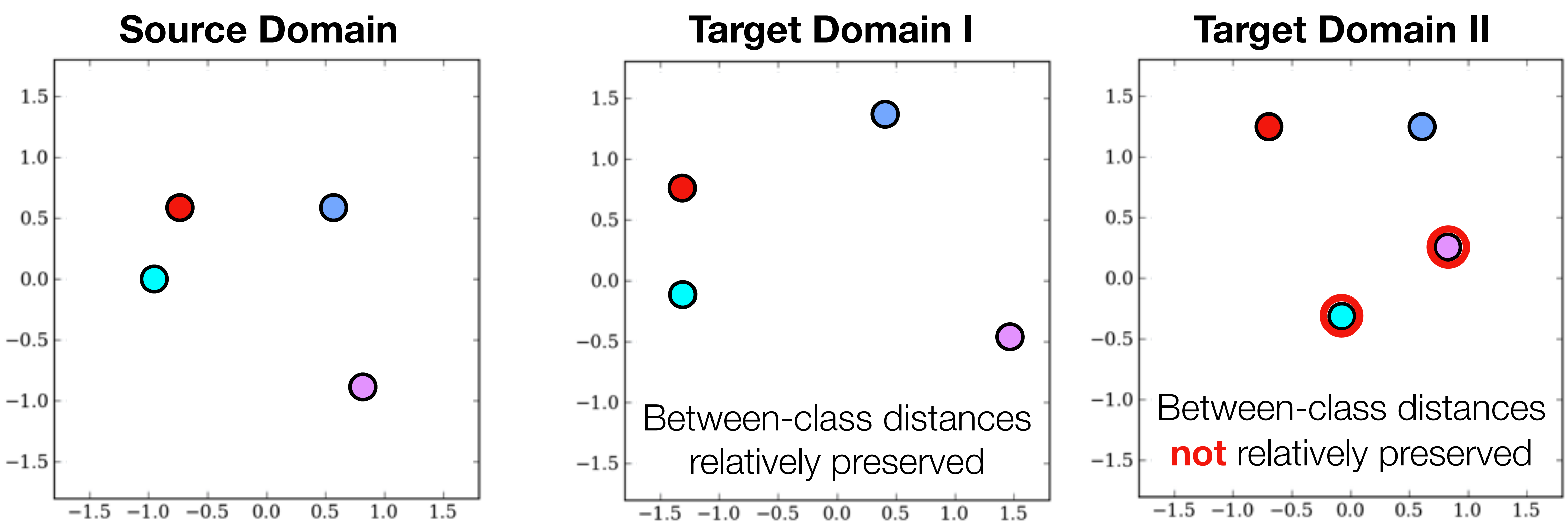
*** Assuming source/target domains similar**

Unsupervised Domain Adaptation

[Bue and Thompson, NIPS Domain Adaptation workshop 2011]

Multiclass Continuous Correspondence Learning (MCCL)

Assume between-class distances relatively preserved across domains

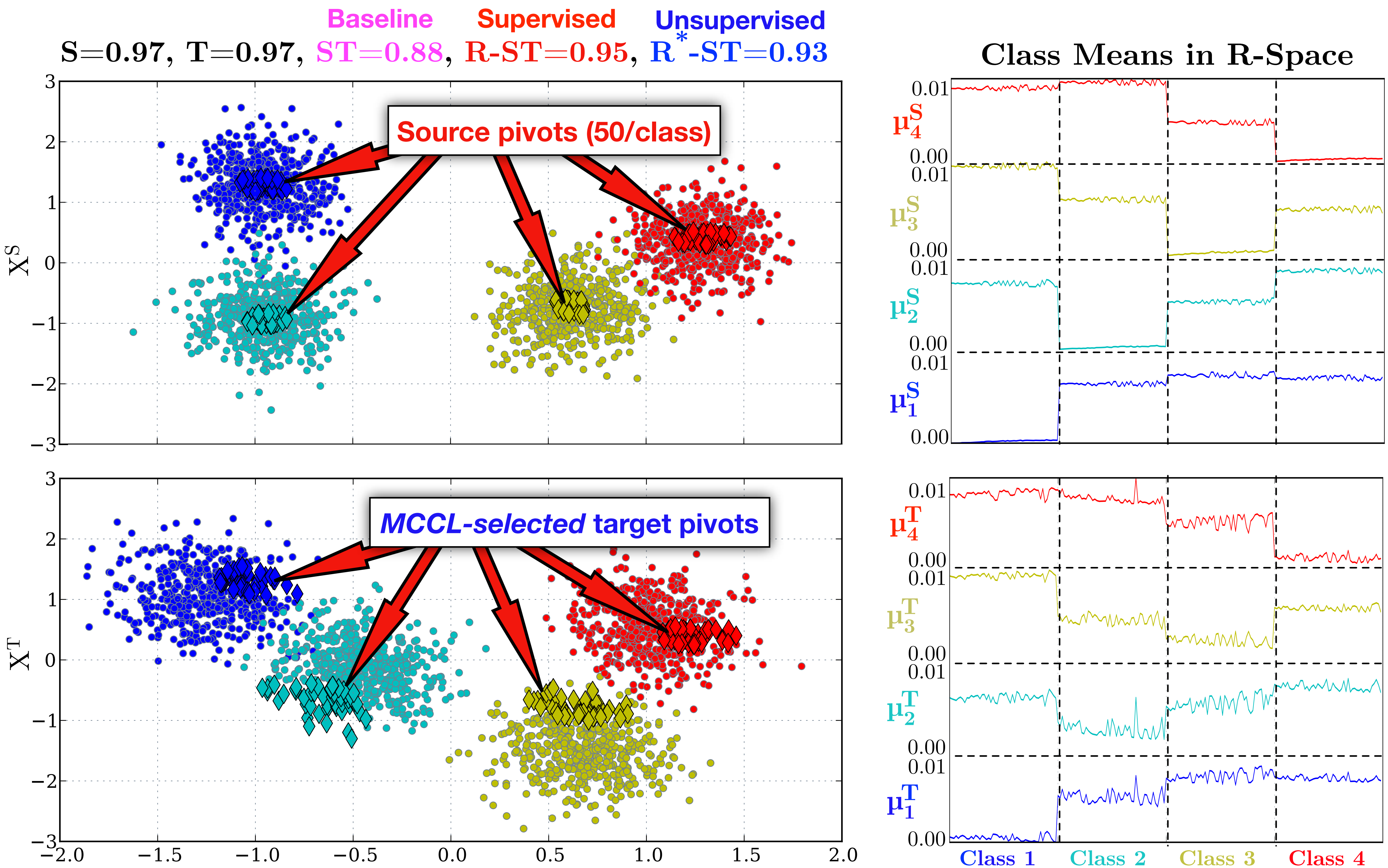


Unsupervised Pivot Selection with MCCL:

For each source pivot $\mathbf{p}_i^S$ in P^S, select target pivot $\mathbf{p}_i^T = \mathbf{x}_\ell^T$ s.t.

$$\ell = \operatorname*{argmin}_j \|R(\mathbf{p}_i^S, P^S) - R(\mathbf{x}_j^T, P^S)\|, \; j \in \{1, \ldots, N^T\}$$

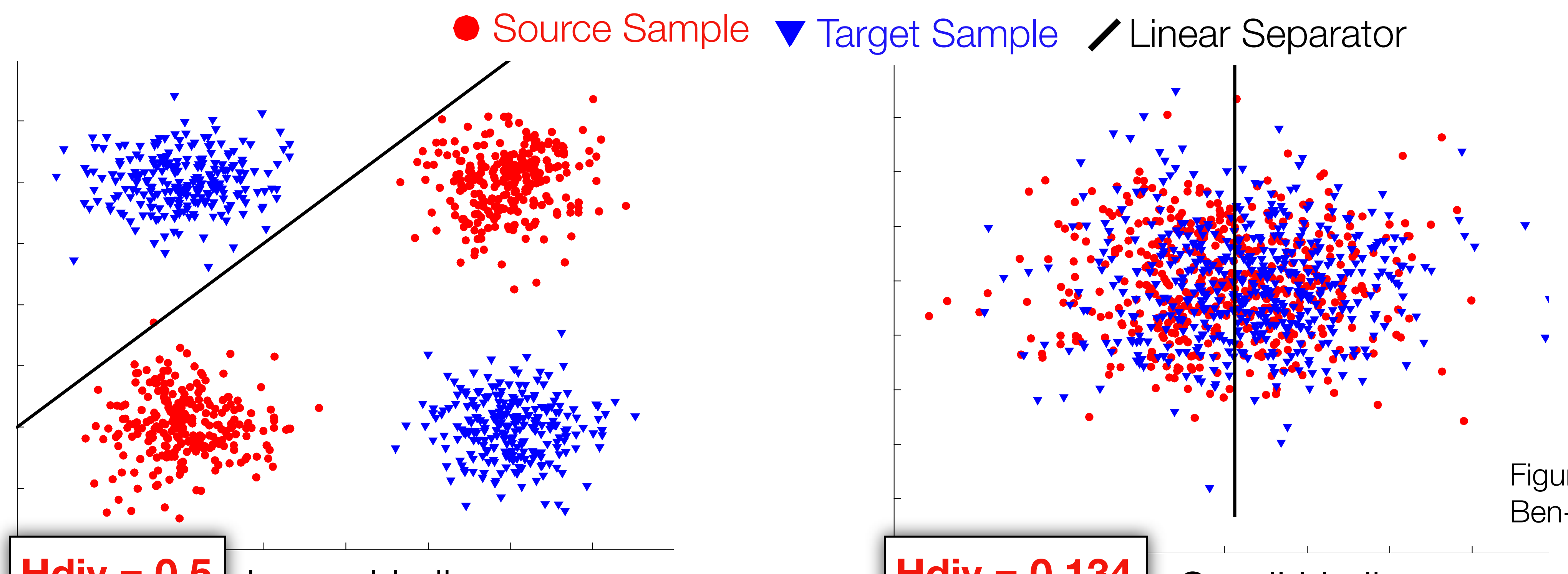
Unsupervised Domain Adaptation
Synthetic Example: Transformed Gaussians



Unsupervised Domain Adaptation
Measuring Pivot Set Quality with H-divergence

[Ben-David et al. 2010]

H-divergence: measures separability of source vs. target domain samples



Figures modified after: Ben-David et al. 2010

Hdiv = 0.5 Large H-divergence ⇒ Source and Target easy to distinguish ⇒ **Poor Source ⇒ Target generalization**

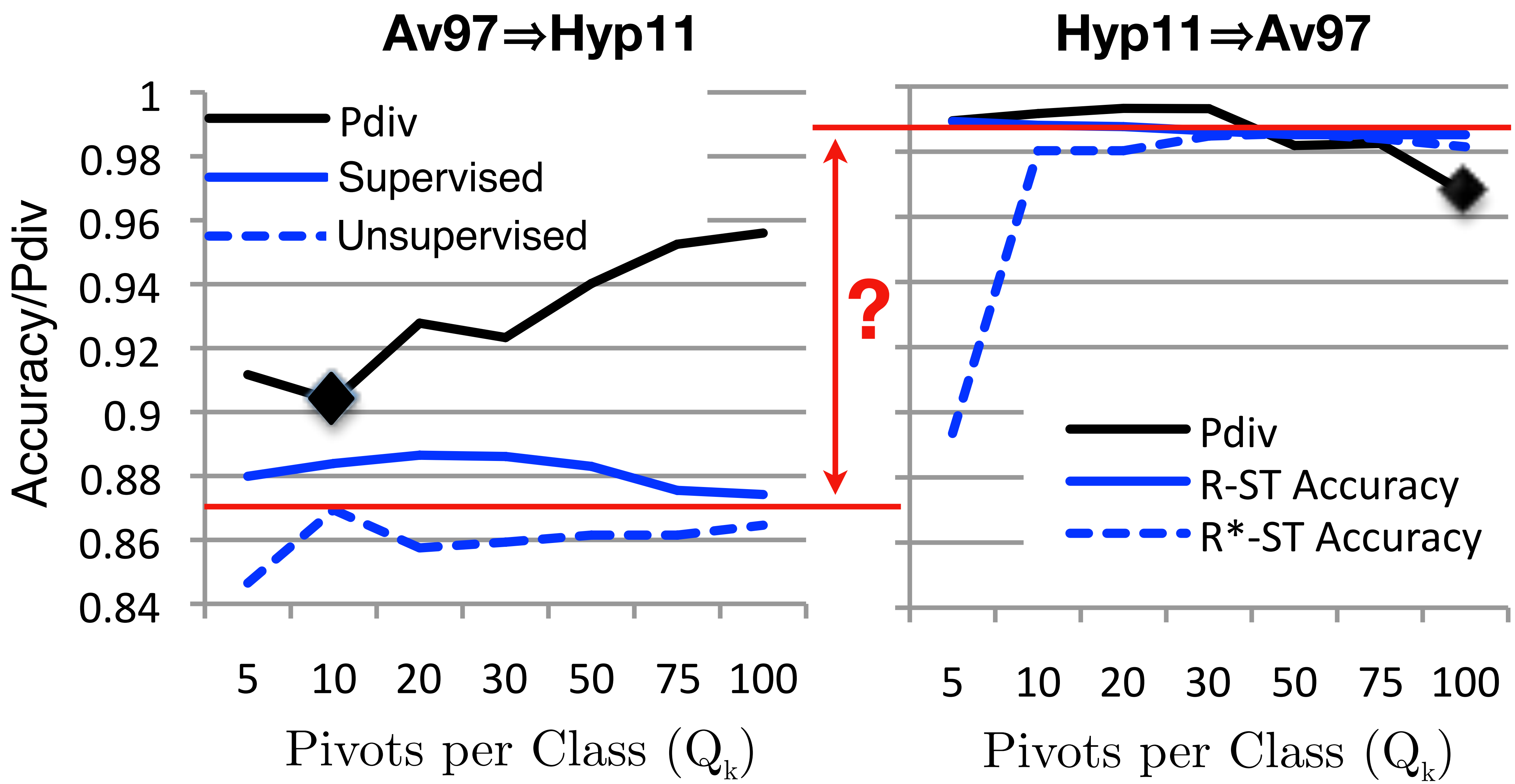
Hdiv = 0.134 Small H-divergence ⇒ Source and Target indistinguishable ⇒ **Good Source ⇒ Target generalization**

Pivot Set Selection Algorithm (Pdiv)

Good set of pivots ⇔ R-space mapping with low H-divergence

- For $Q_k \in Q_{range}$
  - Select Q_k source/target pivots per class via MCCL
  - H_k = H-divergence between $\mathrm{R}(\mathbf{p}^S, P^S)$ and $\mathrm{R}(\mathbf{p}^T, P^T)$ for class k
- Best pivot set ⇔ $\{P^S, P^T\}$ for Q_k minimizing average H_k

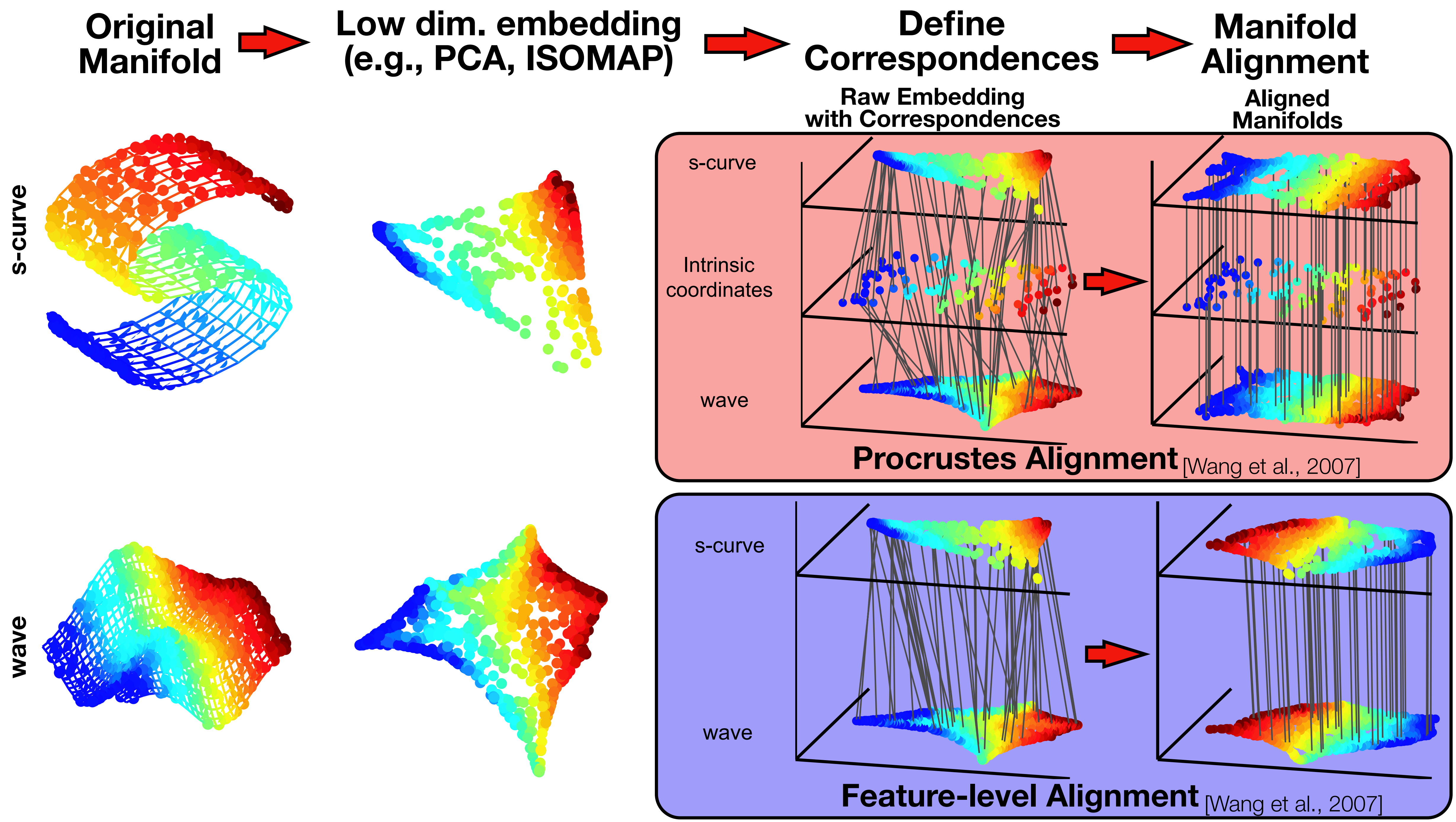
Unsupervised Domain Adaptation
MCCL Accuracy vs. Pdiv: Cuprite Av97 vs. Hyp11



- Maximum unsupervised accuracy at minimum Pdiv value (♦)
- Discrepancy between Av97⇒Hyp11 and Hyp11⇒Av97 accuracies
- Can other techniques do better?

Domain Adaptation

Related Techniques: Manifold Alignment [Ham et. al, 2005, Wang et al., 2007]



Figures modified after: Ham et al., 2005

Domain Adaptation

Related Techniques: EasyAdapt [Daumé, 2007]

Transform all (labeled and unlabeled) source and target samples using the following transformation...

Source/Target Transformations

$$\mathrm{T}^S(\mathbf{x}^S) = [\mathbf{x}^S, \mathbf{x}^S, \mathbf{0}^n]$$

$$\mathrm{T}^T(\mathbf{x}^T) = [\mathbf{x}^T, \mathbf{0}^n, \mathbf{x}^T]$$

Dot (or Kernel) Product in Transformed Space

$$\begin{aligned}
\mathrm{T}^S(\mathbf{x}^S)\cdot\mathrm{T}^S(\mathbf{x}^S) &= [\mathbf{x}^S, \mathbf{x}^S, \mathbf{0}^n]\cdot[\mathbf{x}^S, \mathbf{x}^S, \mathbf{0}^n] \\
&= \boxed{2(\mathbf{x}^S\cdot\mathbf{x}^S)} \\
\mathrm{T}^T(\mathbf{x}^T)\cdot\mathrm{T}^T(\mathbf{x}^T) &= [\mathbf{x}^T, \mathbf{x}^T, \mathbf{0}^n]\cdot[\mathbf{x}^T, \mathbf{x}^T, \mathbf{0}^n] \\
&= \boxed{2(\mathbf{x}^T\cdot\mathbf{x}^T)} \\
\mathrm{T}^S(\mathbf{x}^S)\cdot\mathrm{T}^T(\mathbf{x}^T) &= [\mathbf{x}^S, \mathbf{x}^S, \mathbf{0}^n]\cdot[\mathbf{x}^T, \mathbf{0}^n, \mathbf{x}^T] \\
&= \boxed{\mathbf{x}^S\cdot\mathbf{x}^T}
\end{aligned}$$

Samples in **same domain** twice the weight as samples from **different domains**

Designed for **binary** classification problems [Daumé, 2010]

Unsupervised Domain Adaptation
Comparisons to Related Work

Problem: both manifold alignment and EasyAdapt **supervised** techniques
Solution: use MCCL-selected target pivots as "labeled" target data
Results:

	Av97⇒Hyp11		Hyp11⇒Av97		Overall	
	Mean	**Std**	**Mean**	**Std**	**Mean**	**Std**
Baseline	0.7429	0.0098	0.9428	0.0057	0.8429	0.0078
Procrustes	*0.7685*	0.0134	0.8380	0.0175	0.8033	0.0155
Feature-level	0.7021	0.0152	0.8913	0.0193	0.7967	0.0173
EasyAdapt	0.7623	0.0218	*0.9683*	0.0133	*0.8653*	0.0176
MTL-Trace	0.7449	0.0123	0.9161	0.0172	0.8305	0.0148
RelTrans	*0.8277*	0.0120	*0.9687*	0.0074	*0.8982*	0.0097

Average accuracy using related techniques vs. RelTrans with $Q_k \in$ {5,10,20,50,100}
MCCL-selected target samples (pivots) / class (red=best, blue=second best)

Issues:

Procrustes/Feature-level Manifold Alignment
- Assumes single linear transformation can reconcile source/target domains

EasyAdapt / MTL
- Parameters optimized for *pairs* of classes, independent of other classes

R-space embedding captures multiclass structure of the learning problem

Unsupervised Domain Adaptation
SVM Model Selection and Regularization

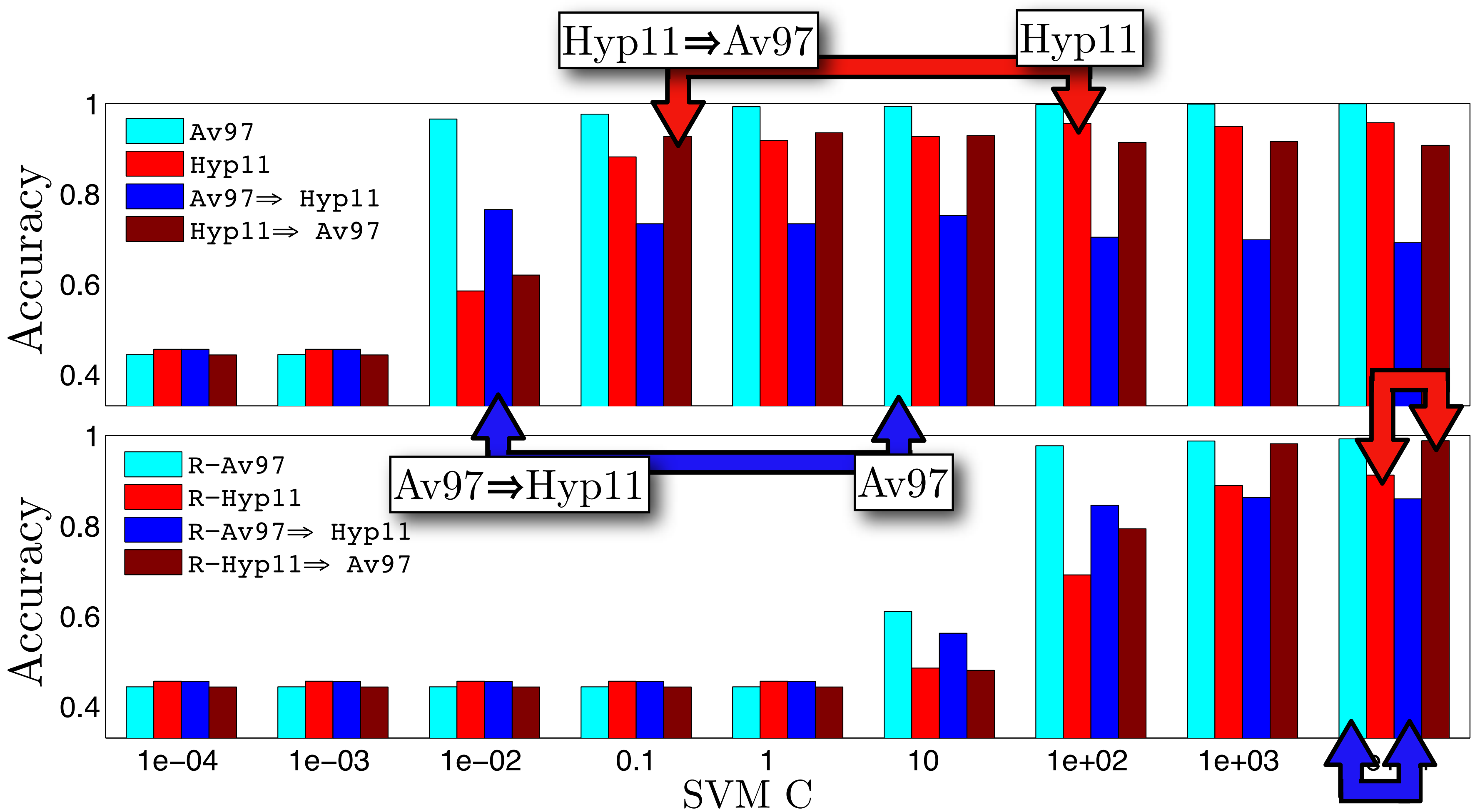
Soft-margin regularization

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

$$\text{subject to } y_i(\mathbf{w}^T\phi(\mathbf{x}_i)+b) \geq 1-\xi_i,\ \xi_i \geq 0.$$

Linear Model in kernel space ϕ

- *C* selected via cross validation on **source** domain data



- Optimal *C* in source domain *suboptimal* for domain adaptation
- Source domain *C* *optimal* for domain adaptation in R-space

Unsupervised Domain Adaptation
Intraclass Distance and Domain Adaptation Accuracy

Why >10% difference in Av97⇒Hyp11 vs. Hyp11⇒Av97 accuracy?

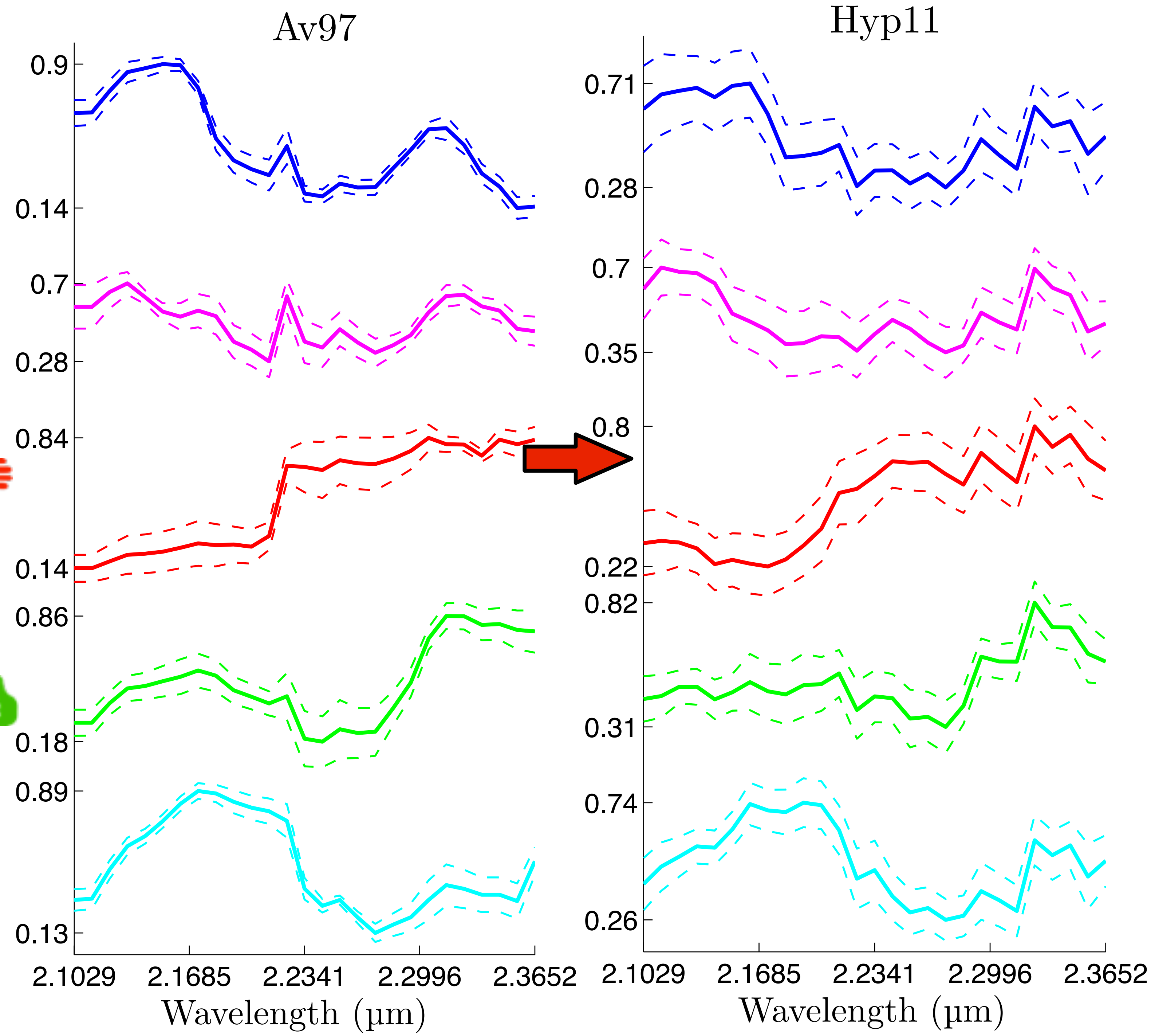
R-space accuracy limited by the amount of target variance the source-selected pivots can explain

Av97⇒Hyp11
Low source class variance +
High target class variance =
Source⇒Target accuracy = **Low**

Hyp11⇒Av97
High source class variance +
Low target class variance =
Source⇒Target accuracy = **High**

Proposed Improvements:

1. Pick better pivots
2. Improve manifold alignment



46

Unsupervised Domain Adaptation
Picking Better Pivots with the Sobolev Metric

Goal: exploit functional relationships to improve pivot selection

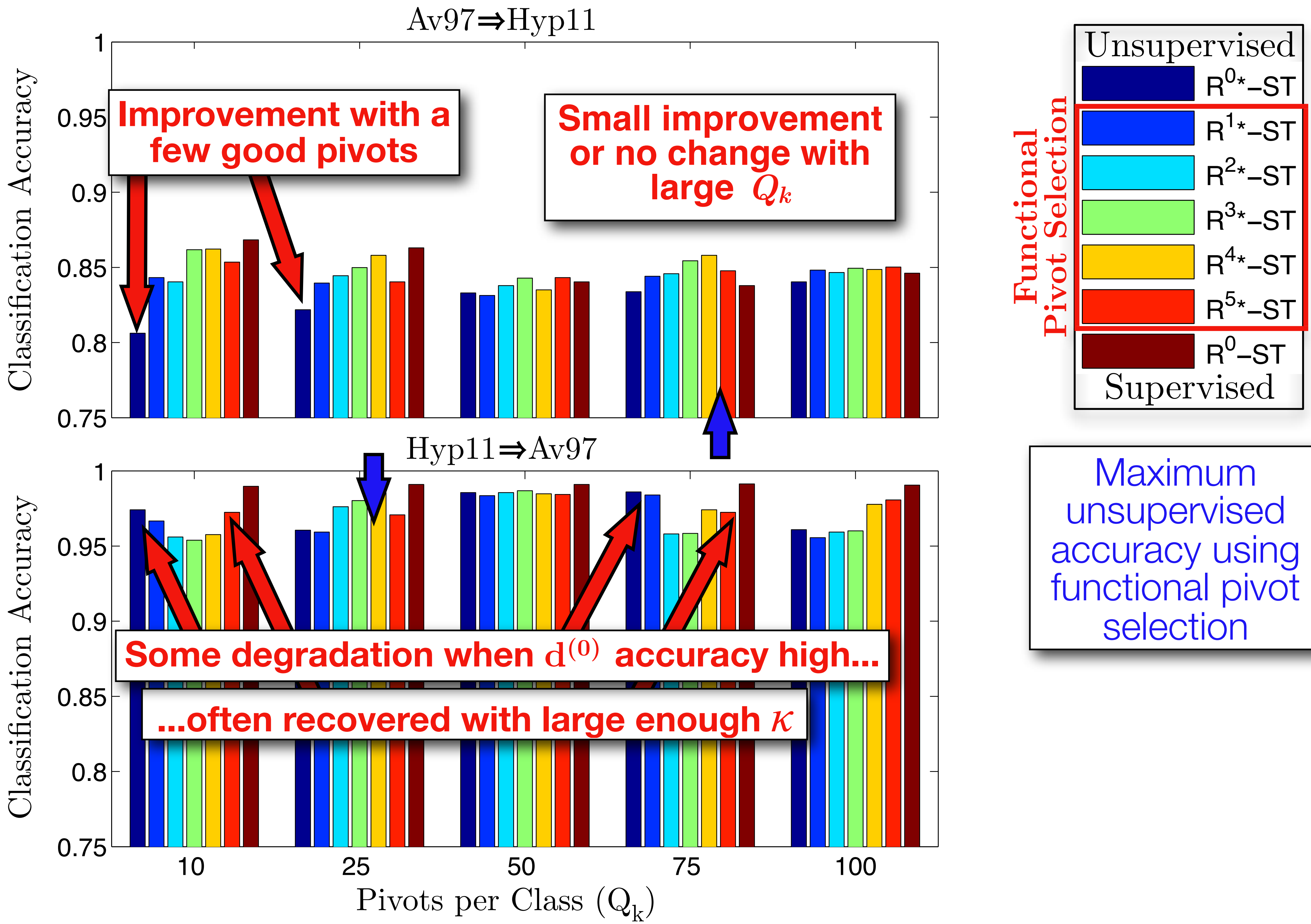
Approach: use Sobolev measure in R-transform

$$\mathrm{R}^{\kappa}(\mathbf{x}^D, P^D) = \frac{1}{\kappa+1}\sum_{l=0}^{\kappa}\left(\frac{\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_1^D)}{\sum_{i=1}^{Q}\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_i^D)}, \ldots, \frac{\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_Q^D)}{\sum_{i=1}^{Q}\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_i^D)}\right)$$

Select target pivot $\mathbf{p}_i^T = \mathbf{x}_\ell^T$ whose *first κ derivates* are most similar to source pivot $\mathbf{p}_i^S \in P^S$

$$\ell = \operatorname*{argmin}_j \|\mathrm{R}^{\kappa}(\mathbf{p}_i^S, P^S) - \mathrm{R}^{\kappa}(\mathbf{x}_j^T, P^S)\|,\ j \in \{1, \ldots, N^T\}$$

Unsupervised Domain Adaptation
Functional Pivot Selection Classification Accuracy



Maximum unsupervised accuracy using functional pivot selection

Unsupervised Domain Adaptation
Multiclass Manifold Alignment with MARTIAL

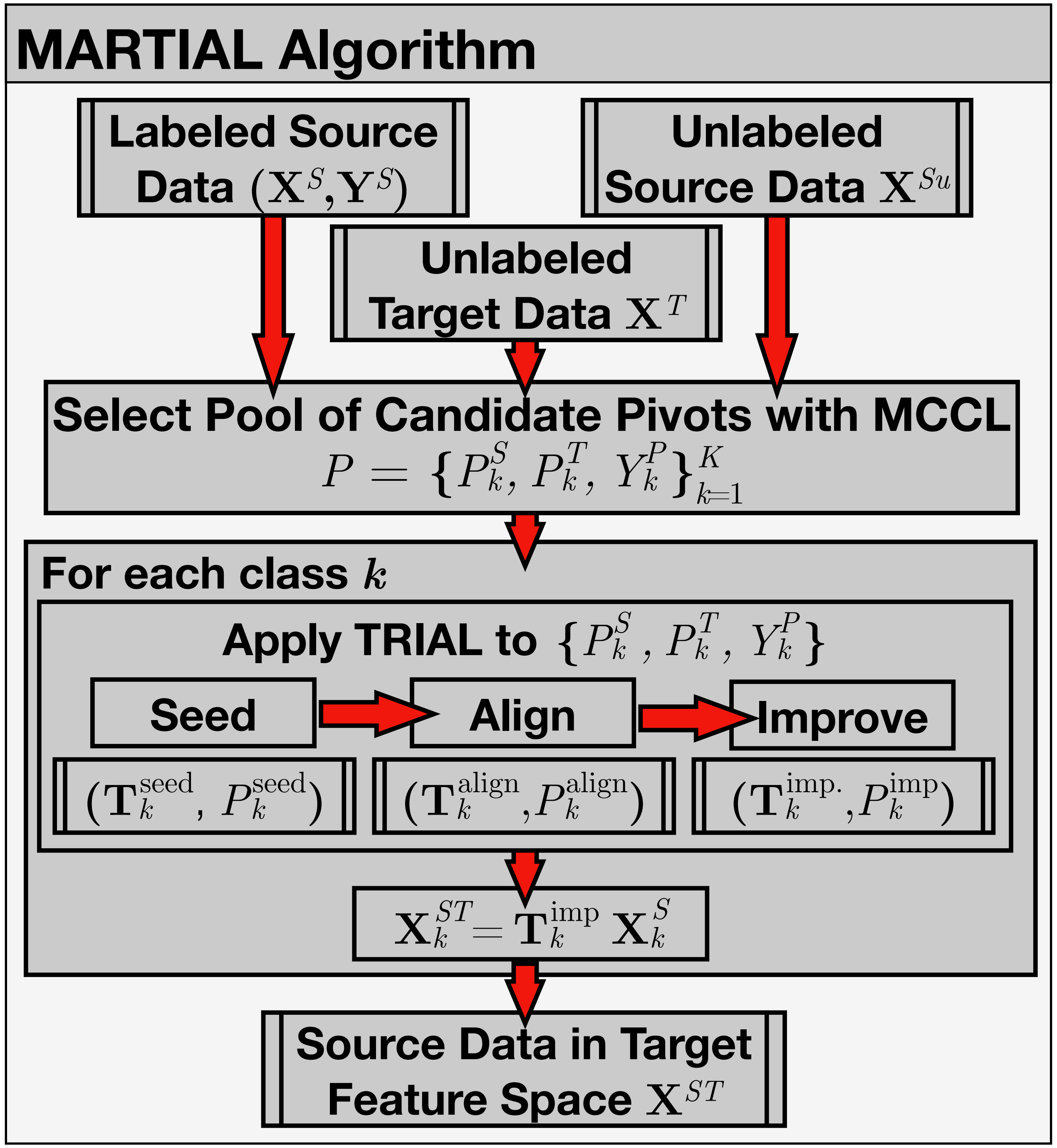
[Bue and Jermaine, WHISPERS 2013]

Current manifold alignment methods:

- assume **labeled** pivots available
- ignore class-specific distinctions

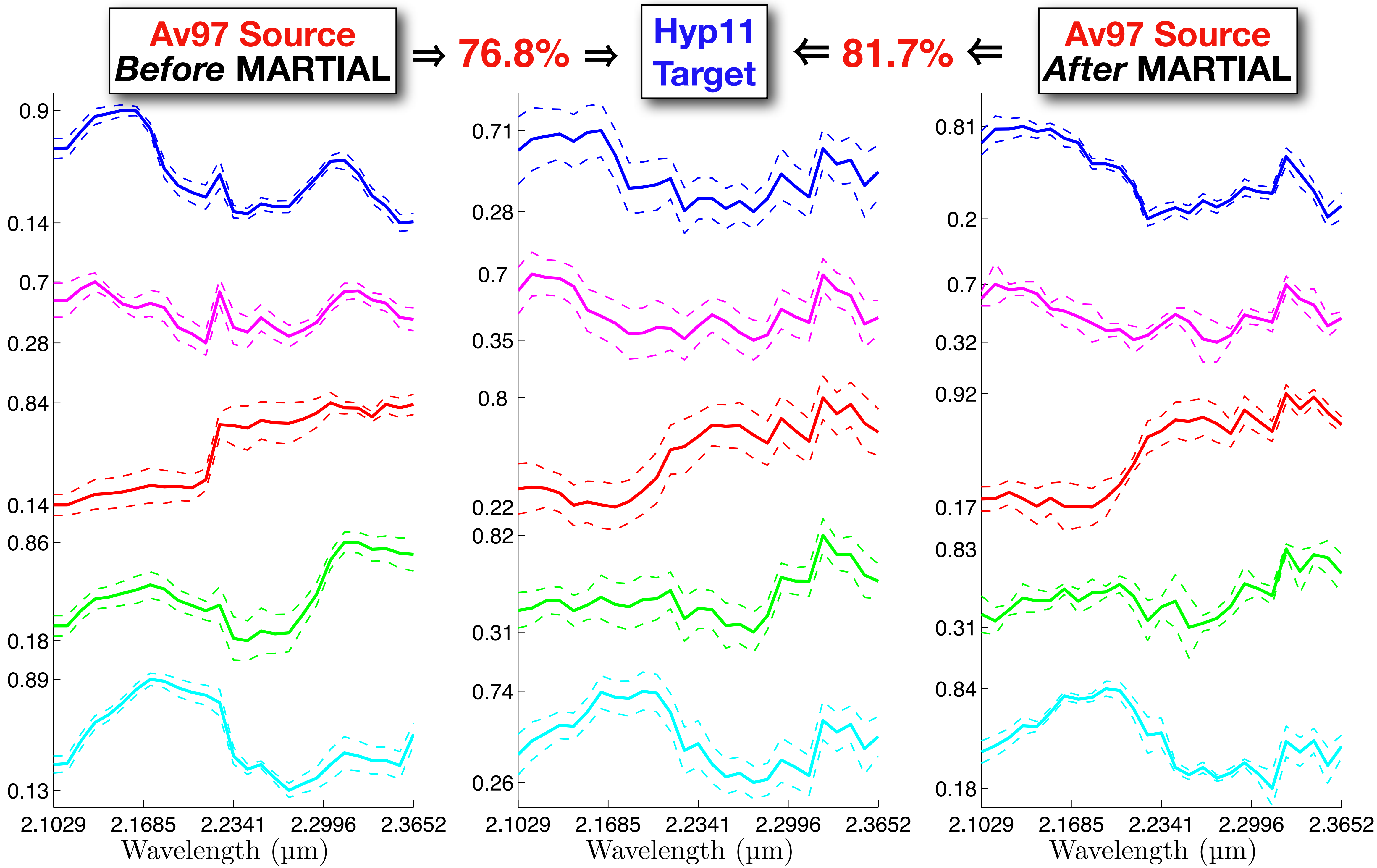
MAnifold **R**econciliation **T**hrough **I**terative **AL**ignment (**MARTIAL**)

- Uses *both* labeled and unlabeled data to align source/target manifolds
- Computes per-class transformations from source to target feature space using MCCL-selected pivots
- Extends TRIAL protein structure alignment algorithm [Venkateswaran et al.] to high-dim, multiclass manifold alignment problems

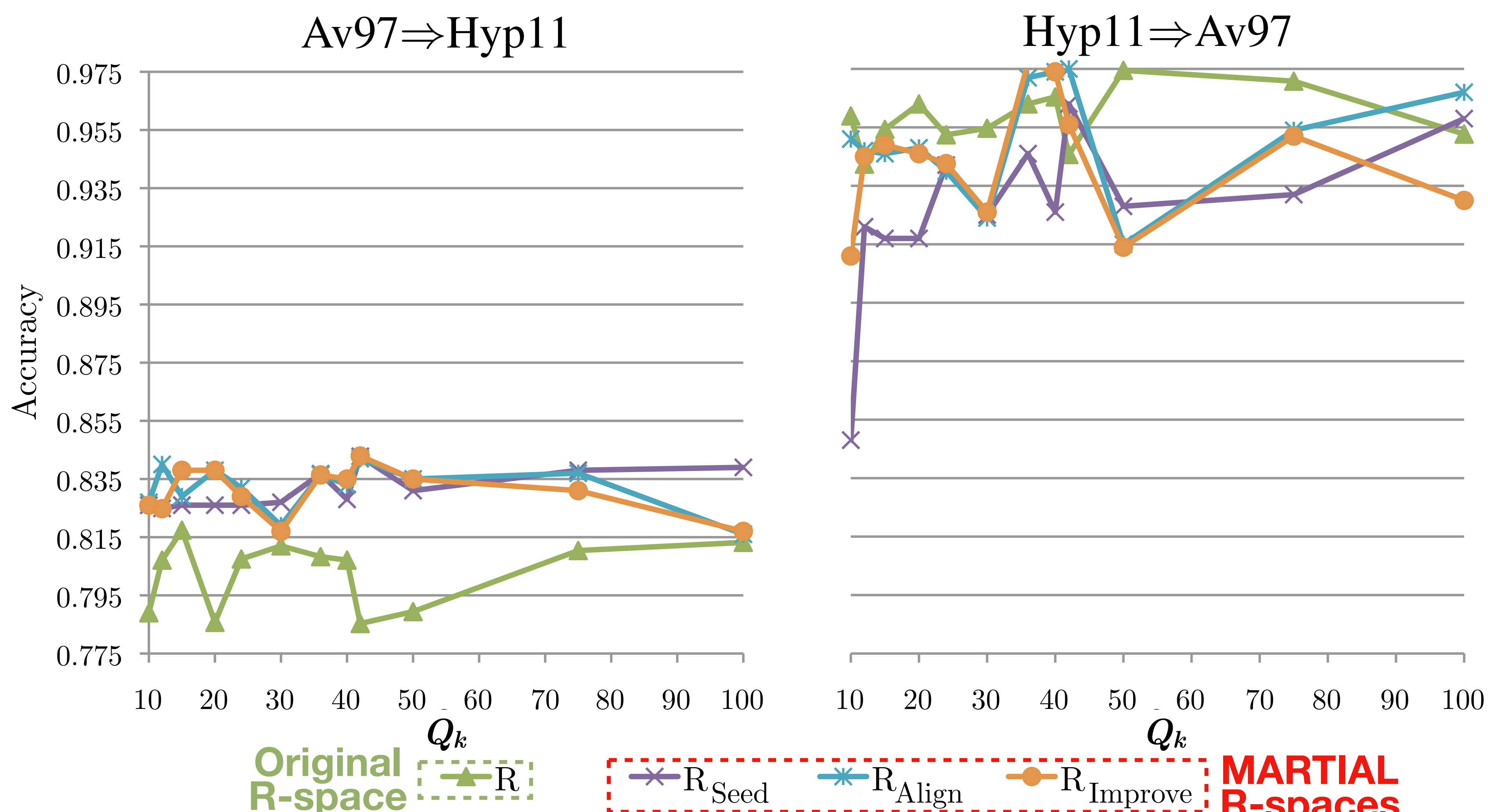


Unsupervised Domain Adaptation

$\mathrm{Av97} \Rightarrow \mathrm{Hyp11}$ classes after applying MARTIAL



MARTIAL Accuracy vs. Q_k in R-space



	ST	Proc.	R	R_{Seed}	R_{Align}	$R_{Imp.}$
Av97⇒Hyp11	71.49	73.29	80.27	*83.10*	*83.20*	83.08
Hyp11⇒Av97	93.99	82.75	*95.82*	92.69	*95.11*	94.36

12% improvement over baseline (ST)

no degradation / small improvement

10-13% improvement over Procrustes Alignment

Up to 3% improvement over original R-Space

Conclusions

Material Identification and Similarity measures

- **Adaptive similarity measures can *substantially improve* material identification results over conventional techniques often *at reduced computational expense***

Intra-domain

- Hybrid similarity measures improve classification accuracy given several *complimentary* notions of similarity
- Regularized LDA produces state-of-the-art results for *low-rank* Mahalanobis metric learning, at a fraction of the computational cost

Inter-domain

- Domain adaptation performance = limited by
  I. similarity *and* separability of source and target domains
  II. differences in intra-class variance
- Capturing *multiclass structure* with RelTrans often significantly increases domain adaptation accuracy over baseline and related techniques

Thank you 👍

Summary of Contributions

- Library-based Material Identification, CICR similarity measure
  - B. Bue, E. Merényi, and B. Csathó. Automated labeling of materials in hyperspectral imagery. IEEE Transactions on Geoscience and Remote Sensing, 48(11):4059–4070, 2010.
  - B. Bue, E. Merényi, and B. Csathó. Automated Labeling of Segmented Hyperspectral Imagery Via Spectral Matching. IEEE WHISPERS, Aug. 2009.
- Adaptive similarity measures for Intra-domain Material Identification
  - Adaptive CICR:
    B. Bue and E. Merényi. An adaptive similarity measure for classification of hyperspectral signatures. IEEE Geoscience and Remote Sensing Letters, 2012.
  - LDA-based Mahalanobis metrics for Image Segmentation:
    B. D. Bue, DR Thompson, M. Gilmore, and R. Castaño, “Metric Learning for Hyperspectral Image Segmentation,” IEEE WHISPERS, Jun. 2011.

Summary of Contributions

- Domain Adaptation for Inter-domain Material Identification
  - Supervised:
    - B. D. Bue, E. Merényi, and B. Csathó, “An Evaluation of Class Knowledge Transfer from Real to Synthetic Imagery,” IEEE WHISPERS, Jun. 2011.
    - B. D. Bue and E. Merényi, “Using spatial correspondences for hyperspectral knowledge transfer: evaluation on synthetic data,” IEEE WHISPERS, Jun. 2010.
  - Unsupervised:
    - B. D. Bue and D. R. Thompson, “Multiclass Continuous Correspondence Learning,” NIPS Domain Adaptation Workshop, Dec. 2011.
    - B. D. Bue and C. Jermaine. “Multiclass Domain Adaptation with Iterative Manifold Alignment”. Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), Jun. 2013 (to appear).

References

- [Merényi et al., 2007] E. Merényi and W. Farrand, "Classification of hyperspectral imagery with neural networks: Comparison to conventional tools," Photogrammetric Engineering Remote Sensing 2007.
- [Pouch et al., 1990] GW Pouch and DJ Campagna, "Hyperspherical direction cosine transformation for separation of spectral and illumination information in digital scanner data," Photogrammetric Engineering and Remote Sensing, vol. 56, no. 4, pp. 475–479, 1990.
- [Schott et al., 1999] JR Schott, SD Brown, RV Raqueno, HN Gross, and G Robinson, "Advanced synthetic image generation models and their application to multi/hyperspectral algorithm development," Proceedings of SPIE, vol. 3584, pp. 211, 1999
- [Csathó et al., 1998] B Csathó, WB Krabill, J Lucas, and T Schenk, "A multisensor data set of an urban and coastal scene," International Archives of Photogrammetry and Remote Sensing, vol. XXXII (3/2), pp. 26–31, Jan 1998.
- [Merényi et al., 2007] E. Merényi, B. Csatho, and K. Tasdemir, "Knowledge discovery in urban environments from fused multi-dimensional imagery", In Proc. 4th IEEE GRSS/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (URBAN 2007).
- [Clark et al., 2003] R. Clark, G. Swayze, K. Livo, R. Kokaly, S. Sutley, J. Dalton, R. McDougal, and C. Gent, "Imaging spectroscopy: Earth and planetary remote sensing with the USGS Tetracorder and expert systems," Journal of Geophysical Research-Planets, vol. 108, no. 12, p. 5131, 2003.
- [Zhou et al. 2012] J. Zhou, J. Chen, J. Ye, "Multi-Task Learning: Theory, Algorithms, and Applications," The Twelfth SIAM Information Conference on Data Mining, April 27, 2012

References

- [Clark et al., 1990] R. N. Clark, T. V. V. King, M. Klejwa, G. A. Swayze, and N. Vergo, “High spectral resolution reflectance spectroscopy of minerals,” J. Geophys. Res, vol. 95, no. 8, pp. 12653–12,680, 1990.
- [Rajan, 2006] S. Rajan, Knowledge Transfer Techniques for Dynamic Environments, Doctoral Dissertation - University of Texas at Austin, 2006.
- [Mendenhall et al., 2008] M. Mendenhall and E. Merényi, “Relevance-Based Feature Extraction for Hyperspectral Images,” Neural Networks, IEEE Transactions on, vol. 19, no. 4, pp. 658–672, 2008.
- [Weinberger et al., 2009] K. Q. Weinberger and L. Saul, “Distance metric learning for large margin nearest neighbor classification,” Journal of Machine Learning Research, vol. 10, pp. 207–244, 2009.
- [Goldberger et al., 2005] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, “Neighbourhood components analysis,” Advances in Neural Information Processing Systems, vol. 17, pp. 513–520, 2005.
- [Herold et al, 2004] M. Herold, D. Roberts, M. Gardner, and P. Dennison, “Spectrometry for urban area remote sensing—development and analysis of a spectral library from 350 to 2400 nm,” Rem. Sens. of Environ., vol. 91, pp. 304–319, 2004.

Feature-based vs. Similarity-based Classification

Input Data (features)

- "Original" representation
- High-dimensional
- Noisy / redundant features

Feature-based Classification

Representation Learning

- Learn improved feature representation
  - *Feature selection/weighting*
  - *Dimensionality reduction*
- Low(er)-dimensional
- More informative features

Feature-based Classification

- Features define inference rules
  - *Naive Bayes*
  - *Multilayer Perceptron*
  - *Graphical Models*
  - *Decision Trees*

Test Predictions

- Predictions based on relevance of features w.r.t. class models derived from training samples

Similarity-based Classification ✓

Similarity Learning

- Learn similarity measure to compare samples
- Learned similarity measure <=> learned feature representation

Similarity-based Classification

- Similarities define inference rules
  - *k-Nearest-Neighbor (kNN)*
  - *Learning Vector Quantization (LVQ)*
  - *Support Vector Machine (SVM)*

Test Predictions

- Predictions based on similarity w.r.t. prototypes derived from training samples

Similarity/Distance Learning ⟷ Representation Learning

Evaluation on Ocean City AVIRIS Image
CICR vs. Feature Selection/Dimensionality Reduction

MinDist Accuracy

	Baseline		Feature Selection/Dimensionality Reduction						$\mathbf{d_{CICR}}$	
	$\mathbf{d_{CI}}$	$\mathbf{d_{CR}}$	χ^2_{25}	χ^2_{50}	**RFE**	$\mathbf{L^1}$	**PCA**	$\mathbf{LDA_{FW}}$	**LDA**	**LS**
Minor	0.8866	0.6580	0.8376	0.8875	0.8848	0.8872	0.8819	*0.9172*	*0.9032*	0.9055
	0.0076	0.0208	0.0090	0.0083	0.0114	0.0163	0.0099	0.0095	0.0049	0.0065
Major	0.9250	0.7750	0.8493	0.8917	0.9330	0.8638	0.9203	*0.9714*	*0.9721*	0.9754
	0.0186	0.0101	0.0093	0.0143	0.0159	0.0573	0.0127	0.0062	0.0075	0.0029
Combined	0.8654	0.6730	0.8302	0.8441	0.8617	0.8310	0.8672	*0.9176*	*0.9076*	0.9207
	0.0069	0.0075	0.0064	0.0060	0.0056	0.0125	0.0063	0.0049	0.0111	0.0045

$\boldsymbol{\chi}^2_p$	p-best features, according to $\boldsymbol{\chi}^2$ criterion
RFE	Recursive Feature Elimination [Guyon et al., 2002]
$\boldsymbol{L}^1$	Features weighted according to L^1-penalized GLM
PCA	PCA components explaining 99% of variance
$\mathbf{LDA_{FW}}$	Feature-weighted LDA, $\mathrm{T}(\mathbf{x})$: $\mathcal{R}^n \rightarrow \mathcal{R}^{K\text{-}1}$

Adaptive Sobolev Metric:
Ocean City Classification Results

Per-derivate Accuracy:

	$\mathbf{d}^{(0)}$	$\mathbf{d}^{(1)}$	$\mathbf{d}^{(2)}$	$\mathbf{d}^{(3)}$
Minor	0.8863	0.7717	0.6362	0.6364
	0.0134	0.0128	0.0067	0.0104
Major	0.9256	0.9299	0.8857	0.8759
	0.0108	0.0093	0.0104	0.0110
Combined	0.8698	0.8276	0.7219	0.7082
	0.0056	0.0137	0.0077	0.0090

Baseline = $\mathrm{d}^{(0)}$
(Euclidean distance)

Sobolev Accuracy (UW=equal α_l weights, LS=line search):

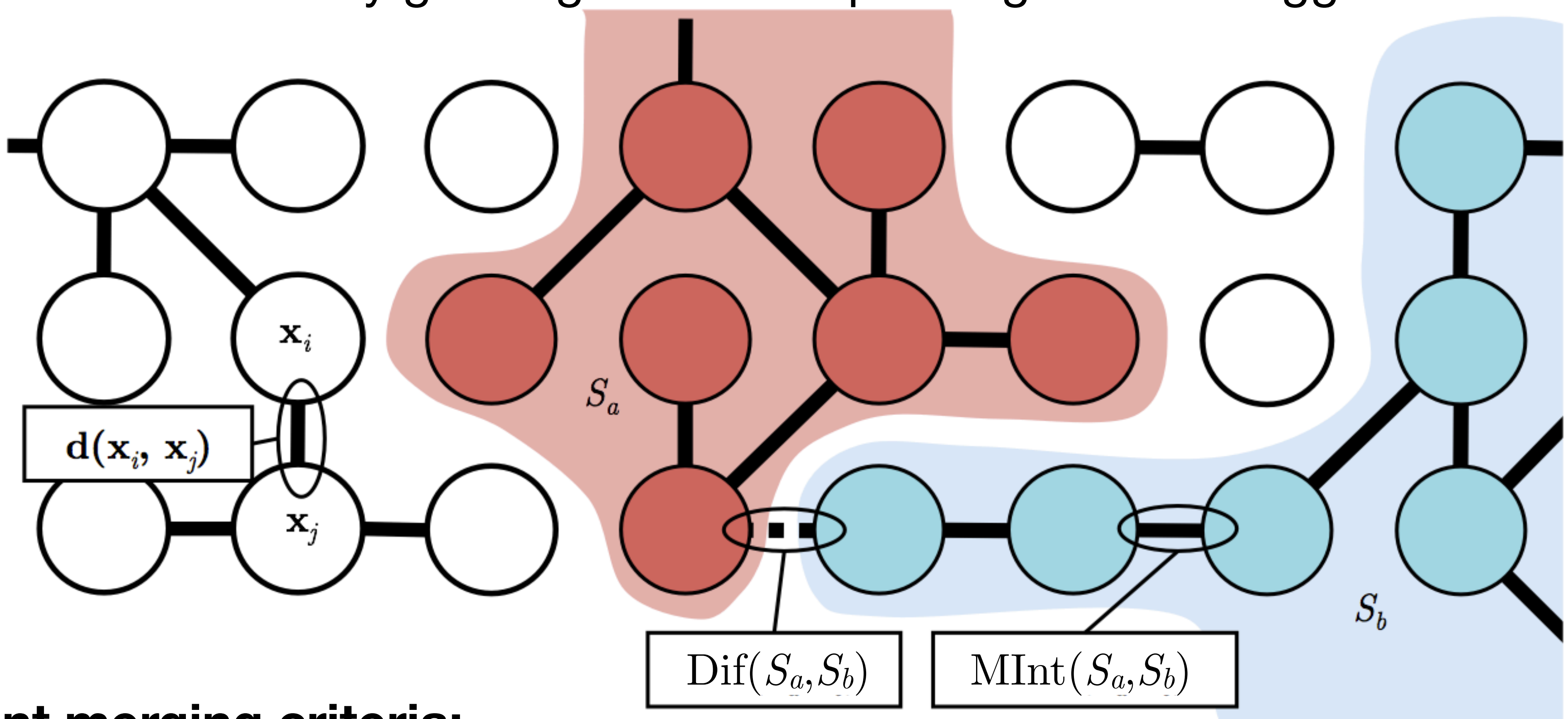
	$\mathbf{d}_{\mathbf{Sobolev}},\ \kappa = 1$			$\mathbf{d}_{\mathbf{Sobolev}},\ \kappa = 2$			$\mathbf{d}_{\mathbf{Sobolev}},\ \kappa = 3$		
	UW	**LDA**	**LS**	**UW**	**LDA**	**LS**	**UW**	**LDA**	**LS**
Minor	0.9047	0.9108	0.9210	0.8808	0.9090	0.9254	0.8627	0.9052	0.9268
	0.0134	0.0052	0.0043	0.0090	0.0077	0.0025	0.0084	0.0121	0.0066
Major	0.9616	0.9659	0.9707	0.9630	0.9703	0.9830	0.9543	0.9688	0.9804
	0.0087	0.0120	0.0063	0.0088	0.0178	0.0022	0.0085	0.0049	0.0031
Combined	0.9123	0.9121	0.9257	0.8976	0.9011	0.9300	0.8925	0.8967	0.9321
	0.0071	0.0070	0.0021	0.0079	0.0081	0.0045	0.0087	0.0075	0.0052

- LDA better than UW, but usually only by ≤1%
- UW, LDA accuracy decreases as κ increases

Feature-weighted Metric Learning
Graph-based Segmentation Algorithm [Felzenszwalb 2004]

Image = 8-connected graph weighted by $\mathrm{d}(\mathbf{x}_i,\mathbf{x}_j)$ between adjacent pixels $\mathbf{x}_i$ and $\mathbf{x}_j$

Segments connected by growing minimum spanning tree with agglomerative clustering



Segment merging criteria:

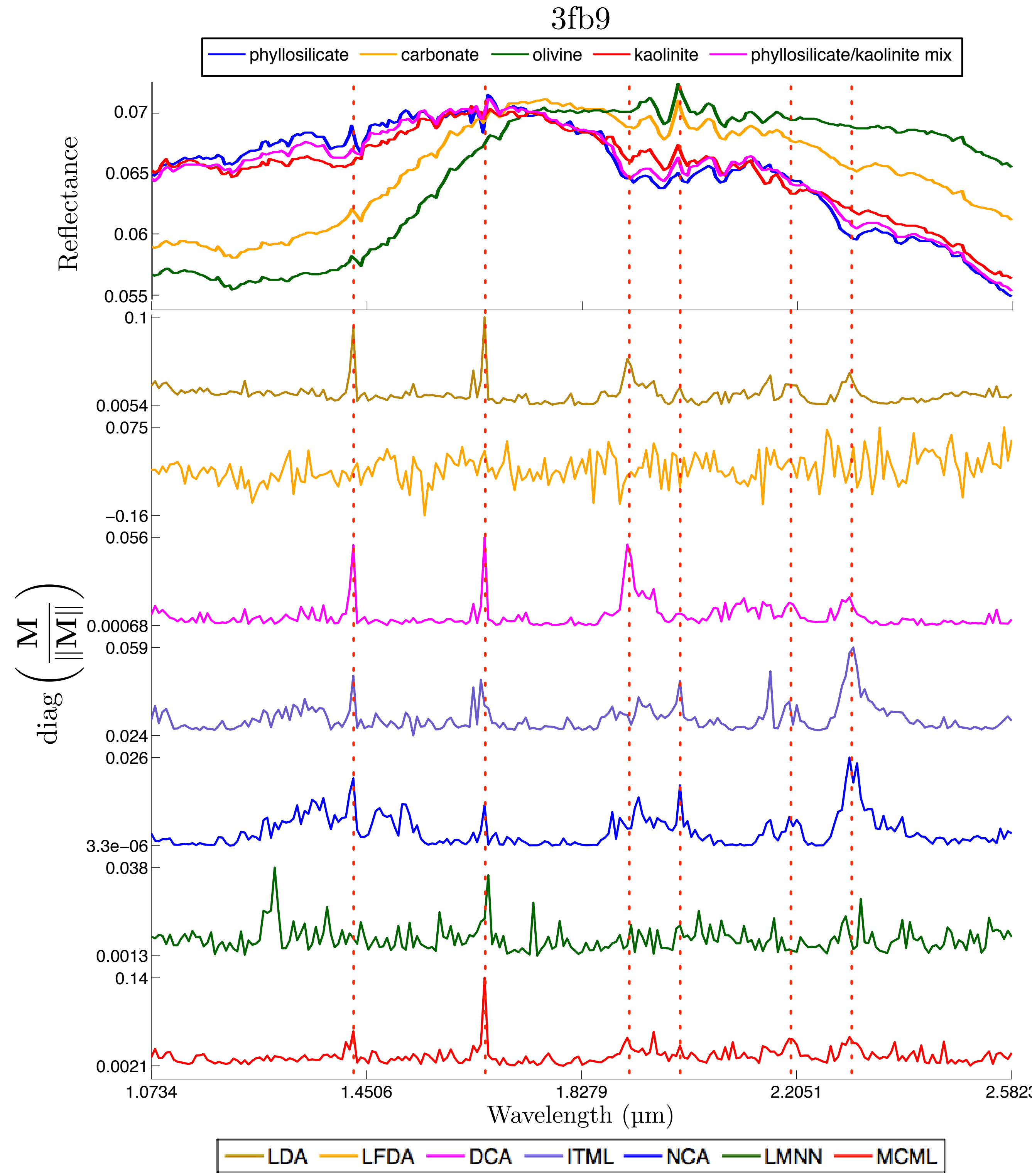
cross segment distance $\mathrm{Dif}(S_a,S_b)$ > minimum internal weight $\mathrm{MInt}(S_a,S_b)$

$$\mathrm{MInt}(S_a, S_b) = \min\left(\mathrm{Int}(S_a) + \frac{b}{|S_a|}, \mathrm{Int}(S_b) + \frac{b}{|S_b|}\right)$$

segments depends on: $\mathrm{d}(\mathbf{x}_i,\mathbf{x}_j)$ and *b* (small *b* = many segments,
large *b* = few segments)

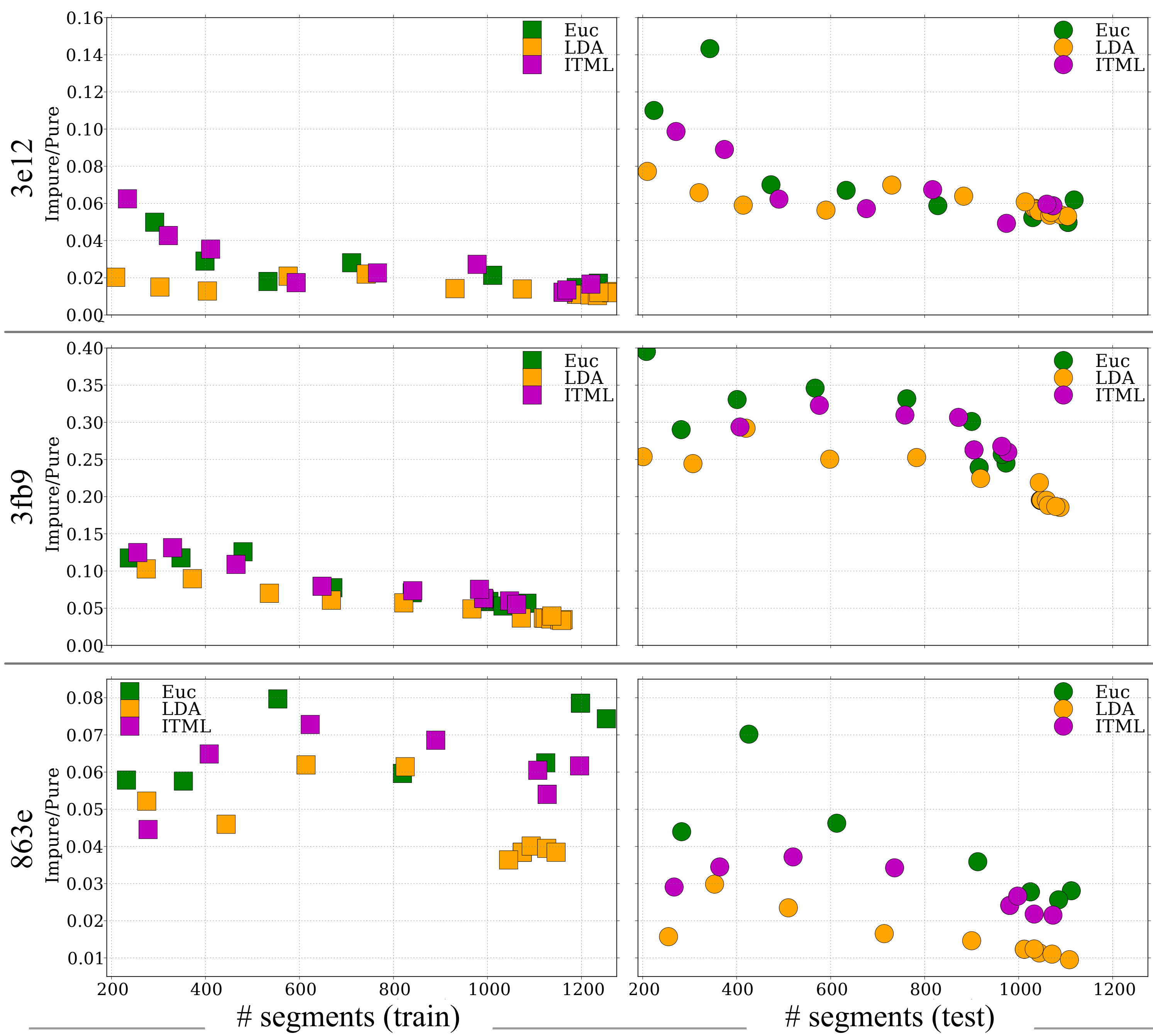
Feature-weighted Metric Learning

Diagonals of Learned Mahalanobis Matrices



Segmentation Results: CRISM Image 863e Euclidean vs. LDA and ITML Metrics

Impurity Ratios (per pixel)

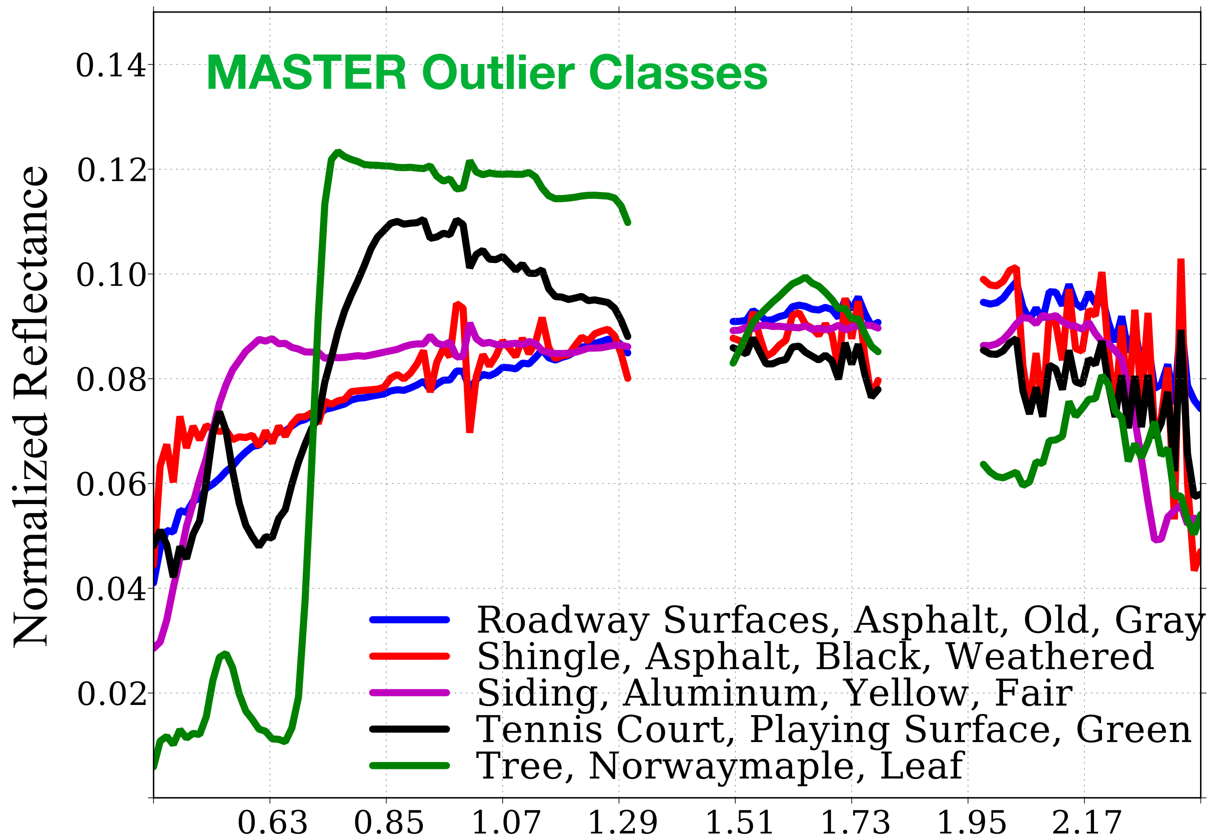


Average Pure Superpixels

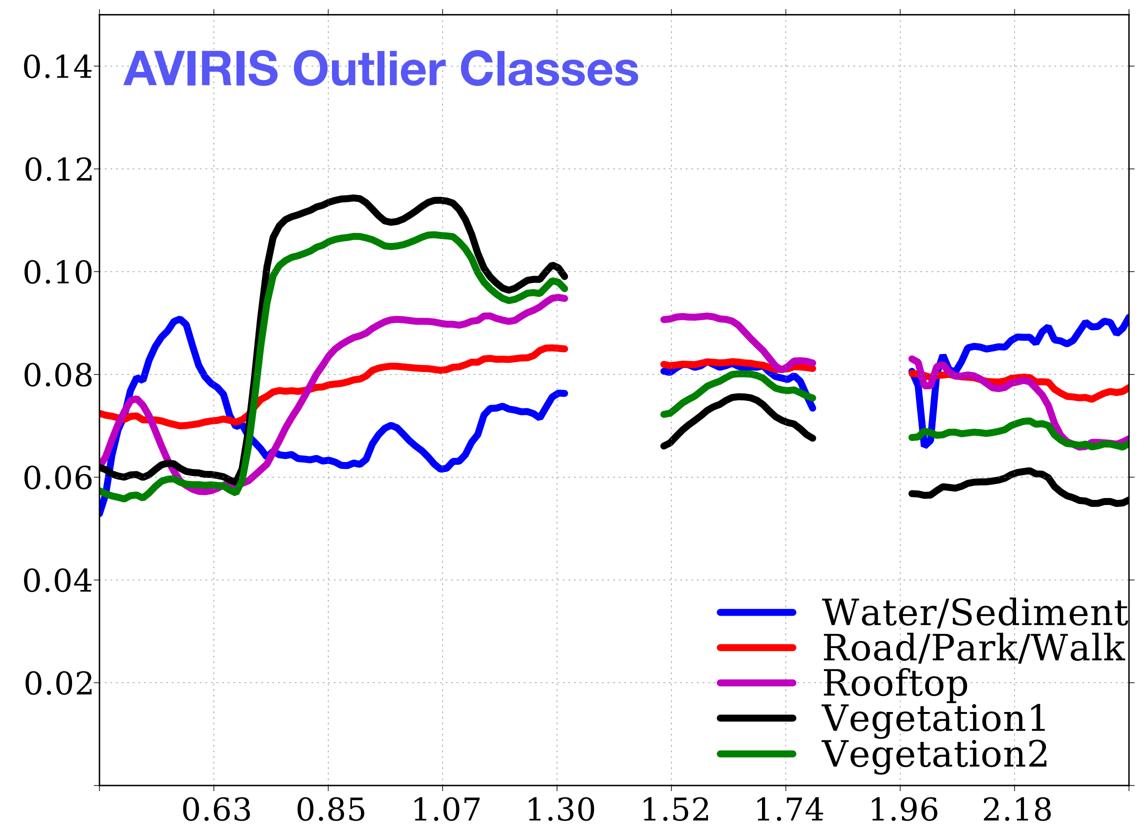
Class (# pixels)	EUC	LDA	ITML
FeMg Smectite (6443)	26	49	48
Kaolinite (4051)	98	99	99
Montmorillonite (10901)	11	31	17
Nontronite (4753)	37	52	40
Neutral Region (115225)	97	99	98
Average	53	66	60

Supervised Domain Adaptation
Outlier Detection Classes

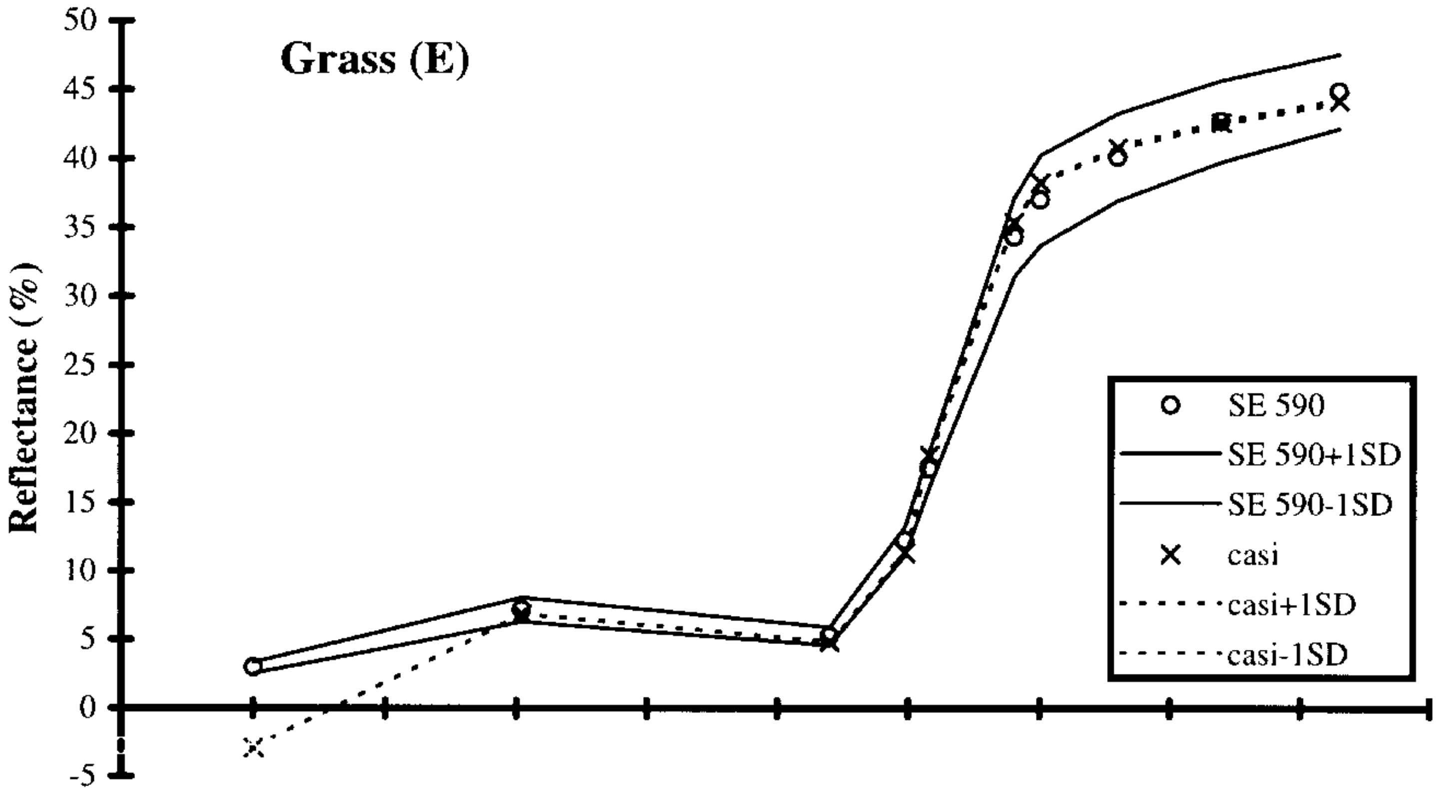
Synthetic hyperspectral (**HYDICE, 210 bands**)
=> **Synthetic** multispectral (**MASTER, 12 bands**)



Synthetic hyperspectral (**HYDICE**)
=> **Real** hyperspectral (**AVIRIS**)



Supervised Domain Adaptation
Radiance to Reflectance EML Approximation

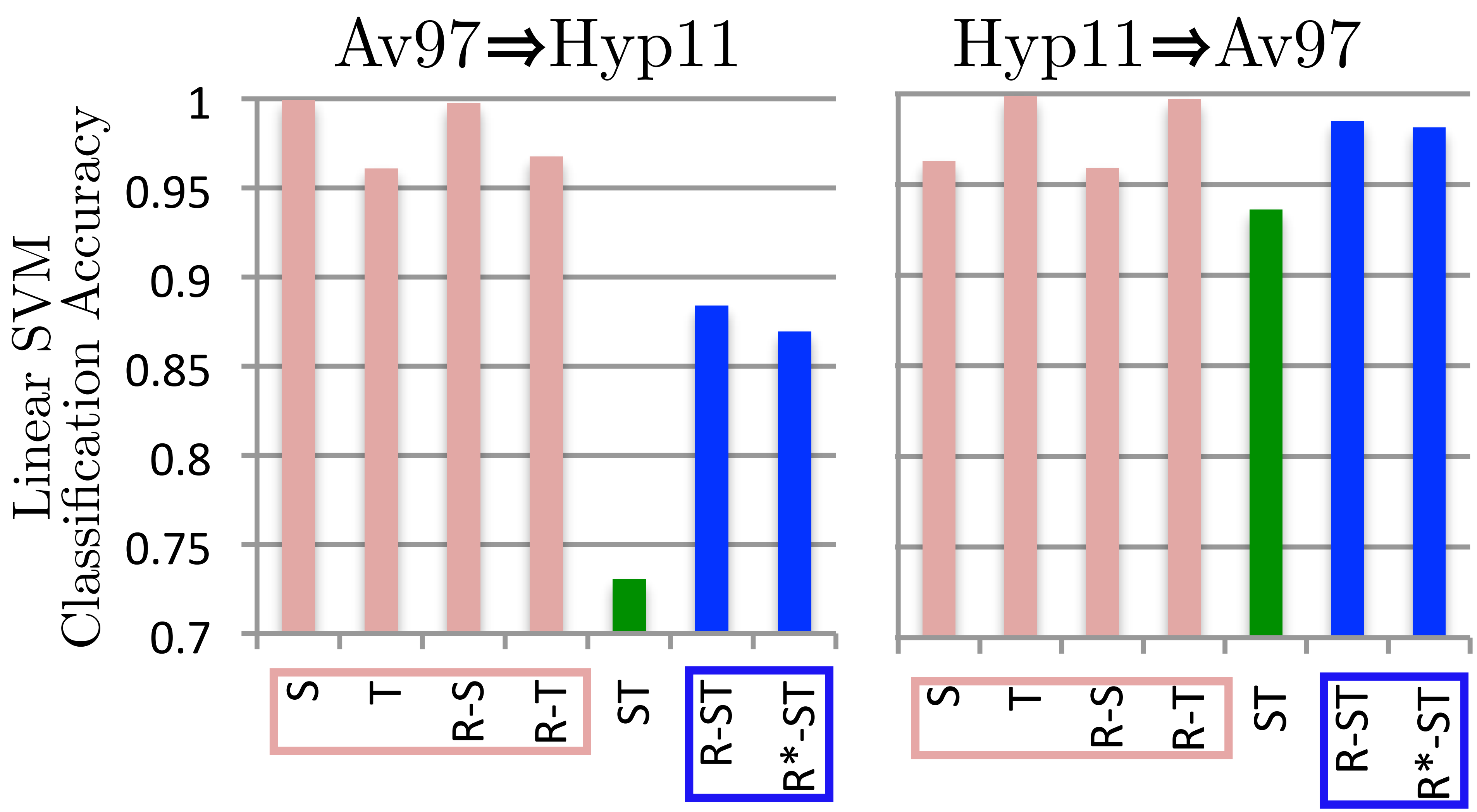


SE 590: Spectron SE 590 Field Spectroradiometer spectrum

CASI: EML-corrected Compact Airborne Spectrographic Imager spectrum

EML linear approximation closely matches field-measured spectrum

Unsupervised Domain Adaptation
Cuprite Av97 vs. Hyp11 Results



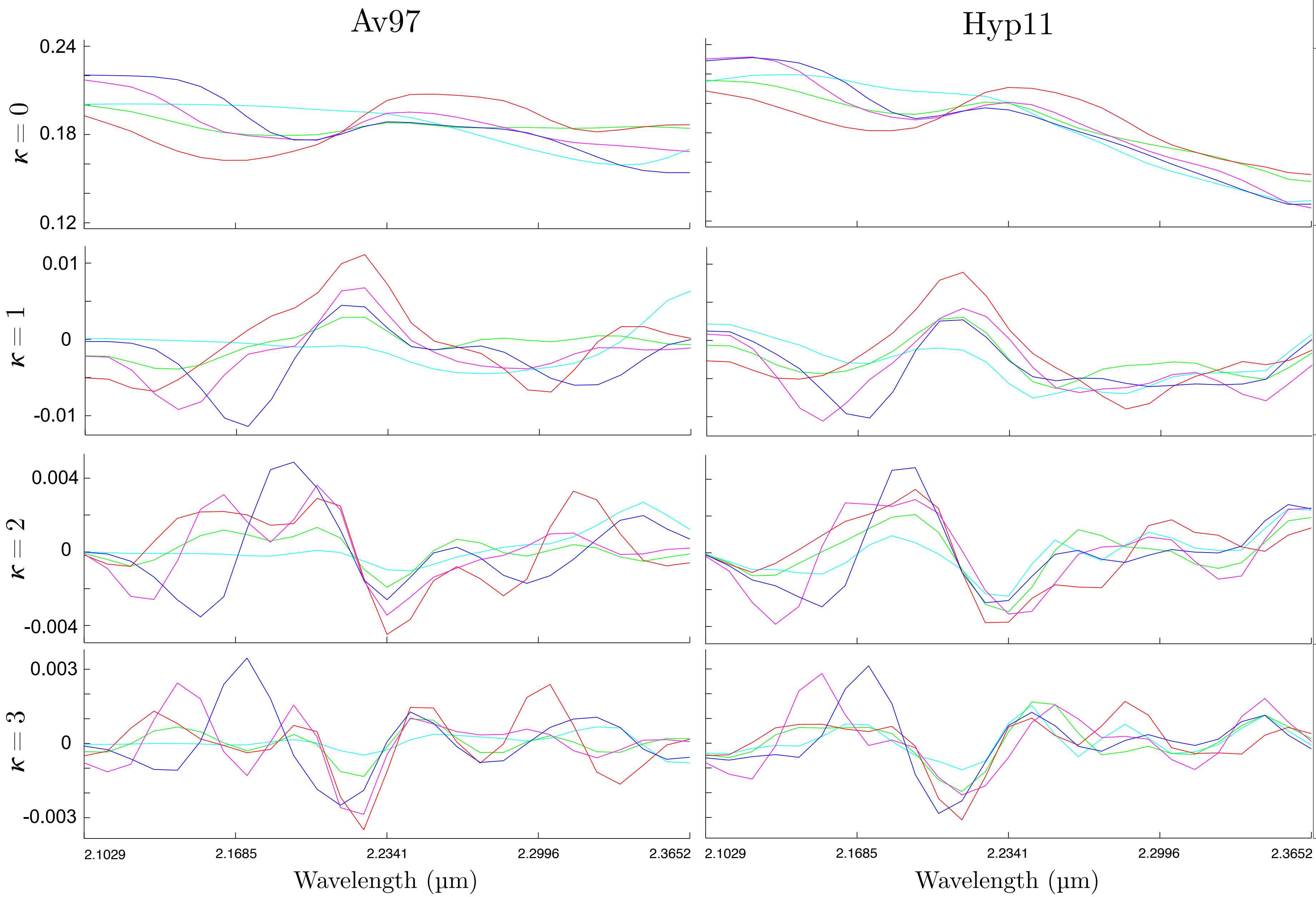
S, **T**: source, target
R-S, **R-T**: R-space source, target
ST: source-to-target (baseline)
R-ST: supervised MCCL
R*-ST: unsupervised MCCL

- R-space as discriminative as original feature space
- R-ST R*-ST comparable (≤2% diff)

Unsupervised Domain Adaptation
Av97, Hyp11 Per-derivate Accuracy

$\mathrm{d}^{\kappa}/\mathrm{d}^{\mathbf{wavelength}}$

Class Means



Accuracy

Av97	Hyp11	Av97⇒ Hyp11	Hyp11⇒ Av97
1.000	0.962	*0.744*	*0.936*
1.000	0.972	*0.764*	*0.909*
0.999	0.941	0.544	0.661
0.996	0.912	0.477	0.443

Kabsch Algorithm (TRIAL transformations)

- Given Q paired samples $(P^S,\ P^T)$, the rotation matrix $\mathbf{T}$ which minimizes

$$\varepsilon = \mathrm{RMSD}(\mathbf{T}\cdot P^S,\ P^T)$$

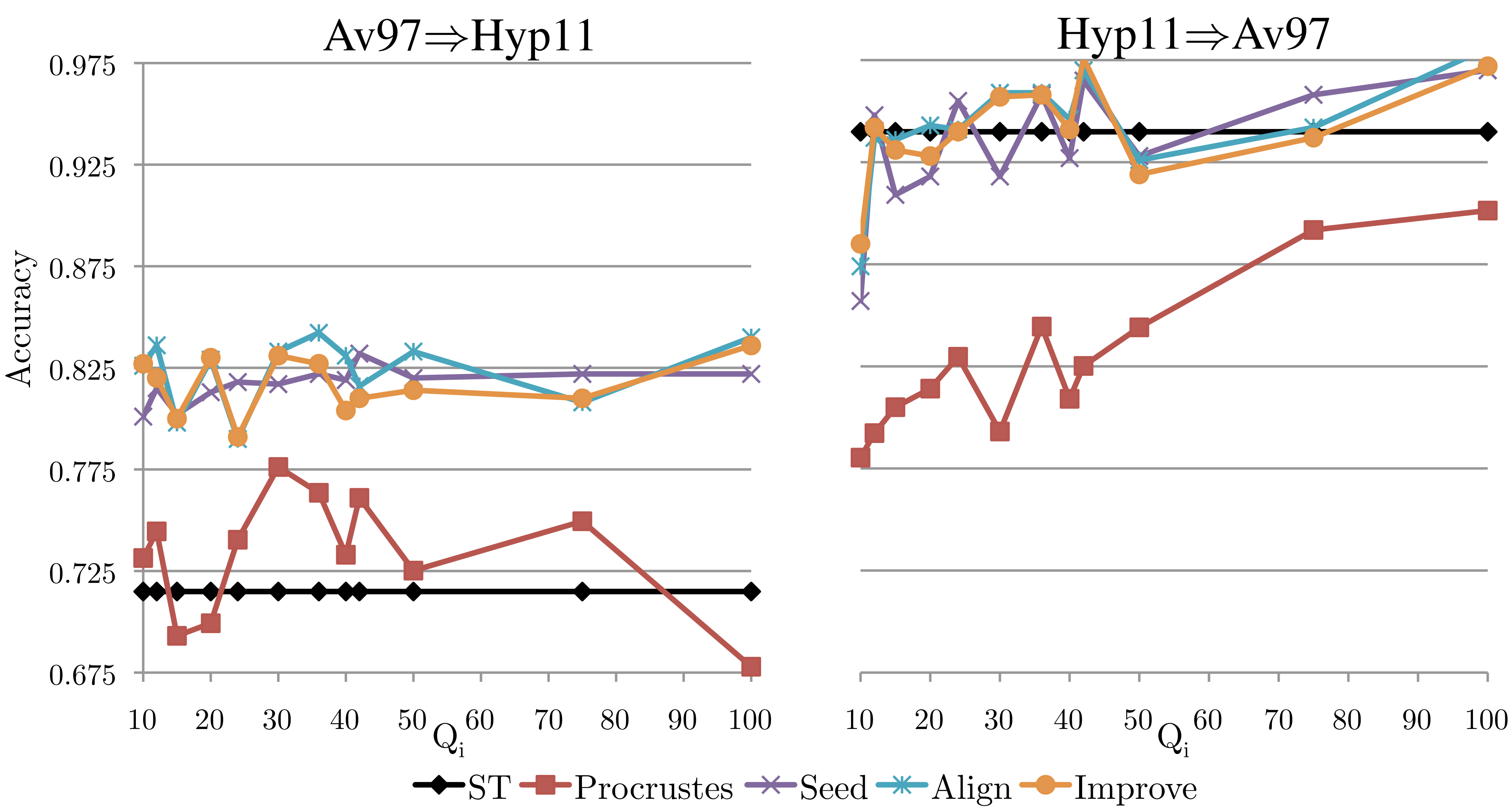
is computed as follows:

$$\mathbf{T} = \mathbf{W}\mathrm{diag}(\mathbf{s})\mathbf{V}^T$$

where

$$\mathbf{VDW}^T = \mathrm{SVD}(\mathrm{COV}(P^S, P^T))$$
$$\mathbf{s} = [1, \ldots, 1, \mathrm{sign}(\det(\mathbf{WV}^T))]$$

MARTIAL Accuracy vs. Q_k



	ST	Proc.	Seed	Align	Imp.
Av97⇒Hyp11	71.49	73.29	81.69	82.35	81.67
Hyp11⇒Av97	93.99	82.75	93.43	94.28	94.05

RelTrans vs. Baseline Techniques

Av97⇒Hyp11		**S**	**T**		**ST**					
	Base	0.9963	0.9679		0.7429					
		0.0027	0.0098		0.0098					
	$\mathbf{Q_k}$	**R-S**	**R-T***	**R-T**	**R-ST***	**R-ST**	**PivST***	**PivST**	**AugST***	**AugST**
	Mean	0.9918	0.9189	0.9211	0.8258	0.8498	0.7478	0.8717	0.7469	0.8638
	Std	0.0042	0.0119	0.0118	0.0111	0.0133	0.0182	0.0120	0.0131	0.0155
Hyp11⇒Av97		**S**	**T**		**ST**					
	Base	0.9679	0.9963		0.9428					
		0.0098	0.0027		0.0057					
	$\mathbf{Q_k}$	**R-S**	**R-T***	**R-T**	**R-ST***	**R-ST**	**PivST***	**PivST**	**AugST***	**AugST**
	Mean	0.9202	0.9925	0.9920	0.9668	0.9908	0.9665	0.9818	0.9611	0.9653
	Std	0.0130	0.0043	0.0037	0.0073	0.0037	0.0121	0.0052	0.0091	0.0065

PivST: Classifier trained with pivots alone (PivST*=unsupervised)

AugST: Classifier trained with source data + pivots (AugST*=unsupervised)

RelTrans: Best performance in unsupervised case
Some misalignment in supervised case (between src/tgt means)