RICE UNIVERSITY

Adaptive Similarity Measures for Material Identification in Hyperspectral Imagery

by

Brian D. Bue

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

Doctor of Philosophy

Approved, Thesis Committee:

Erzsébet Merényi

Erzsébet Merényi, Chair
Research Professor of Statistics and
Electrical and Computer Engineering

Chris Jermaine
Associate Professor of Computer Science

S. Devika

Devika Subramanian
Professor of Computer Science and
Electrical and Computer Engineering

Kiri Wagstaff

Kiri Wagstaff
Senior Researcher
NASA Jet Propulsion Laboratory
California Institute of Technology

Houston, Texas
April, 2013

ABSTRACT

Adaptive Similarity Measures for Material Identification in Hyperspectral Imagery

by

Brian D. Bue

Remotely-sensed hyperspectral imagery has become one the most advanced tools for analyzing the processes that shape the Earth and other planets. Effective, rapid analysis of high-volume, high-dimensional hyperspectral image data sets demands efficient, automated techniques to identify signatures of known materials in such imagery. In this thesis, we develop a framework for automatic material identification in hyperspectral imagery using *adaptive* similarity measures. We frame the material identification problem as a multiclass similarity-based classification problem, where our goal is to predict material labels for unlabeled *target* spectra based upon their similarities to *source* spectra with known material labels. As differences in capture conditions affect the spectral representations of materials, we divide the material identification problem into *intra-domain* (i.e., source and target spectra captured under identical conditions) and *inter-domain* (i.e., source and target spectra captured under different conditions) settings.

The first component of this thesis develops adaptive similarity measures for intra-domain settings that measure the relevance of spectral features to the given classification task using small amounts of labeled data. We propose a technique based on multiclass Linear Discriminant Analysis (LDA) that combines several distinct similarity measures into a single *hybrid* measure capturing the strengths of each of the individual measures. We also provide a comparative survey of techniques for *low-rank* Mahalanobis metric learning, and demonstrate that regularized LDA yields competitive results to the state-of-the-art, at substantially lower computational cost.

The second component of this thesis shifts the focus to inter-domain settings, and proposes a multiclass *domain adaptation* framework that reconciles systematic differences between spectra captured under similar, but not identical, conditions. Our framework computes a similarity-based mapping that captures structured, relative relationships between classes shared between source and target domains, allowing us apply a classifier trained using labeled source spectra to classify target spectra. We demonstrate improved domain adaptation accuracy in comparison to recently-proposed multitask learning and manifold alignment techniques in several case studies involving state-of-the-art synthetic and real-world hyperspectral imagery.

Acknowledgments

As I am finally finishing the last few page of my thesis, I would like to acknowledge the following people who, in one way or another, influenced or supported me. This thesis would not have been possible without you.

Most of all I am indebted to my advisor, Erzsébet Merényi. She gave me the freedom to blaze my own path while always looking out for my best interests.Without her guidance, patience and friendship, this thesis never would have been completed, and I will continue to strive to meet her high standards.

I would also like to thank my committee for their valuable suggestions and contributions: to Devika for her constant support and enthusiasm, and for being one of the best mentors and teachers I have ever had; to Kiri, for her tireless assistance with a myriad of topics both at JPL and during graduate school, and for remaining such an encouraging collaborator over the years; and to Chris, for providing the inspiration and new perspective that gave me the last crucial components of my thesis. Their wisdom was vital to the progress of my thesis and I look forward to our future collaborations.

I owe a deep debt of gratitude to Tom Stepinski. Tom introduced me to machine learning and has been a constant source of inspiration over the years. He taught me much of what I know about the research method by his example, and my early collaborations with him have provided me with more insight and opportunities than I had imagined possible.

I have been extremely fortunate to work with a spectacular group of energetic and fun individuals at JPL. David Thompson has been one of my most steadfast collaborators in recent years, and is one of the most spectacularly inventive individuals I have ever known. Our discussions on metric learning and domain adaptation played a substantial role in the development of this thesis. Lukas Mandrake has been a valuable

sounding board for ideas, and his insight and humor have been much appreciated over the years. Ben Bornstein contributed a great deal to my growth as a software developer and both his dedication and knowledge about a multitude of topics have been particularly inspirational. I also want to thank Steve Chein, Mike Burl, Seungwon Lee and Benyang Tang for their input and contributions on a number of projects at JPL along the way. Finally, I owe many thanks to Becky Castaño, who provided me my first opportunity to work at JPL, and for her guidance during my first few years there; and to Rob Granat for his invaluable advice during graduate school, and for helping me stay on track to return to JPL after finishing.

I am grateful to both Karen Sutherland and Noel Petit during my undergraduate studies at Augsburg. Karen gave me my first taste of computer vision research, and my earliest work with her had a tremendous influence on my decision to pursue a ML/AI-related research career. I also owe Noel Petit a great deal for introducing me to space science research, and for showing me that computer science can have a broad impact on interdisciplinary topics. Without their enthusiasm and encouragement, I would have never started down this path, and my choice to follow it to JPL is a tribute to their efforts.

Before Augsburg, my teachers Ellen Siewert and Dave Saterbak believed in me at a time where I didn't believe I was capable of bigger and better things. The world needs more teachers like them.

The current and former members of my research group at Rice also played a role in the development of this work – to Patrick O'Driscoll for always being a friendly cohort in the lab and for our entertaining discussions; to Lily Zhang for her input on coursework and research topics and for her assistance with software development; and to Mike Mendenhall for his assistance with the software for our group, and for his input on our WHISPERS papers.

I also would like to thank my collaborators for their assistance and input – to James Killian for his invaluable time and effort with our EMG research; to Keith Cooper and Linda Torczon for our discussions regarding the PACE project; to Bea Csathó for her assistance with the Ocean City imagery; to John Kerekes and David Pogorzala at RIT for their assistance the DIRSIG imagery; to Dar Roberts for his assistance with our initial material identification efforts and for providing the urban spectral library; to Martha Gilmore for her expertise on the CRISM and Cuprite imagery; and to Bill Farrand, for our discussions on atmospheric calibration and other topics related to remote-sensing.

Several institutions have contributed financially to support this work. The first three years of my time at Rice was supported by the NASA Graduate Student Researchers Program (GSRP, grant number NNX07AR79H). The NASA Advanced Multi-Mission Operations System Multimission Ground Systems and Services (AMMOS-MGSS) office, the DARPA Platform-Aware Computing Environment (PACE), and also the NSF Vertical Integration of Research and Education (VIGRE) also provided support during my last few years at Rice. I also want to express my thanks to the Lunar and Planetary Institute for their support as a summer intern and later as a visiting graduate student prior to my arrival at Rice.

My friends both in and outside of work and school have helped me maintain my sanity (such as it is) over the years. Thanks to Becky Shaknovich, Shaun Perlow, Dav Johnson, Katie Breslin, Heather Dalton, Jason Laska, Angie Hanson, Camille Plantiveau, Eric Berglund, Jon and Jenny Zoss, Bernice Ye, Mel Mickle, Maggie Glasscoe, Ian Molloy, Kelly Cannon, Chrissie and Katie Clover, Nikki and Travis Halverson, Scott Novich, Eva Dyer, Manjari Narayan, Andrew Waters, Tim and Emily Stough, Kacie Shelton, Chandra Barnett, Jingshu Huang, Tasos Giannoulis, Ben and Heather Jackson, Dan Justice, and many more. Thank you all for being there and for always reminding me that the world is a far, far better place than can be seen from the perspective of a graduate student.

Finally, I owe a great debt of gratitude to my parents, my sister and my girlfriend Elina for their constant support and encouragement, for sharing in my successes and defeats, and for always being there when I needed them.

Contents

List of Illustrations

List of Algorithms

List of Symbols

Scalars, Vectors and Matrices

x	Scalar $\in \mathbb{R}$
$\mathbf{x} = (\mathbf{x})_i = x_i$	Vector $\in \mathbb{R}^n$ with entries x_i
$\mathbf{x}^T$	Transpose of vector $\mathbf{x}$
$(\mathbf{x}^D)^T$	Transpose of vector $\mathbf{x}^D$ from domain $D \in \{S, T\}$
$\mathbf{1}^n$	Vector of n ones in $\mathbb{R}^n$
$\mathbf{0}^n$	Vector of n zeros in $\mathbb{R}^n$
$\mathbf{M} = (\mathbf{M})_{ij} = a_{ij}$	Matrix $\in \mathbb{R}^{n \times m}$ with entries a_{ij}
$\mathbf{M}^T$	Transpose of matrix $\mathbf{M}$
$(\mathbf{M}^D)^T$	Transpose of matrix $\mathbf{M}^D$ from domain $D \in \{S, T\}$
$\mathbf{M}_{\cdot j}$	jth column of matrix $\mathbf{M}$
$\mathbf{M}_{i.}$	ith row of matrix $\mathbf{M}$
$\mathbf{M}^p$	p^{th} power of matrix $\mathbf{M}$
$(\mathbf{M}^D)^p$	p^{th} power of matrix $\mathbf{M}^D$ from domain $D \in \{S, T\}$
$\mathbf{I}^n$	n-dimensional identity matrix

Constants

n, m	Feature space dimensions
K	Number of classes
N	Number of training samples
N^D	Number of samples in domain $D \in \{S, T\}$
N_j	Number of training samples from the j^{th} class
N_j^D	Number of samples from the j^{th} class in domain $D \in \{S, T\}$
Q	Number of pivot samples
Q_k	Number of pivot samples from the k^{th} class

Probability Spaces

$\mathcal{X}$	Sample/Feature space
$\mathcal{Y}$	Label space
$\mathcal{H}$	Hypothesis space

Samples/Labels

X	Subset of samples drawn from $\mathcal{X}$
Y	Labels $y \in Y$ for each sample $\mathbf{x} \in X$
y	Class label $\in \{1, \dots, K\}$
X^{tr}	Set of training samples $\mathbf{x}^{tr} \in X^{tr}$
X_j^{tr}	Set of training samples from the j^{th} class
X^{te}	Set of test samples $\mathbf{x}^{te} \in X^{te}$
X_j^{te}	Set of test samples from the j^{th} class
X^D	Set of samples $\mathbf{x}^D \in X^D$ from domain $D \in \{S, T\}$
X_j^D	Set of samples of the j^{th} class from domain $D \in \{S, T\}$
Y^D	Labels $y^D \in Y^D$ for each sample $\mathbf{x}^D \in X^D$ from domain $D \in \{S, T\}$
P^D	Set of pivot samples $\mathbf{p}^D \in P^D$ from domain $D \in \{S, T\}$
P_j^D	Set of pivot samples of the j^{th} class from domain $D \in \{S, T\}$
Y^P	Set of pivot labels for each of the paired pivot samples $(P^S, P^T, Y^P) = (\mathbf{p}_i^S, \mathbf{p}_i^T, y_i^Q)_{i=1}^Q$

Similarity/Distance Measures

$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{d}_{ij}$	Distance or dissimilarity between samples $\mathbf{x}_i$ and $\mathbf{x}_j$
$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{k}_{ij}$	Similarity between samples $\mathbf{x}_i$ and $\mathbf{x}_j$
$\mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{d}_{ij}^{(l)}$	Distance between the l^{th} derivatives of samples $\mathbf{x}_i$ and $\mathbf{x}_j$

Functions

$\langle \mathbf{x}_i, \mathbf{x}_j \rangle$	Dot product between vectors $\mathbf{x}_i$ and $\mathbf{x}_j$
$\lVert \mathbf{x} \rVert$	L^2-norm of vector $\mathbf{x}$

$\|\mathbf{x}\|_p$	L^p-norm of vector $\mathbf{x}$
$f^{(i)}$	i^{th} derivative of continuous function f
$\mathrm{I}(\cdot)$	Indicator function, returns 1 when argument true, 0 otherwise
$\mathrm{L}(\hat{y}, y)$	Loss function, returns nonzero value when $\hat{y} \neq y$
$\mathrm{diag}(\mathbf{M})$	Diagonal vector of $n \times n$ matrix $\mathbf{M}$
$\mathrm{diag}(\mathbf{x})$	$n \times n$ matrix with the entries of $\mathbf{x}$ on the diagonal
$\mathrm{tr}(\mathbf{M})$	Trace of matrix $\mathbf{M}$

Abbreviations

μm	Microns
FWHM	Full Width at Half Maximum
kNN	k-Nearest Neighbors
ANN	Artificial Neural Network
G(R)LVQ	Generalized (Relevance) Learning Vector Quantization
LDA	Linear Discriminant Analysis
MTL	Multi-task Learning
PCA	Principal Component Analysis
SVD	Singular Value Decomposition
SVM	Support Vector Machine

Greek Symbols

$(\boldsymbol{\psi}_i, \lambda_i)$	i^{th} largest eigenvector with eigenvalue λ_i
$\boldsymbol{\alpha} = \alpha_i$	Weight coefficient vector with entries α_i
$\boldsymbol{\gamma} = \gamma_i$	Vector of regularization parameters with entries γ_i
κ	Number of derivatives
$\boldsymbol{\mu} = \mu_i$	Mean vector in $\mathbb{R}^n$ with entries μ_i

To Elina: for making this half-empty glass half-full.

Introduction

Material Identification in Hyperspectral Imagery

Analysis of remotely-sensed imagery has greatly improved our understanding of the geologic and climactic processes that shape our planet and beyond. On Earth, remotely-sensed imagery has provided a wealth of information for numerous applications including natural resource allocation [Abrams et al., 1977; Merényi et al., 2000], global climate change monitoring [King et al., 1995], urban planning [Herold et al., 2004] and mineralogical surveys [Rowan et al., 2003]. Remote sensing also enables data collection in dangerous or inaccessible areas and has been used to observe active volcanoes [Davies et al., 2006], to monitor the effects of the Chernobyl nuclear accident [Sadowski and Covington, 1988], and, more recently, to inform disaster recovery efforts from the Red Mud spill in Kolontar, Hungary [Lenart et al., 2011]. Additionally, planetary scientists have made extensive use of remotely-sensed imagery to characterize the mineralogical composition of planets in our solar system [Gilmore et al., 2007; Merényi et al., 1996; Pelkey et al., 2007] and other celestial bodies such as asteroids [Howell et al., 1994]. Identification and characterization of materials from imagery is a fundamental component of all of these applications.

Air and spaceborne *hyperspectral* sensors are a powerful enabling technology for remote material identification. Much in the same way as the human eye sees visible light, a hyperspectral sensor captures the interaction of electromagnetic (EM) radiation with materials over tens-to-hundreds of contiguous, narrowly-spaced bandpasses, including ultraviolet (UV) and infrared (IR) wavelengths outside the range of human vision. Figure 1 demonstrates the concept of a hyperspectral image. Each *pixel* (*spectrum* or *spectral signature*) in a remotely-sensed hyperspectral image is a high-dimensional vector that characterizes the material(s) at a particular geographic location. Each entry in a given spectral signature is determined by the interaction of EM radiation at a specific wavelength with the materials it represents. Thus, by comparing hyperspectral image pixels to spectral signatures of known materials, scientists can determine which materials the pixels represent [Adams and Gillespie, 2006]. However, the sheer volume of hyperspectral imagery captured today precludes such comprehensive manual inspection. For instance, the hyperspectral Compact Reconnaissance Imaging Spectrometer for Mars (CRISM, [Murchie et al., 2007]) onboard the Mars Reconnaissance Orbiter (MRO) spacecraft will return over 4 terabytes of imagery to Earth over the duration of its mission. Rapid exploitation of remotely-sensed hyperspectral imagery demands automated techniques that can identify and summarize the most scientifically interesting spectral signatures.

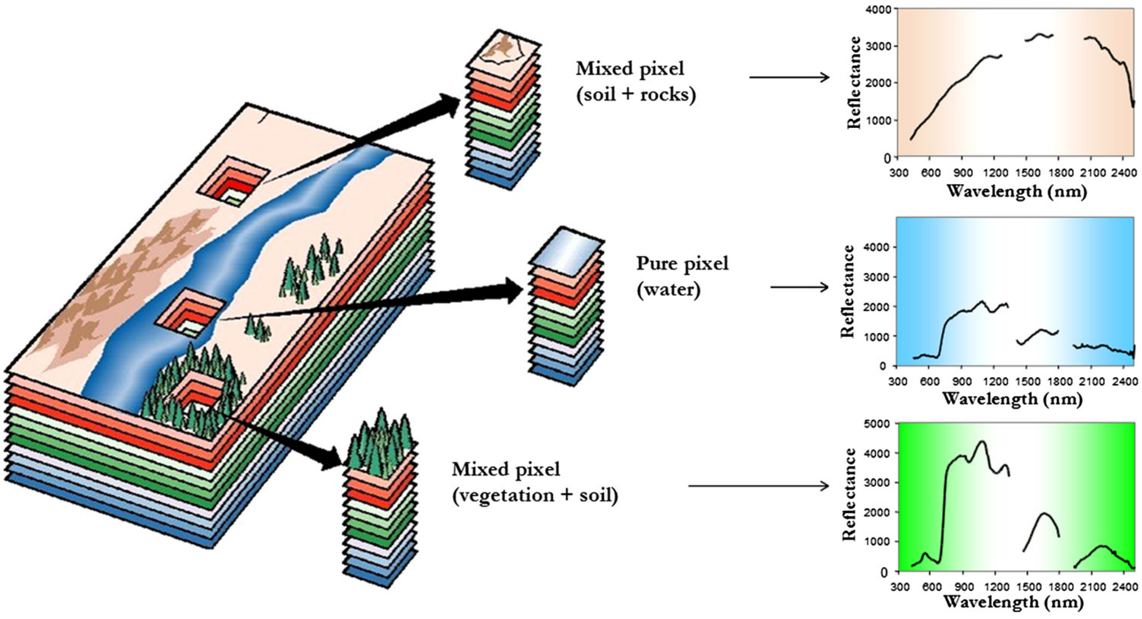


Figure 1 : The concept of hyperspectral imaging. Each pixel characterizes the material(s) it represents. Figure from Ciznicki et al. [2012], printed with permission.

As hyperspectral imagery became more readily available within recent decades, a number of automated approaches to address the material identification problem have been proposed. Among the earliest and most successful automated material identification systems for hyperspectral imagery is the USGS Tetracorder system [Clark et al., 2003; Swayze et al., 1999]. Tetracorder is built upon an extensive library of material signatures, each characterized in terms of diagnostic wavelengths. By measuring the similarity between the reflectance values of the library signatures to the reflectances of an observed spectrum for diagnostic wavelengths, Tetracorder can quite effectively identify a wide range of terrestrial materials [Clark et al., 2003]. However, a number of practical issues limit the applicability of Tetracorder for rapid exploitation of large hyperspectral image data sets. In particular, Tetracorder requires expert input to maximize material identification results [Gilmore et al., 2008], and must also be modified by an expert to match a particular imaging system [Rauss et al., 2000]. Adding new materials to the Tetracorder spectral library requires considerable spectroscopic expertise, as each new material must be characterized according to its diagnostic absorption features.

A more data-centric approach to automated material identification was pioneered by Landgrebe et al. [Landgrebe, 1968; Lee and Landgrebe, 1993]. By building upon established pattern recognition and image analysis techniques for panchromatic or color (i.e., 1-4 spectral bands) imagery, and for *multispectral* (i.e., 10-20 bands) imagery, they developed a methodology for hyperspectral image classification that is still widely used today. In hyperspectral image classification, the goal is to construct statistical models of a set of labeled pixels representing known material or land cover classes such as water, soil or vegetation, and then apply the resulting models to predict class labels for unlabeled pixels. If the labeled classes consist of a set of distinct material species, we can apply hyperspectral image classification techniques to identify materials based upon their spectral signatures.

Many of the early hyperspectral image classification techniques, such as those developed by Landgrebe et al., are based on simple statistical models such as Maximum Likelihood classification. Such techniques perform reasonably well for imagery of low spectral dimensionality and containing few classes of interest, but are often inaccurate for classifying the numerous, and in many cases spectrally similar, material classes present in remotely-sensed imagery. However, new developments within the past 30 years have shown considerable progress in meeting the challenges posed by hyperspectral image classification. Sophisticated classifiers such as Artificial Neural Networks (ANNs [Merényi, 1998; Villmann et al., 2003]) and Support Vector Machines (SVMs [Camps-Valls and Bruzzone, 2005; Gualtieri and Cromp, 1999]) have demonstrated the capability to learn complex and often nonlinear relationships between training classes for applications such as monitoring ecosystem resources [Merényi et al., 2000], analyzing the spread of invasive species [Ustin et al., 2002], and investigating terrestrial analogues for planetary exploration [Gleeson et al., 2010]. Additionally, such classification techniques have been deployed onboard intelligent spacecraft systems, such as the Earth Observing-1 (EO-1) Autonomous Sciencecraft Experiment [Chien et al., 2005], to prioritize scientifically important observations for immediate transmission to Earth and for autonomously scheduling supplementary data collection when interesting science events occur [Bornstein et al., 2011; Thompson et al., 2012].

An outstanding issue in classifying high-dimensional hyperspectral signatures is developing methods to determine which spectral features are discriminative for the materials of interest in each study. By finding a good feature representation that emphasizes distinctions between spectral classes, we can improve both the classification accuracy and the efficiency of hyperspectral image classification algorithms [Keshava, 2004]. However, precisely which spectral features are discriminative depends on the nature of the materials of interest – no “global” set of features exists that are discriminative in all contexts. For example, vegetation can often be characterized by broad, slowly varying spectral features, while other materials, such as minerals and gases, can possess very narrow spectral features [Shaw and Burke, 2003]. A number of *feature selection* techniques have been proposed to determine which spectral features are most relevant to a given classification task [Benediktsson et al., 1995; Berg and Jensen, 2007; Kuo and Landgrebe, 2001; Landgrebe, 1997; Mendenhall, 2006]. Robust feature selection can simplify a challenging classification problem by mapping the original feature space to a typically low(er)-dimensional feature space where class distinctions are emphasized. In this new feature space, we can potentially estimate class parameters with fewer samples than in the original feature space, or we can apply a simple classifier in the new space with results comparable to using a more sophisticated classifier. However, as stated by Guyon and Elisseeff [2003], and Mendenhall and Merényi [2008], many feature selection techniques fail to capture the most relevant information to distinguish between classes, either by operating independently of class labels, or by optimizing criteria that is often uninformative for classification. Moreover, many traditional feature selection techniques assume that labeled and unlabeled pixels are drawn from the same probability distribution or, alternatively, reside in the same feature space. Such techniques are inadequate for classifying spectral signatures when the capture conditions of the labeled and unlabeled pixels differ. These issues demand techniques that optimize classification accuracy to learn good feature representations for high-dimensional data, while being robust to differences between feature spaces.

The problem of finding a good feature representation for a given classification task is closely related to the problem of finding an accurate *similarity measure* to compare samples [Balcan et al., 2006; Balcan et al., 2008a]. Much in the same way that a defining a good set of features emphasizes important class distinctions, a good similarity measure provides contrast to distinguish between samples from different classes [Hertz, 2006]. To illustrate this connection, consider the case when our features consist of the correct class label for each sample, versus the case where our features consist of random noise. We can expect any principled classification algorithm to trivially solve the classification problem posed in the first scenario. In contrast, we cannot expect any classifier to predict labels with accuracy better than random guessing in the second scenario. Analogously, a similarity measure that indicates samples from the same class are near one another while indicating samples in different classes are far apart is far more useful for predicting class labels than a measure that does not discriminate between samples from the same or different classes.

Most hyperspectral image classification algorithms rely upon *unweighted* similarity measures such as the Euclidean Distance or cosine similarity, or various *application-specific* measures that are hand-designed to incorporate domain specific knowledge [Clark et al., 2003, 1990; van der Meer, 2000]. In recent years, there has been a growing body of work in the field of *metric learning* to develop *adaptive* similarity measures that adjust to a given classification task [Alipanahi et al., 2008; Davis et al., 2007; Weinberger et al., 2006], which can be viewed as an alternative to conventional feature selection techniques. Once learned, adaptive similarity measures can be "plugged in" to algorithms relying on distance computations, potentially improving classification accuracy while reducing training and prediction time. Additionally, adaptive similarity measures can potentially be used to reconcile differences between spectra captured under different conditions, allowing the incorporation of multi-source image data in classification.

Overview and Contributions

This thesis develops adaptive similarity measures designed for identifying materials from hyperspectral signatures. Our goal is to predict material labels for a set of unlabeled *target* signatures according to their relationships to a set of *source* signatures with known material labels. We frame the material identification problem as a *similarity-based classification* problem, where our predictions are based upon a pairwise similarity measure that quantifies the relationships between the target spectra and the labeled source spectra or statistical models derived from the source spectra. We demonstrate that augmenting existing similarity-based classification algorithms with adaptive, task-specific similarity measures improves classification accuracy and also often reduces computation time, thereby improving our capabilities for rapid exploitation of high-dimensional, hyperspectral imagery. Additionally, we provide a novel framework that uses a specific form of adaptive similarity measure to reconcile differences between spectral signatures captured under different conditions (e.g., by different sensors, at different capture times, or at different spatial locations). This framework allows us to incorporate data from multiple image sources in classification, thereby mitigating issues with small training sets which commonly occur when classifying hyperspectral imagery.

For clarity, these contributions are organized into three parts. Part I gives an overview of the fundamental concepts of hyperspectral imaging and material identification we consider in this work, and the challenges involved in automatic material identification. We begin in Chapter 1 with a description of the automated material identification problem, starting from the basic concepts involved in interpreting remotely-sensed hyperspectral imagery, followed by a formal definition of the material identification problem as a similarity-based classification problem. In Chapter 2 we assess the capabilities of several canonical spectral similarity measures for material identification using a technique known in the remote-sensing community as *spectral matching*. This evaluation provides insight into the spectral material identification problem and allows us to identify circumstances where employing adaptive similarity measures is desirable. We demonstrate that a similarity measure that captures the shape of the spectral signature with particular emphasis on the positions of discriminative spectral features that characterize the composition of materials can significantly improve material identification accuracy.

Our investigation of adaptive similarity measures for material identification begins in Part II. Here, we consider methods for *intra-domain* material identification, where source and target spectra are captured under identical conditions. We start with an investigation of the problem of *hybrid* metric learning in Chapter 3. We propose a technique that uses multiclass LDA to combine several distinct similarity measures into a single hybrid similarity measure that captures the strengths of the individual measures with respect to a given material identification task. We propose two novel hybrid measures: the Continuum-Intact/Continuum-Removed measure, and the adaptive Sobolev measure, and demonstrate improved classification accuracy using our hybrid measures in comparison to the Euclidean distance and several conventional feature selection techniques.

In Chapter 4 we consider the technique of Mahalanobis metric learning, which assigns weights to individual spectral features according to their task-specific relevances. We begin with a survey of state-of-the-art Mahalanobis metric learning techniques applied to hyperspectral image classification problems. We show empirically on several hyperspectral data sets that Mahalanobis metrics computed using regularized multiclass Linear Discriminant Analysis (LDA) often achieves better, more stable results than several state-of-the-art Mahalanobis metric learning techniques. We then investigate the application of Mahalanobis metric learning techniques to hyperspectral image segmentation tasks. We show that a learned Mahalanobis metric not only increases separation between known image classes, but also suppresses noisy bands, thereby improving the fidelity of image segments for subsequent analyses.

Part III shifts the focus to *inter-domain* material identification scenarios. We introduce our novel domain adaptation algorithm, *RelTrans*, in Section 5.1, which allows us to use labeled source spectra to classify target spectra captured under similar, but not identical, conditions. In Chapter 5 we provide an analysis and evaluation of RelTrans for supervised domain adaptation problems, where a small number of labeled target spectra are available to construct a mapping between the source and target feature spaces. Then in Chapter 6, we extend RelTrans to automatically construct a mapping from the source to the target feature space using only labeled source spectra and unlabeled target spectra. We conclude Part III with a review of functional and band-weighted extensions for both supervised and unsupervised domain adaptation.

We conclude with a summary of our findings and directions for future research in Chapter 7.

This thesis is the culmination of a variety of intensive collaborations. Where appropriate, the first page of each chapter provides a footnote listing primary collaborators, who share credit for this work.

Part I

Hyperspectral Imaging and Material Identification

Chapter 1

Automated Spectral Identification of Materials

1.1 Material Identification from Spectral Signatures

This chapter reviews the fundamental concepts that make material identification from spectra possible, and the challenges involved in applying material identification techniques to hyperspectral data. We begin in Section 1.2 with a description of how we interpret the measurements captured by a spectral sensor. Section 1.3 then describes issues specific to classifying hyperspectral data, and summarizes which issues we address in this work, and Section 1.4 reviews the similarity-based classification algorithms we apply to the material identification problem. However, our central focus is not on defining new classification algorithms, but rather on developing similarity measures that perform well for classifying high-dimensional, hyperspectral data independent of the classification algorithm. Thus, Section 1.5 then reviews the canonical spectral similarity measures we evaluate, and Section 1.6 briefly summarizes the types of adaptive similarity measures that we develop in this thesis.

1.2 Interpretation of Spectral Measurements

Hyperspectral sensors are a type of *spectral imaging system*. Spectral imaging systems measure variations in the way materials respond to electromagnetic (EM) radiation. EM radiation is a form of energy manifested as energy waves of varying lengths (*wavelengths*). When EM radiation reaches surface materials, a number of interactions may occur – the energy may be reflected off of the surface, it may be absorbed by the surface, or it may be transmitted through the surface. The nature of the interaction depends on the type of material, the wavelength, and the environmental conditions under which the measurements are captured. The physics behind these interactions are well-understood (see, e.g., the seminal works of Hunt and Salisbury on the spectroscopic properties of rocks and mineral identification [Hunt and Salisbury, 1970, 1971, 1976a,b; Hunt et al., 1971a, 1973a, 1971b, 1974, 1972, 1973b,c; Salisbury et al., 1975]). Based upon the amount of light reflected in each bandpass, we can derive the molecular composition and some physical properties of materials from the measurements captured by a spectral sensor. We refer to the measured response of the sensor at a particular wavelength as a *spectral feature*. The set of spectral features ordered by increasing wavelength at a particular geographic location is called a *pixel*, *spectrum* or *spectral signature*. We refer to the integer-valued index of a specific wavelength in a hyperspectral image or a given spectral signature as a *spectral band*.

We interpret each spectral signature as having two components: *absorption features* and *continuum*. Figure 1.1 illustrates the difference between these two components. Absorption features (or, simply, *absorptions*) are contiguous wavelengths of a spectral signature where EM radiation is absorbed. Absorptions only occur at wavelengths where EM radiation resonates with energy needed to trigger certain electronic or vibrational processes related to particular materials [Adams and Gillespie, 2006]. The positions and widths of absorption features reveal information about the material composition, and the depths of the absorptions indicate the concentration of the material(s) the spectrum represents. The continuum of a spectral signature is the "background absorption" onto which absorption features are superimposed [Clark, 1999].

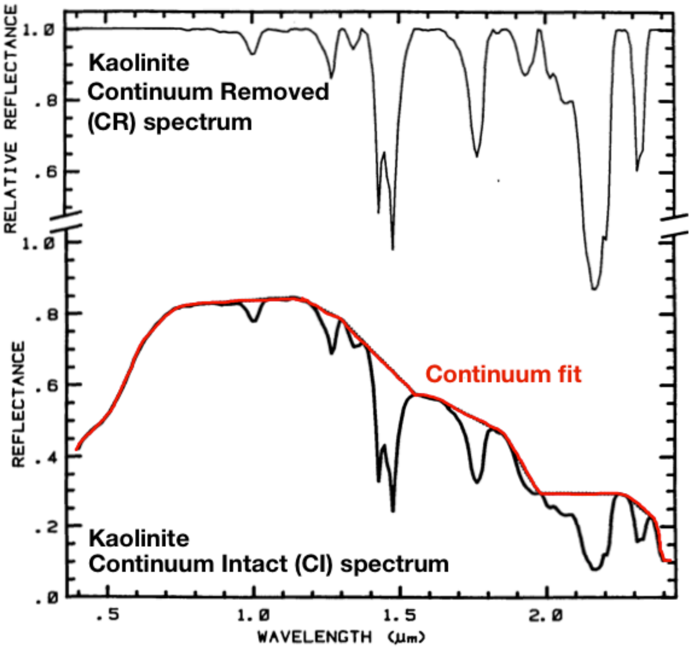


Figure 1.1 : Continuum-intact (CI) spectral signature for Kaolinite (bottom, black line), continuum fit (bottom, red line) and continuum-removed (CR) absorption features (top, black line). The widths and positions of absorption features characterize the material composition of the spectrum. Figure modified from Clark et al. [1987], printed with permission.

1.3 Fundamentals of Hyperspectral Image Classification

1.3.1 Background and Notation

The variability of hyperspectral signatures according to their material compositions make them well-suited as input to an automatic pattern recognition system. Thus, we can view the automated material identification problem an instance of a hyperspectral image classification problem. In hyperspectral image classification, our goal is to infer labels Y^T for a set of M unlabeled *test* spectra X^T, based upon their similarities to a set of labeled *training* spectra $(X^{tr}, Y^{tr}) = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ representing K discrete classes (e.g., distinct materials) with labels $y_i \in \{1, \ldots, K\}$. To fix notation: We use the symbol $\mathcal{X}$ to refer to the n-dimensional feature space in which the samples reside, and the symbol $\mathcal{Y}$ to refer to the label space. We initially assume that the training and the test sets are drawn i.i.d. from the same joint probability distribution $\mathrm{p}(\mathcal{X}, \mathcal{Y})$, and refer to such classification problems as an *intra-domain* problems. Later in this work, we consider problems where the training set are drawn from *source distribution* $\mathrm{p}^S(\mathcal{X}, \mathcal{Y})$, and the test samples are drawn from a similar, but not identical *target distribution* $\mathrm{p}^T(\mathcal{X}, \mathcal{Y})$. We refer to such classification problems as *inter-domain* classification problems, and use notation (X^S, Y^S) and (X^T, Y^T) to refer to the training and test samples, respectively. Consequently, we use the terms *source samples* and *target samples* interchangeably with *training samples* and *test samples*, respectively.

1.3.2 Challenges of Hyperspectral Image Classification

Automated material identification in hyperspectral imagery faces several major theoretical and computational challenges. A detailed survey of these challenges is given by Merényi in [Merényi, 1998]. In this section, we summarize these challenges.

Spectral and Spatial Resolution To distinguish between spectrally similar but distinct materials, a spectral imaging sensor must capture spectra at sufficient spectral resolution [Adams et al., 1989]. The high spectral resolution of hyperspectral sensors comes increased expectations to identify a wider range of materials than with low spectral resolution panchromatic or multispectral sensors. Such sensors often capture signatures with insufficient spectral resolution to distinguish between similar materials, and analyses using such data are intrinsically limited by the spectral resolution of the sensor [Goetz et al., 1985]. Hyperspectral sensors, on the other hand, can capture fine-grained differences between spectrally similar, but compositionally unique materials [Clark and Rousch, 1984; Goetz et al., 1985; Merényi, 2000].Figure 1.2 illustrates the difference in spectral resolution between a lab-measured spectral signature of the material kaolinte (bottom) vs. the signatures of the same material captured by the multispectral Landsat TM sensor (top) and the hyperspectral AVIRIS

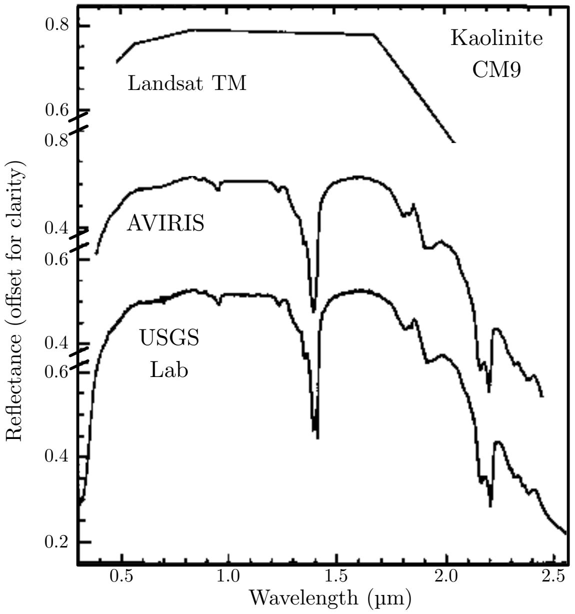


Figure 1.2 : Comparison of Landsat TM (top), AVIRIS (middle) and lab-measured (bottom) spectra. Figure modified after Swayze [1997], printed with permission.

sensor (middle). The low spectral resolution of the multispectral sensor often does not resolve distinct materials distinguished by narrow-band spectral features [Vane and Goetz, 1993]. The increased spectral resolution of hyperspectral imagery not only allows a wider range of materials to be discriminated, but also enables the analysis of materials in terms of their intrinsic properties such as temperature [Roush and Singer, 1986] and grain size [Ross et al., 1969].

While modern hyperspectral imaging systems benefit from high spectral resolution, their high spectral resolution typically comes at the cost of lower spatial resolution – usually on the meter to tens-of-meters / pixel scale. The spatial resolution of the sensor also affects our capability to distinguish between materials, as low spatial-resolution sensors often capture pixels that represent mixtures of several distinct, spatially adjacent materials [Keshava and Mustard, 2002]. The presence of mixed spectra greatly complicates precise material identification, as the properties of the materials present in each pixel, along with the proportions in which they occur, can mask or distort characteristic spectral features of the individual materials [Adams et al., 1989]. Resolving materials on the sub-pixel scale is a difficult problem since it requires advance knowledge of which materials may contribute to a given signature, their concentrations within the signature, and the types of physical interactions which can occur between the materials in the mixture.

Atmospheric Contamination The measurements of EM radiation collected by a spectral imaging system are modified by scattering and absorption by gases and aerosols while traveling through the atmosphere from the Earth's surface to the sensor [Adams and Gillespie, 2006; Schott, 2007]. In the context of material identification, these atmospheric interactions are usually viewed as contamination, and are typically resolved by converting the *radiance* measurements observed at the sensor to *surface* *reflectance*. Surface reflectance is a relative measure of how much a material departs from being a "perfect" reflector. To retrieve the surface reflectance from remotely-sensed radiance spectra, it is necessary to apply *radiometric calibration* techniques. Radiometric calibration consists of two components: sensor calibration and atmospheric calibration. Sensor calibration normalizes the responses captured by the sensor to those of a standard light source, and validates the integrity of the observed spectra according to the effective radiance reaching the sensor [Schott, 2007]. Atmospheric calibration converts the sensor-calibrated radiances to surface reflectance by estimating atmospheric parameters either directly from atmospheric measurements, or from ground measurements of surface materials. After performing radiometric calibration, spectral data are mapped to the same relative radiometric scale, allowing us to potentially compare spectral signatures captured under different conditions, or by different sensors. However, as the interactions caused by atmospheric scattering and absorption are too complex to completely characterize, radiometrically calibrated spectra are still only an approximation of the true reflectances. Additionally, some spectral bands are unrecoverable even after calibration. For instance, in terrestrial imagery, spectral bands in the $[1.3, 1.5]$ and $[1.7, 2.0]$ μm range are typically saturated due to water vapor absorptions. Such noisy bands are generally removed, as they are too noisy to provide useful discriminating information for most applications. Unless otherwise indicated, we assume the hyperspectral data we analyze in this document are reflectance spectra, and we will specify the range of wavelengths in each image we examine. Additionally, to account for linear scaling factors caused by varying illumination conditions, we scale pixels signatures by their L^2 norm.

Dimensionality of Spectra, Quantity and Quality of Labels The lack of exhaustive and detailed ground-truth labels for large-scale remote-sensing surveys also makes objective evaluation of spectral material identification challenging. Exhaustive labels are not available due to the fact that obtaining labeled data for large-scale remote-sensing surveys is expensive, time-consuming, and in some cases (e.g., planetary missions) impossible. As a consequence, labels are often defined by human experts via photogeologic interpretation of imagery, which is tedious, error-prone, and subjective. Thus, the available labeled data is not only in limited quantity, but spectra may be mislabeled. These issues make classifying hyperspectral data particularly challenging using conventional techniques due to the well-known *Hughes phenomenon* [Hughes, 1968]. Specifically, the Hughes phenomenon occurs when the number of feature dimensions outnumbers the number of available samples per class. In such cases, conventional classification techniques often unreliably capture those poorly-represented classes. Developing robust automated material identification techniques that are robust to limited quantities of training samples is one of the main focuses of this thesis.

Another major issue is that labels are often provided on the *object*, rather than the *material*, level. We define an object as a collection of one or more materials collectively described as a high-level semantic concept. This distinction is crucial, as determining the object to which a spectral signature belongs is often impossible without additional context. For instance, an asphalt rooftop and an asphalt road cannot be differentiated by their spectral signatures alone, as their signatures only reflect their material composition (i.e., asphalt), and not the objects (i.e., rooftop and road) composed of that (and possibly other) material(s). Conversely, determining the material compositions of a spectral signature given only an object label also requires additional context. For example, it is possible to determine if spectral signatures labeled “rooftop” and “road” represent similar materials, but it is not possible to automatically determine their materials given only their object labels. Consequently, unless otherwise specified, we assume that material, rather than object, labels are provided for all labeled spectral signatures.

Spectral Sensitivity to Capture Conditions Incorporating labeled data from previous analyses of similar imagery can potentially be a great resource in mitigating the paucity of labeled data which commonly occurs when classifying hyperspectral imagery. However, in spectral material identification, we typically assume that the source and target spectra are captured under the same conditions. In such scenarios, we can justifiably assume that the source and target spectra are drawn from the same joint probability distribution, i.e., the classification problem is an intra-domain problem. However, the spectral representations of identical materials varies across sensors, geospatial regions, or under different environmental conditions, and we must reconcile differences between spectra captured under different conditions in order to incorporate them in classification tasks. In other words, when source and test spectra are captured under different conditions, the assumption that the source and test spectra are both drawn from the same joint distribution does not hold. When the source and target distributions are similar, we can apply *domain adaptation* techniques to reconcile differences between the source and target distributions. In such scenarios, robust domain adaptation allow us to increase the effective number of samples available to train a classifier, potentially increasing classification accuracy and allowing us to classify a wider range of classes than with the available intra-domain training data. However, when the source and target distributions differ significantly, we cannot expect a classifier trained on the source data to yield performance better than random guessing on test data that is irrelevant to the source data. For example, a classifier trained on spectra of man-made materials such as concrete and asphalt would likely generate inaccurate predictions for spectra representing different types of vegetation. We discuss the issues involved in inter-domain material identification in greater detail in Part III.

1.4 Similarity-based Classification Algorithms

We consider *similarity-based* classification techniques in this thesis. A similarity-based classification algorithm predicts the label of a given unlabeled test sample based upon the similarity – or dissimilarity – of that sample vs. a set of labeled training samples. Similarity-based classification techniques are distinguished from *feature-based* classification algorithms in that a similarity-based classifier generates predictions based solely upon the (dis)similarity measurements between samples, and does not require direct access to the features of the test or training samples. Similarity-based classification algorithms are among the oldest and most widely-used pattern recognition techniques [Duda and Hart, 1973], and a number of recent works have investigated their theoretical properties (e.g., [Balcan et al., 2008a; Cazzanti, 2007; Chen et al., 2009; Kar and Jain, 2011; Pekalska, 2005]). In this section, we review the similarity-based classification algorithms we consider in this dissertation.

Nearest Neighbor The most straightforward similarity-based classifier is the *nearest-neighbor* (NN) classifier [Duda and Hart, 1973]. A nearest-neighbor classifier assigns the label y of the nearest training sample $\mathbf{x}'$ to the test sample $\mathbf{x}$.

$$y = \operatorname*{argmin}_{j \in \{1,\ldots,K\}} \left(\min_{\mathbf{x}' \in X_j^{tr}} (\mathrm{d}(\mathbf{x}, \mathbf{x}')) \right) \tag{1.1}$$

where X_j^{tr} is the set of training samples from the j^{th} class. A popular variant of nearest-neighbor is the k *nearest-neighbor* (KNN) classifier, which predicts the label of $\mathbf{x}$ via majority vote from the k nearest training samples.

Minimum Distance The minimum distance (MinDist) classifier assigns the label of the nearest class mean to test sample $\mathbf{x}$

$$y = \underset{j \in \{1,\ldots,K\}}{\operatorname{argmin}} \big(\mathrm{d}(\mathbf{x}, \boldsymbol{\mu}_j)\big) \tag{1.2}$$

where $\boldsymbol{\mu}_j$ is the mean of the training samples in the j^{th} class. When the distance measure d is the Euclidean distance, the MinDist classifier computes the maximum likelihood estimate that $\mathbf{x}$ is an instance a multivariate Gaussian centered at $\boldsymbol{\mu}_j$ with unit covariance [van Otterloo and Young, 1978]. Despite its simplicity, the MinDist classifier often performs surprisingly well for hyperspectral image classification tasks, often yielding competitive results with significantly more sophisticated classification algorithms [Merényi et al., 2011].

Artificial Neural Networks Artificial Neural Networks (ANNs, [Rumelhart et al., 1986]) are a sophisticated suite of techniques that have produced state-of-the-art hyperspectral image classification results [Merényi, 1998; Merényi et al., 2011]. An ANN is a finely-distributed, massively-parallel learning machine that emulates the information processing of a biological nervous system. While numerous types of neural networks have been proposed (e.g., [Ackley et al., 1985; Hopfield, 1982; Kohonen, 1995; Merényi, 1998; Rumelhart et al., 1986]), they typically consist of an interconnected network of nodes (weights) that are iteratively adjusted according to the characteristics of training data to minimize the difference between the network predictions and its desired outputs (i.e., the training labels). One form of neural network that we consider in this work is known as Generalized Learning Vector Quantization (GLVQ, [Sato and Yamada, 1996]), and its extension, Generalized Relevance Learning Vector Quantization (GRLVQ, [Hammer and Villmann, 2002]). The weights in an GLVQ network are a set of L labeled prototypes $W = \{(\mathbf{w}_l, y_l)\}_{l=1}^{L}$, $y_l \in \{1, \ldots, K\}$, that reside in the same n-dimensional feature space as the training samples. Learning a GLVQ network involves adjusting the prototypes through by randomly selecting a training sample $\mathbf{x}_i \in X^{tr}$ and moving $\mathbf{w}^+$ – the nearest prototype (in terms of the squared Euclidean distance $\mathrm{d}(\cdot, \cdot)$) with the same label as training sample $\mathbf{x}_i$ – towards $\mathbf{x}_i$, and moving $\mathbf{w}^-$ – the nearest prototype with a different label as $\mathbf{x}_i$ – away from $\mathbf{x}_i$, according to the update rule

$$\Delta \mathbf{w}^{\pm} = \frac{\mp\eta \ \mathrm{d}(\mathbf{x}, \mathbf{w}^{\pm})}{(\mathrm{d}(\mathbf{x}, \mathbf{w}^{+}) + \mathrm{d}(\mathbf{x}, \mathbf{w}^{-}))^2} \frac{\delta}{\delta \mathbf{w}^{\pm}} \mathrm{d}(\mathbf{x}, \mathbf{w}^{\pm}), \tag{1.3}$$

where $\frac{\delta}{\delta \mathbf{w}^{\pm}}$ denotes the gradient with respect to $\mathbf{w}^+$ or $\mathbf{w}^-$, respectively. The learning rate η controls the rate of gradient descent. This procedure minimizes the energy function

$$E(W) = \sum_{i=1}^{N} \Phi(\mu(\mathbf{x}_i)) \text{ for } \mu(\mathbf{x}_i) = \frac{\mathrm{d}(\mathbf{x}_i, \mathbf{w}^{+}) - \mathrm{d}(\mathbf{x}_i, \mathbf{w}^{-})}{\mathrm{d}(\mathbf{x}_i, \mathbf{w}^{+}) + \mathrm{d}(\mathbf{x}_i, \mathbf{w}^{-})} \tag{1.4}$$

Here, $\Phi(x)$ is a monotonically increasing function. One proposed choice of Φ is the logistic function $\Phi(x) = (1 + \exp(-\sigma \cdot x))^{-1}$, where σ controls the steepness of the function [Kästner et al., 2011; Sato and Yamada, 1996]. In a GRLVQ network, we not only learn the prototypes W, but also an n-dimensional vector $\boldsymbol{\lambda}$ that characterizes the relevances of each feature. Learning $\boldsymbol{\lambda}$ involves an additional stochastic gradient-descent procedure where we minimize Equation (1.4) using the weighted distance $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathrm{diag}(\boldsymbol{\lambda})^{1/2} (\mathbf{x}_i - \mathbf{x}_j)$. Once learned, both GLVQ and GRLVQ predict the label of test sample $\mathbf{x}$ based on the label of its nearest prototype $\mathbf{w}'$.

$$y = \underset{j \in \{1, \ldots, K\}}{\operatorname{argmin}} \left(\min_{\mathbf{w}' \in W_j} (\mathrm{d}(\mathbf{x}, \mathbf{w}')) \right), \tag{1.5}$$

where W_j is the set of prototypes representing the j^{th} class.

Support Vector Machines Support Vector Machines (SVMs, [Cortes and Vapnik, 1995]) are another type of sophisticated classification algorithm that have shown good performance for high-dimensional data. Given the set of training samples, a SVM constructs a (possibly high-dimensional) hyperplane that separates each pair of training classes with the largest margin. Specifically, given the set of training samples $\mathbf{x}_i \in X^{tr}$ with labels $y_i \in \{-1, 1\}$, the SVM classifies test sample $\mathbf{x}$ according to

$$y = \sum_{i=1}^{N} y_i \alpha_i \mathrm{k}(\mathbf{x}, \mathbf{x}_i) + b, \tag{1.6}$$

where k is a kernel function, $\boldsymbol{\alpha}$ is a vector of weights that produce the largest margin between the training samples from each class, and $b \in \mathbb{R}$ is a bias term. Computing the $\alpha_i \in [0, C]$ weights involves solving a quadratic programming optimization problem in $\boldsymbol{\alpha}$ and b (details regarding solving this optimization problem can be found in [Cortes and Vapnik, 1995]). C is a regularization parameter that controls the convexity of the optimization function. A large value of C increases the difficulty of the optimization problem, as it involves finding a less-convex decision boundary that closely fits the manifold of the feature space, in comparison to the smoother decision boundaries produced by smaller values of C. The best value of C depends on the characteristics of input data, and is typically selected via cross-validation. When the number of training classes is greater than two, a single SVM is learned for each pair of classes (one-vs-one), or, alternatively, for each individual class vs. the remaining classes (one-vs-rest). The resulting predictions for each SVM are then combined using a variety of methods. Hsu et al. give a thorough review of many such methods in [Hsu and Lin, 2002].

1.5 Spectral Similarity Measures

The fundamental challenge we consider in this work lies in accurately measuring the similarity between hyperspectral signatures. Accurate similarity measurements are crucial, as they provide the mathematical basis used to distinguish between signatures representing different materials. To measure similarity between spectra, we use a pairwise function $\mathrm{k} : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ that produces a scalar, real-valued output given a pair of samples $\mathbf{x}_i$, $\mathbf{x}_j$. We call k a *similarity measure* when its output increases with the similarity of $\mathbf{x}_i$ and $\mathbf{x}_j$. Similarity is often phrased in terms of inverse distance: i.e., as the distance between a pair of samples decreases, their similarity increases. Similarity measures of this form are called *distance* or *dissimilarity* measures, and we use the notation $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ to denote such measures. Formally speaking, a distance measure is a function that satisfies the first two of the following properties [von Luxburg, 2004]

D1: $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_i) = 0$ (identity)
D2: $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) \geq 0$ (non-negativity)
D3: $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = d(\mathbf{x}_j, \mathbf{x}_i)$ (symmetry)
D4: $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) + d(\mathbf{x}_j, \mathbf{x}_l) \geq d(\mathbf{x}_i, \mathbf{x}_l)$ (triangle inequality)
D5: $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = 0 \implies \mathbf{x}_i = \mathbf{x}_j$ (definiteness)

When a distance measure satisfies all of the above properties, we refer to it as a *metric*. Since distance and similarity are closely-related concepts, several techniques exist to convert a similarity measure to a distance measure and vice-versa (see, e.g., [Hertz, 2006; von Luxburg, 2004]).

Most hyperspectral image classification algorithms employ *unweighted* (dis)similarity measures to compare spectral signatures. Unweighted measures make no assumptions on the relevances of individual spectral features, and thus, each feature contributes equally in measuring the distance between each pair of signatures. The (squared) Euclidean distance

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|^2 = (\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j), \tag{1.7}$$

the Spectral Angle (sometimes called the Spectral Angle Mapper, SAM [Yuhas et al., 1992]) distance

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = \cos^{-1}\left(\frac{\mathbf{x}_i^T \mathbf{x}_j}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|}\right) \tag{1.8}$$

and the Spectral Information Divergence (SID [Chang, 2000]),

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{KL}(\mathbf{x}_i||\mathbf{x}_j) + \mathrm{KL}(\mathbf{x}_j||\mathbf{x}_i) \tag{1.9}$$

which is a symmetrized version of the Kullback-Leibler divergence [Kullback and Leibler, 1951]

$$\mathrm{KL}(\mathbf{x}_i||\mathbf{x}_j) = \sum_{\ell=1}^{n} \mathbf{p}_{i,\ell} \log\left(\frac{\mathbf{p}_{i,\ell}}{\mathbf{p}_{j,\ell}}\right) \tag{1.10}$$

$$\mathbf{p}_i = \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|_1}, \quad \mathbf{p}_j = \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|_1} \tag{1.11}$$

are examples of unweighted dissimilarity measures often used to compare spectral signatures [Chang, 2000; Keshava, 2004; Li et al., 2006; Robila, 2004; van der Meer, 2006].

Kernel functions are another type of similarity measure that are widely used in the remote-sensing community due to their attractive theoretical properties [Camps-Valls and Bruzzone, 2005; Mwebaze et al., 2011]. A kernel function $\mathrm{k} : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is a similarity measure that satisfies

$$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \tag{1.12}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product between the two arguments, and $\phi(\mathbf{x}) : \mathbf{x} \rightarrow \phi(\mathbf{x}) \in \Phi$ is a mapping to a (possibly high-dimensional) dot product space Φ. Commonly-used kernels are the linear kernel

$$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle = \mathbf{x}_i^T \mathbf{x}_j, \tag{1.13}$$

and the radial basis function (RBF) kernel with width γ

$$\mathrm{k}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\gamma} \right). \tag{1.14}$$

In some cases, scientists are aware of relevant characteristics of data or specific classes of interest. For example, spectra representing vegetation can often be characterized by broad, slowly varying spectral features, while materials such as minerals and gases posses very narrow spectral features [Shaw and Burke, 2003]. When such domain-specific knowledge is available, we can design a *application-specific* similarity measure that emphasizes the aspects of the data that are useful for predicting class labels. For instance, the positions/widths of spectral absorption features typify the composition and abundance of material(s) the spectra represent. Several similarity measures have been proposed that emphasize spectral absorption features by measuring differences between Continuum-Removed (CR) spectral signatures, rather than the original, Continuum-Intact (CI) spectra. Figure 1.1 illustrates the difference between CI and CR spectra. Examples of CR-based measures include Spectral Feature Fitting (SFF, [Clark et al., 2003, 1990]) and Cross-Correlation Spectral Matching for

Continuum Removed signatures (CCSM-CR, [van der Meer, 2000]). These application-specific similarity functions often achieve state-of-the-art performance, owing to their capability to emphasize a fixed set of discriminative spectral features for spectral classes of interest. However, while domain knowledge can provide guidance regarding which spectral features are relevant for specific materials, their relative importances depend on the spectra considered in each study.

1.6 Developments in Adaptive Similarity Measures

A drawback of the aforementioned measures is that they do not consider which features are most relevant to a specific classification task. Such measures are susceptible to noise or features irrelevant to the task, and may produce ambiguous or misleading outputs when the chosen features are not discriminative. For instance, Clark et al. [2003] show that very different materials may have nearly identical CR spectra, and thus, CR-based similarity measures cannot distinguish between such materials. Rather than manually evaluating many different similarity measures to determine which best suits a given task, it is often advantageous to employ *adaptive* similarity measures that automatically adjust to characteristics of input data. We describe the three different (but not necessarily mutually exclusive) types of adaptive similarity measures we develop in this thesis, below.

1.6.1 Hybrid Similarity Measures

In many real-world situations, viewing different aspects of the data may lead to several different, but equally valid, notions of similarity. When such notions of similarity are complimentary to one another, a weighted combination of measurements produced by several distinct similarity measures can produce more accurate results than those produced by each of the individual measures. One means to combine similarity measures is to define a hybrid measure consisting of convex combination of L distinct similarity measures d_l as follows:

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{l=1}^{L} \alpha_l \mathrm{d}_l(\mathbf{x}_i, \mathbf{x}_j), \;\; \alpha_l \in [0, 1], \; \sum_{l=1}^{L} \alpha_l = 1 \tag{1.15}$$

Here, α_l is a convex weight parameter that determines how much each of the d_l contributes to d. The objective in hybrid metric learning is to choose α_l that combines the individual d_l measures in such a way that d is more accurate than each of the d_l measures. We propose a novel technique to learn α_l weights based upon multiclass LDA, and introduce several new hybrid similarity measures in Chapter 3.

1.6.2 Band-Weighted Similarity Measures

Learning a weighted combination of distinct similarity measures is not the only way to adapt a similarity measure to characteristics of data. A complimentary approach to the hybrid metric learning approach is to assign a weight to each spectral band according to its relevance to the classification problem. A widely-used approach (e.g.,[Davis et al., 2007; Globerson and Roweis, 2006; Goldberger et al., 2005a; Tsang et al., 2005; Weinberger et al., 2006; Xing et al., 2003]) to achieve this is to compute a transformation matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$ that maps $\mathbf{x}_i$ and $\mathbf{x}_j$ to an m-dimensional feature space where classes are better separated. This transformation induces a Mahalanobis measure [Mahalanobis, 1936]

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j), \tag{1.16}$$

where $\mathbf{M}$ is a positive semidefinite matrix that can be decomposed into the product $\mathbf{M} = \mathbf{A}^T\mathbf{A}$. One classical technique for Mahalanobis metric learning that has generated much recent interest in the hyperspectral imaging community (e.g., [Bandos et al., 2007; Du, 2007; Weizman and Goldberger, 2009]) is multiclass Linear Discriminant Analysis (LDA). Multiclass LDA is an extension of the original two-class LDA formulation by Fisher [1936] proposed by Rao [1948] that maps n-dimensional samples belonging to K classes into a $K-1$-dimensional feature space where, under certain conditions, classes are better separated. Ghodsi et al. [2008] recently showed that the closed-form multiclass LDA solution performed competitively compared to the state-of-the-art metric learning algorithms which use computationally expensive optimization routines proposed by Xing et al. [2003] and Globerson and Roweis [2006]. We provide a detailed evaluation of, and novel extensions to, Multiclass LDA in Chapter 4.

1.6.3 Inter-domain Similarity Measures

Conventional metric learning techniques are designed for intra-domain scenarios, and typically perform poorly in inter-domain scenarios where the training and test samples are drawn from different distributions. In such inter-domain scenarios, our objective is to learn a similarity measure to compare samples from different domains. By incorporating domain-specific characteristics into a similarity measure, we can use pre-existing classifiers or models rather than building new models tied to a specific learning system. One means to build such a similarity measure is to embed samples into a *dissimilarity space*. In a dissimilarity space, each sample is replaced with a new representation consisting of similarity measurements to a reference set consisting of several training samples. This transformation was initially proposed by Pkalska and Duin [2002] who showed empirically that when the reference set is discriminative for the classes of interest, the dissimilarity space is better separated than the original feature space. Recent work by Balcan et al. [2006] provides theoretical justification for this phenomenon by showing that if a set of classes are linearly separable using a particular similarity or distance measure, the dissimilarity space representation is potentially as expressive as a high-dimensional kernel space. In Part III, we describe how we leverage these results to design a novel similarity measure capable of reconciling differences between samples residing in similar, but not identical, feature spaces.

Chapter 2

Material Identification with Library-based Spectral Matching

Portions of this chapter are based upon the following publications:

- BD Bue, E Merényi, and B Csathó. “Automated Labeling of Segmented Hyperspectral Imagery via Spectral Matching”. *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* [Aug. 2009].
- BD Bue, E Merényi, and B Csathó. “Automated Labeling of Materials in Hyperspectral Imagery”. *IEEE Trans. on Geoscience and Remote Sensing* 48.11 [2010], pp. 4059–4070.

2.1 Library-based Spectral Matching

To gain insight into the challenges in automated material identification using hyperspectral imagery, we begin with an evaluation of *spectral matching* techniques. Because hyperspectral signatures are of sufficient spectral resolution to uniquely identify the materials they represent, spectroscopists can identify materials for unlabeled spectra by comparing them to *ground-truth* spectra captured under controlled conditions (i.e., field- or lab-measured spectra) with known material labels. Spectral matching techniques mimic this approach by predicting the material composition of unidentified spectra based upon their similarities to spectral signatures in libraries. Spectral matching algorithms are an indispensable tool for spectroscopists that greatly reduce the amount of time necessary to search through large spectral libraries such as those provided by the USGS [Clark et al., 2007] and NASA (e.g., RELAB [Pieters, 1990], ASTER [Baldridge et al., 2009] and CRISM [Slavney and Murchie, 2006]). The main challenge in spectral matching involve accurately and efficiently measuring the similarity between large quantities of spectral signatures. In this chapter,[*] we describe a spectral matching methodology that matches clusters of hyperspectral image signatures to library signatures of known material species. We evaluate the performance of several spectral similarity measures using this methodology, and propose a new similarity measure that accounts for spectral characteristics that are often poorly captured by canonical similarity measures. We show that our new measure yields more accurate material identifications than conventional measures, both visually and in terms of information-theoretic criteria.

2.2 Spectral Matching for Cluster Signatures

One challenge in spectral matching lies in efficiently comparing thousands/millions of hyperspectral image pixels to each spectrum in a spectral library (which may, itself, consist of thousands of signatures). In addition to the computational expense, pixels taken independently are sensitive to instrument noise and intra-class variability [Thompson et al., 2010]. A promising method to reduce both noise and computational costs in spectral matching is to consider *clusters* of similar spectra that capture the most relevant spectral variations in the image, rather than individual pixels. This gives rise to the methodology shown in Figure 2.1. After a hyperspectral image has been clustered, each cluster consists of the set of pixels most similar to one another, according to the clustering algorithm. Because the pixels in each cluster are similar, we can summarize each cluster by its mean spectral signature (we use *mean signature* and *cluster signature* interchangeably). To assign a material label to a cluster, we calculate

[*] The material presented in this chapter was performed in collaboration with Erzsébet Merényi and Bea Csathó, with assistance from Dar Roberts and Bill Farrand.

the similarity between its mean signature and the library signatures. We assume that each spectral signature in the library is a unique descriptor for the material it represents. Therefore, if the similarity measure yields a high similarity score for a given cluster signature and a particular library signature, we can assign the material label from the library signature to the members (pixels) of that cluster.

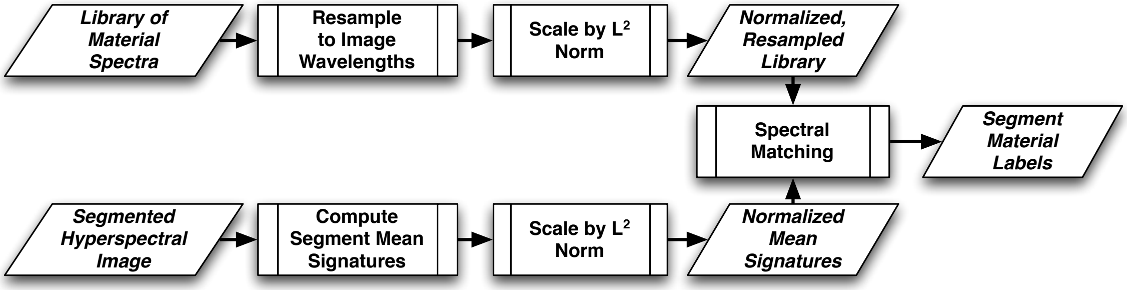

Library of Material Spectra
Resample to Image Wavelengths
Scale by L2 Norm
Normalized, Resampled Library
Spectral Matching
Segment Material Labels
Segmented Hyperspectral Image
Compute Segment Mean Signatures
Scale by L2 Norm
Normalized Mean Signatures

Figure 2.1 : Spectral matching methodology. Image pixels representing unknown materials are identified by comparing the mean signatures of groups of similar pixels (clusters) to library signatures with known material labels. Both library and mean signatures are normalized by their L^2 norms to account for linear illumination effects.

However, while current spectral libraries generally contain a wide variety of distinct material spectra, they often do not capture the diverse variations of individual materials that can be extracted from hyperspectral imagery. In particular, typical spectral libraries contain few (usually less than ten) samples of each distinct material species. Consequently, conventional classification techniques cannot robustly model many distinct material classes with so few samples of each class, especially for high-dimensional hyperspectral signatures. In contrast, spectral matching techniques simply return a set of “hit lists” of the most similar material constituents for unidentified spectra, which a scientist can interpret to verify if the correct material(s) are identified.

2.3 Similarity Measures for Continuum-Intact (CI) and Continuum-Removed (CR) Signatures

We evaluate spectral matching performance using both the Euclidean distance (denoted d_{ED}, Equation (1.7)), which is one of the most commonly used similarity measures used to compare spectral signatures [Chang, 2000; Du and Chang, 2001; Keshava, 2004; Sweet, 2004; Tarabalka et al., 2009b], and the Spectral Information Divergence (denoted $\mathrm{d}_{\mathrm{SID}}$, Equation (1.9)), which was recently shown to outperform several canonical spectral similarity measures (including d_{ED}) in discriminating between spectrally-similar spectra representing minerals such as alunite, kaolinite, montmorillionite and quartz [van der Meer, 2006].

The d_{ED} and $\mathrm{d}_{\mathrm{SID}}$ are examples of similarity measures which take Continuum-Intact (CI) spectral signatures as input (as shown in Figure 1.1). However, as mentioned in Section 1.5, CR spectral signatures often better capture the composition and concentration of the material(s) each spectral signature represents [van der Meer, 2004] than their CI counterparts. For this reason, we also consider the spectral matching performance using the d_{ED} and $\mathrm{d}_{\mathrm{SID}}$ similarity measures with CR signatures as input

$$\mathrm{d}_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{d}(\mathrm{CR}(\mathbf{x}_i), \mathrm{CR}(\mathbf{x}_j)), \tag{2.1}$$

where $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ is either d_{ED} or $\mathrm{d}_{\mathrm{SID}}$, and $\mathrm{CR}(\mathbf{x})$ is a function that returns the CR representation of $\mathbf{x}$. The output of $\mathrm{CR}(\mathbf{x})$ is a vector of the same dimensionality as $\mathbf{x}$ with components in the range $[0, 1]$, where values of zero lie on the estimated continuum and values greater than zero indicate the depth of absorptions relative to the estimated continuum. We estimate the continuum of a given spectrum by fitting a piecewise linear function to local maxima, which is then divided out of the original CI spectrum. Pseudocode for our continuum-removal algorithm, loosely based upon Clark et al. [1987], is given in Appendix 2.A.

2.3.1 The CICR Similarity Measure

While the CR representation better accounts for differences in absorption features than the CI representation, the CR representation alone can be an unreliable descriptor for material identification because signatures with considerably different continuua can have equivalent CR representations [Clark and Rousch, 1984; Howell et al., 1994]. To compensate for this shortcoming, we introduce a new, *hybrid* similarity measure, $\mathrm{d}_{\mathrm{CICR}}$, that combines CI and CR distance measurements, thereby capturing differences in both continuum shape and absorption features

$$\mathrm{d}_{\mathrm{CICR}}(\mathbf{x}_i, \mathbf{x}_j, \alpha) = \mathrm{d}_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) + \alpha \mathrm{d}_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) \tag{2.2}$$

$$\mathrm{d}_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{v_{\mathrm{CI}}} \mathrm{d}\left(\mathbf{x}_i, \mathbf{x}_j\right) \tag{2.3}$$

$$\mathrm{d}_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) = \frac{1}{v_{\mathrm{CR}}} \mathrm{d}\left(\mathrm{CR}(\mathbf{x}_i), \mathrm{CR}(\mathbf{x}_j)\right). \tag{2.4}$$

Here $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathrm{CR}(\mathbf{x})$ are defined as in Equation (2.1), α is a scalar weight parameter that controls the influence of the d_{CI} vs. d_{CR} terms, v_{CI} and v_{CR} are scaling factors (described below) that equalize the influence of the d_{CI} and d_{CR} measures. We set v_{CI} (v_{CR}) to the variance of all pairwise CI (CR) distances between the library and mean signatures, and set $\alpha = 0.5$ so the d_{CI} and d_{CR} distances contribute equally to the similarity measurement. We investigate the influence of these parameters later in Chapter 3. To distinguish between the CI-based (d_{CI}), CR-based (d_{CR}) and CICR-based ($\mathrm{d}_{\mathrm{CICR}}$) distance measures, we hereafter refer to the corresponding d_{ED} and $\mathrm{d}_{\mathrm{SID}}$ measures as ($\mathrm{CI}_{\mathrm{ED}}$, $\mathrm{CR}_{\mathrm{ED}}$, $\mathrm{CICR}_{\mathrm{ED}}$) and ($\mathrm{CI}_{\mathrm{SID}}$, $\mathrm{CR}_{\mathrm{SID}}$, $\mathrm{CICR}_{\mathrm{ED}}$), respectively. As described in Section 1.3.2, we normalize the CI signatures by their L^2 norms to mitigate scaling factors caused by linear illumination effects.[†]

2.4 Criteria for Evaluating Similarity Measures

2.4.1 Information-Theoretic Criteria

We consider the information-theoretic criteria proposed by Chang in [Chang, 2000]: the Spectral Discriminatory Probability ($\mathrm{SDP}^{\mathrm{d}}$), Spectral Discriminatory Entropy ($\mathrm{SDE}^{\mathrm{d}}$), and the Power of Spectral Discrimination (PW^{d}). Each of these three criteria characterizes the capability of a distance measure $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ to distinguish a reference signature from a set of library signatures. In this work, we measure the discriminatory capabilities of each measure with respect to each cluster mean signature $\mathbf{c}$, and the *hit list* $\mathbf{L}^{\mathbf{c}} = \{\mathbf{l}_1, \ldots, \mathbf{l}_m\}$ consisting of the m library signatures most similar to $\mathbf{c}$. We consider hit lists of size three ($m = 3$) to balance the amount of manual validation while providing a satisfactory demonstration of the technique.

The Spectral Discriminatory Probability calculates the probability of distinguishing cluster signature $\mathbf{c}$ from a library signature $\mathbf{l}_k \in \mathbf{L}^{\mathbf{c}}$ using distance measure $\mathrm{d}(\cdot, \cdot)$.

$$\mathrm{SDP}^{\mathrm{d}}(\mathbf{c}, \mathbf{l}_k) = \frac{\mathrm{d}(\mathbf{c}, \mathbf{l}_k)}{\sum_{j=1}^{m} \mathrm{d}(\mathbf{c}, \mathbf{l}_j)}. \tag{2.5}$$

A small $\mathrm{SDP}^{\mathrm{d}}$ value indicates the probability of distinguishing the cluster signature and library signature is low, within the context of hit list. Thus, the “best” matches,

[†] Under this scaling, the d_{ED} distance is functionally equivalent to the cosine (or SAM) distance (Equation (1.8)): $\mathrm{d}_{\mathrm{ED}}(\mathbf{x}_i, \mathbf{x}_j) = 2\sqrt{1 - \cos(\mathrm{d}_{\mathrm{SAM}}(\mathbf{x}_i, \mathbf{x}_j))}$.

according to measure $\mathrm{d}(\cdot,\cdot)$, are those with the smallest $\mathrm{SDP}^{\mathrm{d}}$ values.

The Spectral Discriminatory Entropy quantifies the uncertainty in identifying cluster signature $\mathbf{c}$ from the spectra in the hit list $\mathbf{L}^{\mathbf{c}}$.

$$\mathrm{SDE}^{\mathrm{d}}(\mathbf{c},\mathbf{L}^{\mathbf{c}}) = -\sum_{j=1}^{m} \mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_j)\log \mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_j). \tag{2.6}$$

The $\mathrm{SDE}^{\mathrm{d}}$ takes values in the range $0 < \mathrm{SDE}^{\mathrm{d}} \leq \log\frac{1}{m}$, reaching its maximum when all m values are equal. A smaller value indicates a better chance of distinguishing $\mathbf{c}$ from the library signatures $\mathbf{L}^{\mathbf{c}}$.

The Power of Spectral Discrimination estimates the power of distance measure $\mathrm{d}(\cdot,\cdot)$ to discriminate between a pair of library signatures $\mathbf{l}_i, \mathbf{l}_j \in \mathbf{L}^{\mathbf{c}}$, with respect to cluster signature $\mathbf{c}$.

$$\begin{align}\mathrm{PW}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_i,\mathbf{l}_j) &= \max\left\{\frac{\mathrm{d}(\mathbf{c},\mathbf{l}_i)}{\mathrm{d}(\mathbf{c},\mathbf{l}_j)}, \frac{\mathrm{d}(\mathbf{c},\mathbf{l}_j)}{\mathrm{d}(\mathbf{c},\mathbf{l}_i)}\right\} \tag{2.7}\\ &= \max\left\{\frac{\mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_i)}{\mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_j)}, \frac{\mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_j)}{\mathrm{SDP}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_i)}\right\}. \tag{2.8}\end{align}$$

PW^{d} values near one indicate that $\mathbf{l}_i$ and $\mathbf{l}_j$ are "indistinguishable" with respect to cluster signature $\mathbf{c}$. To consider hit lists with more than two signatures, we calculate the mean Power of Spectral Discrimination for the corresponding m library signatures in the hit list.

$$\overline{\mathrm{PW}^{\mathrm{d}}(\mathbf{c},\mathbf{L}^{\mathbf{c}})} = \frac{2}{m(m-1)}\sum_{i=1}^{m}\sum_{j=i+1}^{m} \mathrm{PW}^{\mathrm{d}}(\mathbf{c},\mathbf{l}_i,\mathbf{l}_j) \tag{2.9}$$

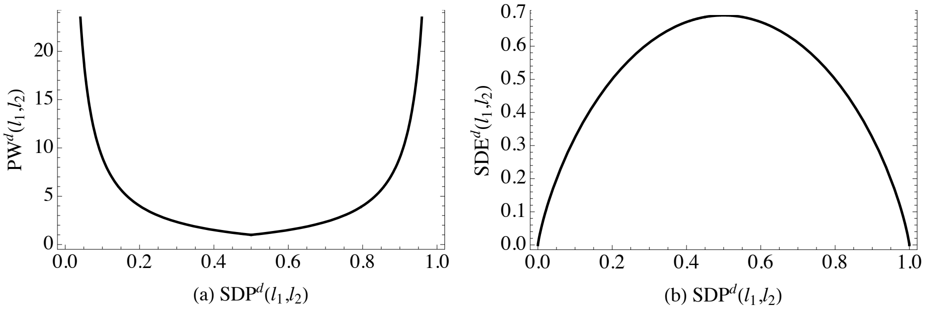
The mean PW^{d} for library signatures in $\mathbf{L}^{\mathbf{c}}$ characterizes how "tightly packed" the distances are between the library signatures in the hit list and the cluster signature $\mathbf{c}$. We want this value to increase for dissimilar signatures, and to approach unity for similar signatures. However, the PW^{d} may become skewed if distances between $\mathbf{c}$ and the signatures in its hit list are relatively far apart (as demonstrated in Section 2.4.4). This often indicates that $\mathbf{c}$ is not well-represented in the spectral library, since the dissimilar library signatures potentially represent materials different from the other signatures in the hit list.

On the Order Equivalence of $\mathbf{PW^d}$ and $\mathbf{SDE^d}$

Both the mean $\mathrm{PW^d}$ and $\mathrm{SDE^d}$ estimate the uncertainty in distinguishing cluster signatures from library signatures. By estimating this uncertainty, we can compare the capabilities of one similarity measure vs. another with respect to a fixed set of spectral signatures [Chang, 2000; Du and Chang, 2001; Du et al., 2004]. However, we will now show that the ordering produced by the $\mathrm{PW^d}$ is equivalent to the ranking generated by the $-\mathrm{SDE^d}$ (i.e., the $\mathrm{PW^d}$ and $\mathrm{SDE^d}$ produce rankings that are order isomorphic).

Figure 2.2 shows the functional behavior of the $\mathrm{PW^d}$ and $\mathrm{SDE^d}$ for a given reference signature $\mathbf{c}$ vs. two library signatures $\mathbf{l}_1$and $\mathbf{l}_2$. We can see that the $\mathrm{PW^d}$ ($\mathrm{SDE^d}$) is a convex (concave) function, minimized (maximized) at the location $\frac{1}{2}$. Taken independently, the $\mathrm{PW^d}$ is better suited to discriminate between values at the extreme ends of the distribution, whereas the $\mathrm{SDE^d}$ gives better separation across the mid-range. However, the two functions are order isomorphic because the $\mathrm{PW^d}$ and $-\mathrm{SDE^d}$ are monotonically decreasing before, and monotonically increasing after the minimum location, and consequently produce the same order of rankings.

This isomorphism also holds for hit lists with $m > 2$ library signatures. In this case, the mean $\mathrm{PW^d}$ is used (Equation (2.9)). We previously showed that, when $m = 2$, the $\mathrm{PW^d}$ is a convex function. Since the sum of convex functions is convex, the mean $\mathrm{PW^d}$ is also convex, and attains its minimum value of one at $\frac{1}{m}$ (when all outcomes are equiprobable). Because entropy is a concave function, $-\mathrm{SDE^d}$ is a convex function,



(a) SDP$^d(l_1,l_2)$ (b) SDP$^d(l_1,l_2)$

Figure 2.2 : The functional forms of the $\mathrm{PW^d}$ (a) and $\mathrm{SDE^d}$ (b) for $\mathbf{L^c} = \{\mathbf{l}_1, \mathbf{l}_2\}$, according to Equations (2.6) and (2.8).

with minimum at $\log \frac{1}{m}$, also attained when all outcomes are equiprobable. Once again, we have two convex functions with minima at the same location, where the $-\mathrm{SDE^d}$ (and also, the $\mathrm{PW^d}$) monotonically increases after the minimum. Thus, the order isomorphism holds when $m > 2$. Consequently, if our objective is to rank the relative performances of different similarity measures, the $\mathrm{PW^d}$ and $\mathrm{SDE^d}$ both yield the same ranking. For this reason, our subsequent analysis focuses only on the $\mathrm{PW^d}$.

Significance Testing

We assess the significance of our $\mathrm{PW^d}$-based comparisons using the Wilcoxon Signed-Rank Test (WSRT) [Wilcoxon, 1945, 1947]. The WSRT is a non-parametric statistical hypothesis test for paired measurements – in our case, N measurements $(\mathrm{d}_{1,i}, \mathrm{d}_{2,i})$, $i \in \{1, \ldots, N\}$ using two different similarity measures d_1 and d_2 – on a single sample (i.e., a cluster signature). Three quantities define the WSRT: the number of trials performed, N_t, the sum of positive differences in paired measurements, $W^+ = \sum_{i=1}^{N} \mathrm{I}(\mathrm{d}_{1,i} - \mathrm{d}_{2,i} > 0)$, and the sum of negative differences in paired measurements, $W^- = \sum_{i=1}^{N} \mathrm{I}(\mathrm{d}_{1,i} - \mathrm{d}_{2,i} < 0)$, where $\mathrm{I}(\cdot)$ is the indicator function. Equal measurements are handled by adding their mean to both W^+ and W^-. The significance of the performance is based on N_t and $\max(W^+, W^-)$. Using the WSRT to test significance of spectral similarity measure comparisons has several advantages. First, it makes no assumptions on the underlying distribution of the measurements. Second, greater emphasis is placed on larger differences in measurements than on smaller ones. Third, because the statistic for the signed rank test is unaffected by changes in a few observations (i.e., it is a *resistant* statistic), outliers are naturally suppressed when the number of outliers is not particularly large. For a detailed discussion on the WSRT, see [Demšar, 2006].

2.4.2 Visual Criteria

As we see later in this chapter, the information-theoretic measures described in the previous section are sensitive to spectral representation (i.e., CI vs. CR), and often do not capture visually strong matches. Therefore, we provide a manual assessment of the perceived quality of spectral matches by assigning a *visual score* (VS) in the range $[0, 3]$ to each in the set of m signatures the hit list for each cluster signature, as determined by each similarity measure. A visual score of zero indicates poor quality of all m matches, in terms of overall spectral shape and the positions of absorption features. A score of one indicates the majority (but not all) of the m matches are of poor quality, a score of two indicates the majority of the matches are of good quality, and if all m signatures strongly match the cluster signature, we assign a score of three. Four independent observers assessed the hit lists produced by each measure to corroborate the visual scores with adequate confidence.

2.4.3 Case Study: Ocean City AVIRIS Spectra

We evaluate spectral matching and material identification performance on a Low-Altitude Airborne Visible / Infrared Imaging Spectrometer (AVIRISLA) [Green et al., 1998] hyperspectral image of Ocean City, MD [Csathó et al., 1998]. This image (acquired Nov 5, 1998, with 4 m / pixel spatial resolution, in 224 spectral bands from 0.4 2.5 μm) is an example of the complexity in a real urban study, with many ($\sim$30) material species of interest. It was analyzed in previous work to capture spectral clusters, verify them against field knowledge and identify materials they represent, as reported in detail in [Merényi et al., 2007]. Figure 2.3 gives a false color composite of the image. The white boxes indicate the sub-regions we consider in this work. We consider two different clusterings of the Ocean City image, both generated and analyzed by Merényi et al. [2007]. The first clustering was produced using a Self-Organizing Map (SOM), and is shown in Figure 2.4. The high spatial and spectral resolution of AVIRIS imagery, along with the sensitive SOM-based clustering technique allowed discrimination of 35 clusters with varied characteristics including (very) small and spectrally similar ones. As verified from field data, most of the SOM clusters represent objects associated with distinct material types. Examples of these are: water tower, buildings, roads, boardwalks, parking lots, a mini golf course, a coast guard lookout tower, and landscape units. However, we do not have corresponding material labels for some of the clusters that can be recognized on the functional level (e.g., the tennis court, mini golf course clusters). The second clustering was produced using the ISODATA algorithm, and is shown in Figure 2.5. We emphasize

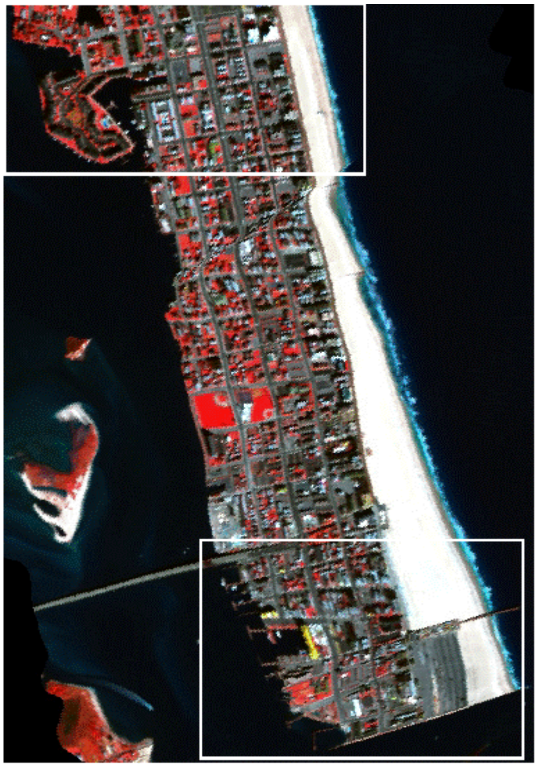


Figure 2.3 : Color composite of Ocean City, MD AVIRIS image using the (0.8749, 0.683, 0.5468) μm bands. Figure credit: Merényi et al. [2007].

that the cluster labels (colors) in the SOM-based and ISODATA segmentation are not consistent with each other because reconciling clusters is nontrivial or impossible, since there is not a one-to-one (or even a clean one-to-many) correspondence between the two clusterings. The ISODATA clustering contains a total of 21 clusters that differ greatly from the SOM clusters. In particular, a number of spectrally-similar materials delineated in the SOM clustering are assigned to several quite different ISODATA clusters. We anticipate poor spectral matching performance using the ISODATA clustering, since the ISODATA cluster signatures do not accurately capture distinctions between material species. Thus, many of the ISODATA clusters give no clear, or worse, misleading material interpretations.

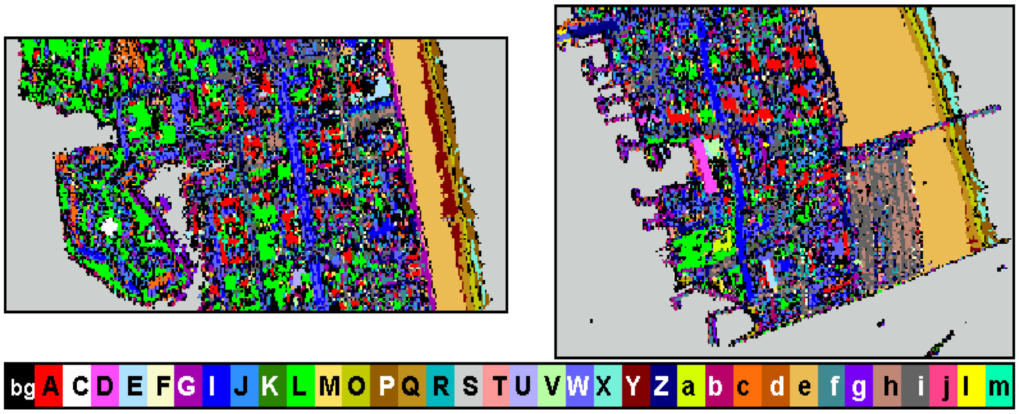


Figure 2.4 : The 35 clusters of the Self-Organizing Map-based segmentation of the Ocean City AVIRIS image produced by Merényi et al. [2007]. Left: the northern boxed area from Figure 2.3. Right: the southern boxed area from Figure 2.3. Black ("bg") pixels indicate regions that are not assigned to any of the 35 clusters. Figure credit: Merényi et al. [2007]

The spectral library we use consists of 1250 signatures from three sources: (1) 1164 field-measured spectra of mostly urban materials acquired in 1075 wavelengths in the 0.35 to 2.4 μm range (described in [Herold et al., 2004]); (2) 17 lab-measured vegetation spectra from the USGS splib06a spectral library [Clark et al., 2007]; (3) 21

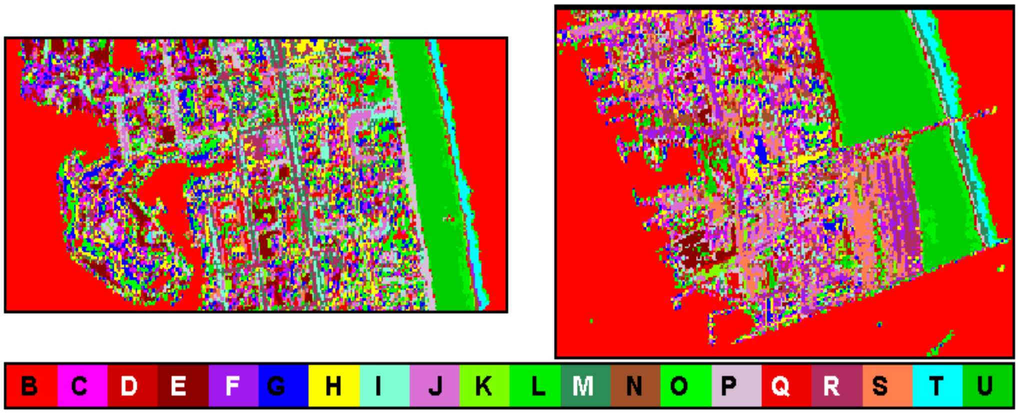


Figure 2.5 : The 21 clusters of the ISODATA segmentation of the Ocean City AVIRIS image produced by Merényi et al. [2007]. Left: the northern boxed area from Figure 2.3. Right: the southern boxed area from Figure 2.3. Figure credit: Merényi et al. [2007]

AVIRIS image spectra (mostly vegetation and soil types) from expert-labeled regions described in [Merényi et al., 2000]. All library signatures are tagged with metadata indicating the objects measured, and most include a corresponding material label. We resampled the library signatures to the appropriate AVIRIS wavelengths and full width at half maximum (FWHM) parameters, and exclude wavelengths outside of the range [0.42, 2.39] μm due to noise present in some of the library signatures. The remaining 165 of the original 224 AVIRIS bands are used for spectral matching.

2.4.4 Evaluation of Spectral Matching Results

Here, we evaluate the spectral matching performance on the Ocean City SOM clusters according to the d_{CI}, d_{CR}, $\mathrm{d}_{\mathrm{CICR}}$-based measures. We provide the mean PW^{d} scores for the hit lists consisting of the top three library matches (i.e., $m = 3$) for each cluster mean signature in Figure 2.6. Table 2.1 gives the set of all visual scores for the Ocean City SOM cluster hit lists, along with summary statistics for the visual and PW^{d} scores for all cluster signatures (hereafter referred to as **All**) and signatures which are adequately represented in the library (hereafter referred to as **Selected**). We make the simplifying assumption that if the mean visual score (for all measures) for a cluster signature is zero, then that signature does not have a representative library signature. These include clusters P, S, X, a, and c. We also exclude clusters C, F and d because the corresponding materials could not be precisely determined from field data with adequate confidence. The *objects* which these clusters represent are as follows: C is a green tennis court shown in Figure 2.8, discussed below; F is a street/sidewalk; and d is likely a mixture of water and nearby building materials. For each similarity measure, we also provide the mean visual and $\mathrm{PW^d}$ scores, respectively in Tables 2.2 and 2.3.

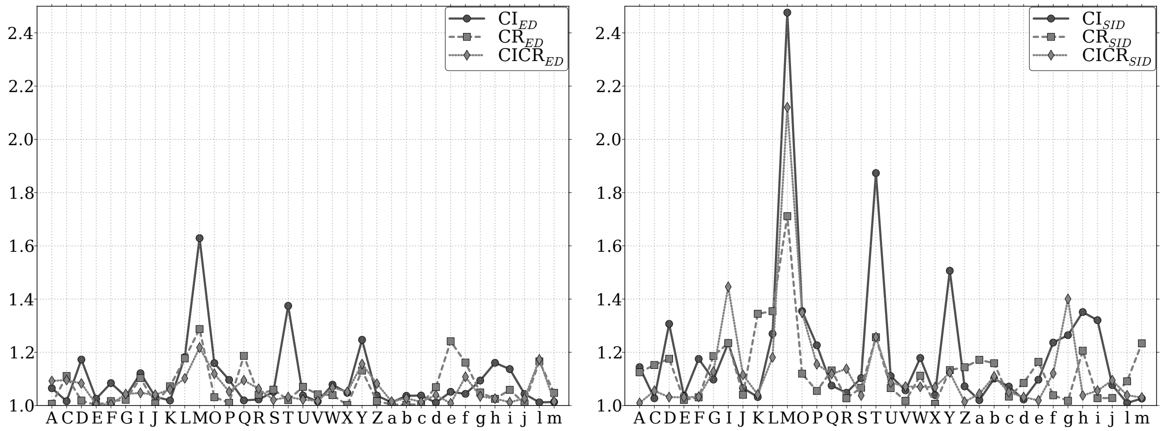


Figure 2.6 : Left: mean $\mathrm{PW^d}$ scores ($m = 3$) for the Ocean City SOM cluster signatures according to the $\mathrm{CI_{ED}}$ (solid line, circle marker), $\mathrm{CR_{ED}}$ (dashed line, square marker), and $\mathrm{CICR_{ED}}$ (dotted line, diamond marker) measures. Right: mean $\mathrm{PW^d}$ scores for each cluster signature using the $\mathrm{CI_{SID}}$ (solid line, circle marker), $\mathrm{CR_{SID}}$ (dashed line, square marker), and $\mathrm{CICR_{SID}}$ (dotted line, diamond marker) measures. On average, $\mathrm{d_{ED}}$-based measures yield lower $\mathrm{PW^d}$ scores than $\mathrm{d_{SID}}$-based measures, indicating that the signatures in the $\mathrm{d_{ED}}$-based hit lists are more similar to one another than the $\mathrm{d_{SID}}$-based hit lists. Cluster M (discussed in detail below) is the most spectrally ambiguous according to the similarity measures we consider.

As shown in Table 2.2, on average, the $\mathrm{d_{ED}}$ and $\mathrm{d_{SID}}$ yield equivalent performance as indicated by their visual scores. However, the mean $\mathrm{PW^d}$ scores shown in Table 2.3

	Visual Scores for Individual Clusters																																			Mean Visual Score		Mean PW^d	
	A	C	D	E	F	G	I	J	K	L	M	O	P	Q	R	S	T	U	V	W	X	Y	Z	a	b	c	d	e	f	g	h	i	j	l	m	All	Selected	All	Selected
CI$_{ED}$	2	1	1	1	1	1	2	1	2	3	1	1	0	1	1	0	1	2	1	2	0	2	2	0	1	0	0	2	2	0	1	1	1	1	1	1.1143	1.1053	1.0614	1.0708
CR$_{ED}$	0	2	2	0	1	0	2	0	2	3	2	1	0	1	1	0	1	2	2	1	0	1	1	0	1	0	1	2	2	1	0	1	1	1	0	1.0000	0.8947	1.0441	1.0503
CICR$_{ED}$	2	2	2	1	1	1	2	1	1	3	2	1	0	1	1	0	1	2	1	2	0	2	1	0	1	1	1	2	2	1	1	1	1	2	1	**1.2571**	**1.3684**	**1.0395**	**1.0439**
CI$_{SID}$	2	1	2	1	1	2	2	1	2	3	1	1	0	1	1	0	1	2	1	2	0	2	2	0	1	0	0	2	2	0	1	1	1	1	1	1.1714	1.2105	<u>1.1354</u>	<u>1.1586</u>
CR$_{SID}$	0	2	1	0	1	0	2	0	2	3	2	1	0	1	1	0	1	2	2	1	0	1	1	0	1	0	1	2	2	1	0	1	1	1	0	<u>0.9714</u>	<u>0.8421</u>	1.0908	1.1029
CICR$_{SID}$	1	2	2	1	1	1	2	1	2	3	2	1	0	1	1	0	1	2	2	1	0	1	1	0	1	1	1	2	2	1	1	1	1	2	1	1.2286	1.3158	1.0898	1.1047
Average	1	2	2	1	1	1	2	1	2	**3**	2	1	*0*	1	1	*0*	1	2	2	2	*0*	2	1	*0*	1	*0*	1	2	2	1	1	1	1	1	1				

Table 2.1 : Visual scores and averages for Ocean City SOM clusters. Visual scores are in the range $\{0, \dots, 3\}$, which increase in proportion to the perceived quality of the hit lists. The mean PW^d scores are in the range $[1, \infty]$, where a score of 1 indicates that all members of the hit list are equidistant from the cluster signature. Average scores are given for all clusters ("All") and clusters represented in the library ("Selected"). We assume clusters with average visual scores of zero (indicated by italics) are not represented in the library. The best scores are given in bold text, and the worst scores are underlined. On average d_{ED} and d_{SID} produce comparable performance as indicated by their visual scores, though the d_{ED}-based measures achieve lower PW^d scores than the d_{SID}-based measures. The d_{CICR}-based measures outperform both d_{CI} and d_{CR}-based measures with both low PW^d and high visual scores.

	$\mathbf{d_{CI}}$	$\mathbf{d_{CR}}$	$\mathbf{d_{CICR}}$	**Mean**
$\mathbf{d_{ED}}$	1.1143/1.1053	**1.0000/0.8947**	**1.2571/1.3684**	**1.1238/1.1228**
$\mathbf{d_{SID}}$	**1.1714/1.2105**	0.9714/0.8421	1.2286/1.3158	**1.1238/1.1228**
Mean	1.1429/1.1579	0.9857/0.8684	**1.2429/1.3421**	1.1238/1.1228

Table 2.2 : Mean visual scores for All/Selected clusters according to d_{ED} and d_{SID} using d_{CI}, d_{CR}, and d_{CICR}-based measures. The best scores are indicated in bold text. Both d_{ED} and d_{SID} produce equivalent performance on average, and are most visually agreeable using the d_{CICR}-based measure.

indicate that the hit lists produced by the d_{ED} are quantitatively more similar to their respective cluster signatures than those produced by the d_{SID}, particularly on the Selected clusters. This suggests that the d_{ED} (and, equivalently, the SAM distance) has slightly greater discriminatory power than the d_{SID} for this spectral matching task using both the CI and the CR representation.

When we compare the d_{CI}, d_{CR} and d_{CICR}-based measures independently, we also observe similar performance by both the d_{ED} and d_{SID}. In fact, the d_{CI}-based visual scores differ for only two clusters: **D** and **G**; and the d_{CR}-based scores differ only on cluster **D**.

	$\mathbf{d_{CI}}$	$\mathbf{d_{CR}}$	$\mathbf{d_{CICR}}$	**Mean**
$\mathbf{d_{ED}}$	**1.0614/1.0708**	**1.0441/1.0503**	**1.0395/1.0439**	**1.0483/1.1053**
$\mathbf{d_{SID}}$	1.1354/1.1586	1.0908/1.1029	1.0898/1.1047	1.0550/1.1221
Mean	1.0984/1.1147	1.0675/1.0766	**1.0647/1.0743**	1.0517/1.1137

Table 2.3 : Mean $\mathrm{PW^d}$ scores for All/Selected clusters according to the $\mathrm{d_{ED}}$ and $\mathrm{d_{SID}}$ using $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$, and $\mathrm{d_{CICR}}$-based measures. The best scores are indicated with bold text. The $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$ and $\mathrm{d_{CICR}}$ measures using the $\mathrm{d_{ED}}$ achieve lower $\mathrm{PW^d}$ scores than those using the $\mathrm{d_{SID}}$, indicating that the signatures in the $\mathrm{d_{ED}}$-based hit lists are more similar to one another than the $\mathrm{d_{SID}}$-based hit lists. Both $\mathrm{d_{ED}}$ and $\mathrm{d_{SID}}$ achieve their lowest respective $\mathrm{PW^d}$ with the $\mathrm{d_{CICR}}$ measure.

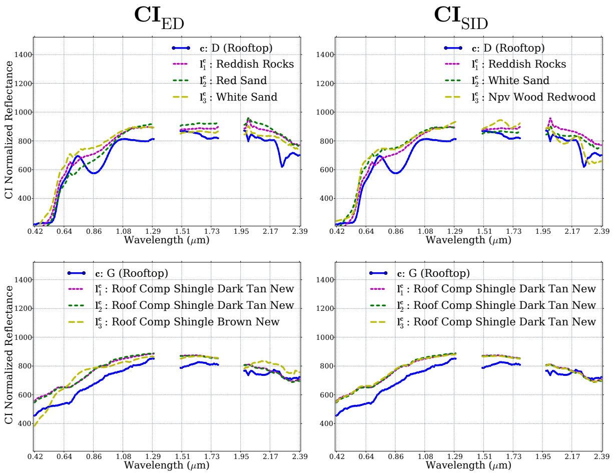


Figure 2.7 : $\mathrm{d_{CI}}$-based hit lists for SOM clusters **D** (top) and **G** (bottom) using the $\mathrm{d_{ED}}$ (left) vs. $\mathrm{d_{SID}}$. (right)

Figure 2.7 gives the $\mathrm{d_{CI}}$-based matches for clusters **D** and **G**, and suggests that the hit lists produced by $\mathrm{d_{SID}}$ include spectra that vary smoothly with the cluster signatures than the signatures in the $\mathrm{d_{ED}}$-based hit lists, whereas the $\mathrm{d_{ED}}$-based matches are closer in the least-squares sense, but occasionally do not follow the shape of the spectrum as well as the $\mathrm{d_{SID}}$-based matches. This gives the $\mathrm{d_{SID}}$ a slight edge in performance in terms of visual scores over the $\mathrm{d_{ED}}$ using CI-based signatures. On the often jagged CR signatures, however, the $\mathrm{d_{SID}}$ performs slightly worse than $\mathrm{d_{ED}}$. Because the $\mathrm{d_{SID}}$ assumes that the densities derived from the spectral signatures (Equation (1.11)) are smooth functions, so such performance is expected.

However, the $\mathrm{d_{CICR}}$ results are not only more visually agreeable than the $\mathrm{d_{CI}}$ and $\mathrm{d_{CR}}$ results, but better represent spectroscopic similarities between signatures by accounting for both spectral shape and absorption features. Such improvements are reasonably intuitive: $\mathrm{d_{CI}}$ measures produce spectral matches that correspond well in terms of spectral shape, but often fail to capture characteristic absorption features, as shown in Figure 2.8. Here, we compare the $\mathrm{d_{CI}}$ and $\mathrm{d_{CICR}}$ spectral matching results for Ocean City cluster C (a green tennis court). The signature has several significant absorptions at ∼0.45, 0.64 and 2.22 μm that are captured by both $\mathrm{d_{CICR}}$ measures, but poorly captured by $\mathrm{d_{CI}}$ measures. Conversely, using CR spectra alone will often yield matches that differ greatly in spectral shape. Figure 2.9 gives the $\mathrm{d_{CR}}$ vs. the $\mathrm{d_{CICR}}$-based matches for cluster signature E (a metal rooftop). These signatures have very similar CR representations, but differ significantly in terms of continua, resulting in unsatisfactory $\mathrm{d_{CR}}$-based matches.

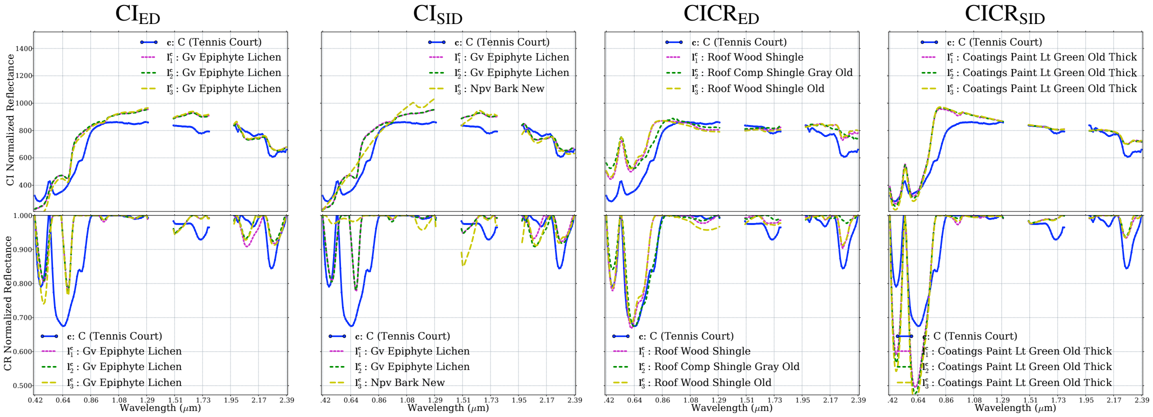


Figure 2.8 : Top: Hit lists of CI signatures for SOM cluster C using $\mathrm{CI_{ED}}$, $\mathrm{CI_{SID}}$, $\mathrm{CICR_{ED}}$ and $\mathrm{CICR_{SID}}$ measures. Bottom: Corresponding CR spectra. Measures using CI signatures can poorly capture differences in absorption bands. The $\mathrm{d_{CICR}}$-based measures can exploit differences in absorption band characteristics, potentially improving material identification capabilities.

Tables 2.2 and 2.3 indicate that the $\mathrm{d_{ED}}$ outperforms the $\mathrm{d_{SID}}$ using the $\mathrm{d_{CICR}}$ measure in terms of visual and $\mathrm{PW^d}$ scores, respectively. To corroborate these results, we had the same four independent observers assign visual scores to an additional set of randomly-selected matches selected from the $\mathrm{CICR_{ED}}$ and $\mathrm{CICR_{SID}}$ hit lists. The

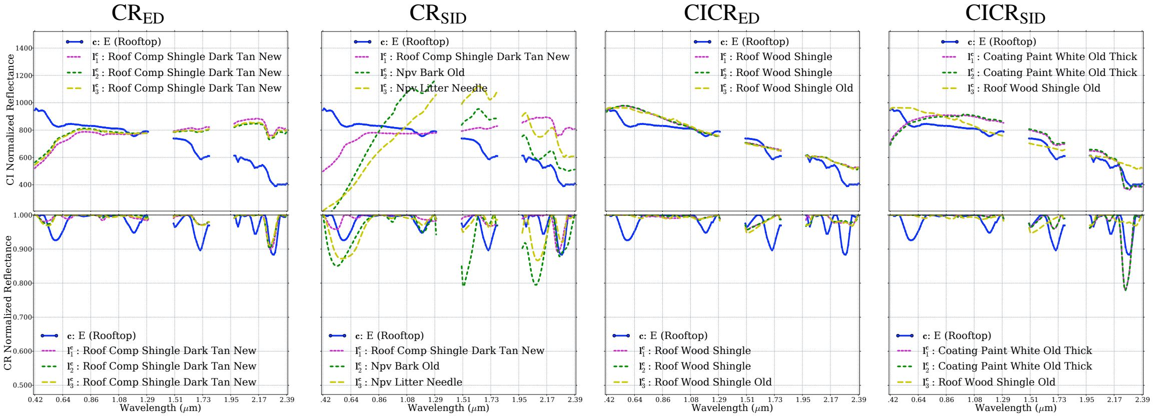


Figure 2.9 : Top: Hit lists for SOM cluster E using $\mathrm{CR_{ED}}$, $\mathrm{CR_{SID}}$, $\mathrm{CICR_{ED}}$ and $\mathrm{CICR_{SID}}$ measures. Bottom: Corresponding CR spectra. Because the CR representation discards information on the shape of the continuum in favor of absorption band characteristics, spectral matching with $\mathrm{CR_{ED}}$ and $\mathrm{CR_{SID}}$ can yield poor results. The $\mathrm{d_{CICR}}$ measures yield improved matches since both the continuum and the absorption features are considered.

	$\mathrm{CICR_{ED}}$			$\mathrm{CICR_{SID}}$		
observer	$\mathrm{n_{ranked}}$	**mean**	**std.**	$\mathrm{n_{ranked}}$	**mean**	**std.**
1	80	1.7000	0.7649	88	1.4545	0.7371
2	10	2.1000	0.9434	3	1.6667	0.4714
3	158	1.4177	1.0139	172	1.3895	0.9791
4	17	1.1176	0.8319	19	1.0526	0.7591
mean	66.25	1.5838	0.8885	70.5	1.3908	0.7367

Table 2.4 : Average, per-observer visual scores of the n_{ranked} spectral matches randomly selected from the $\mathrm{CICR_{ED}}$ and $\mathrm{CICR_{SID}}$ hit lists.

improved performance by $\mathrm{CICR_{ED}}$ over $\mathrm{CICR_{SID}}$ is confirmed by Table 2.4, which gives the mean and standard deviation of the visual scores per-observer, along with the number of matches they ranked (n_{ranked}) for each of the $\mathrm{d_{CICR}}$ measures.

Table 2.5 gives the WSRT p-scores for the $\mathrm{PW^d}$ for each similarity measure, evaluated on the 35 SOM clusters. Larger values (shown in bold text) indicate that the distributions of $\mathrm{d_{CR}}$ and $\mathrm{d_{CICR}}$ similarity values differ (i.e., low statistical significance), and therefore should not be directly compared. The p-scores are also relatively high

	$\mathbf{CI_{ED}}$	$\mathbf{CR_{ED}}$	$\mathbf{CICR_{ED}}$	$\mathbf{CI_{SID}}$	$\mathbf{CR_{SID}}$	$\mathbf{CICR_{SID}}$
$\mathbf{CI_{ED}}$	0.0000	**0.1542**	0.0885	0.0000	0.0200	0.0797
$\mathbf{CR_{ED}}$	**0.1542**	0.0000	**0.7064**	0.0007	0.0034	0.0238
$\mathbf{CICR_{ED}}$	0.0885	**0.7064**	0.0000	0.0001	0.0004	0.0036
$\mathbf{CI_{SID}}$	0.0000	0.0007	0.0001	0.0000	**0.1957**	0.0769
$\mathbf{CR_{SID}}$	0.0200	0.0034	0.0004	**0.1957**	0.0000	**0.6465**
$\mathbf{CICR_{SID}}$	0.0797	0.0238	0.0036	0.0769	**0.6465**	0.0000

Table 2.5 : WSRT-based p-values for the $\mathrm{PW^d}$ using $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$ and $\mathrm{d_{CICR}}$ similarity measures for the 35 Ocean City SOM cluster signatures. Significantly higher p-values between CI and CR-based similarity measures indicate that the similarity values produced by these measures do not follow the same distribution.

between $\mathrm{d_{CI}}$ and $\mathrm{d_{CR}}$ measures, since spectra that are very different in terms of continuum shape can be identical after continuum removal (as shown by the $\mathrm{d_{CR}}$ hit lists in Figure 2.9, for instance). Because similar signatures produce $\mathrm{PW^d}$ scores near 1.0, the $\mathrm{d_{CR}}$ measures appear more discriminatory, in terms of low $\mathrm{PW^d}$ values, than the other measures. However, as we show in Figures 2.6, 2.8 and 2.9, low $\mathrm{PW^d}$ values do not necessarily indicate that a measure is performing well, as the $\mathrm{PW^d}$ scores for clusters C and E are among the lowest for the clusters and similarity measures we consider.

Conversely, large mean $\mathrm{PW^d}$ scores do not necessarily indicate a similarity measure is performing poorly. Instead, the average of the top m matches may be skewed due to the relative scaling of the similarity values. For instance, consider the rather large $\mathrm{PW^d}$ scores shown in Figure 2.6 for signatures M (vegetation), and, to a lesser degree, T (asphalt) and Y (sand). A closer look at the similarity and pairwise $\mathrm{PW^d}$ scores for the hit list of cluster signature M are given in Table 2.6. In this case (and similarly with signatures T and Y), a single similarity score is relatively distant from the remaining two scores, resulting in relatively high $\mathrm{PW^d}$ values. Note that each of the $\mathrm{d_{ED}}$ and $\mathrm{d_{SID}}$-based $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$ and $\mathrm{d_{CICR}}$ similarity measures returns a different hit list, yet we observe the same effect on the $\mathrm{PW^d}$ values. This suggests that the $\mathrm{PW^d}$

could potentially be used as an indicator that a given library contains fewer than m suitable match candidates for a particular signature.

$\mathbf{d}(\cdot,\cdot)$	$\mathbf{M}, \mathbf{l}_1$	$\mathbf{M}, \mathbf{l}_2$	$\mathbf{M}, \mathbf{l}_3$
$\mathbf{CI}_{\mathbf{ED}}$	**27.097** (shaded concrete)	51.683 (green paint)	51.683 (grass)
$\mathbf{CR}_{\mathbf{ED}}$	**94.866** (grass)	137.969 (tall grass)	138.628 (gray shingle)
$\mathbf{CICR}_{\mathbf{ED}}$	**165.857** (grass)	217.640 (shaded concrete)	222.888 (sage brush)
$\mathbf{CI}_{\mathbf{SID}}$	**0.187** (shaded concrete)	0.5817 (green paint)	0.610 (green paint)
$\mathbf{CR}_{\mathbf{SID}}$	**0.052** (grass)	0.107 (palm tree)	0.110 (green paint)
$\mathbf{CICR}_{\mathbf{SID}}$	**0.309** (shaded concrete)	0.793 (green paint)	0.844 (green paint)

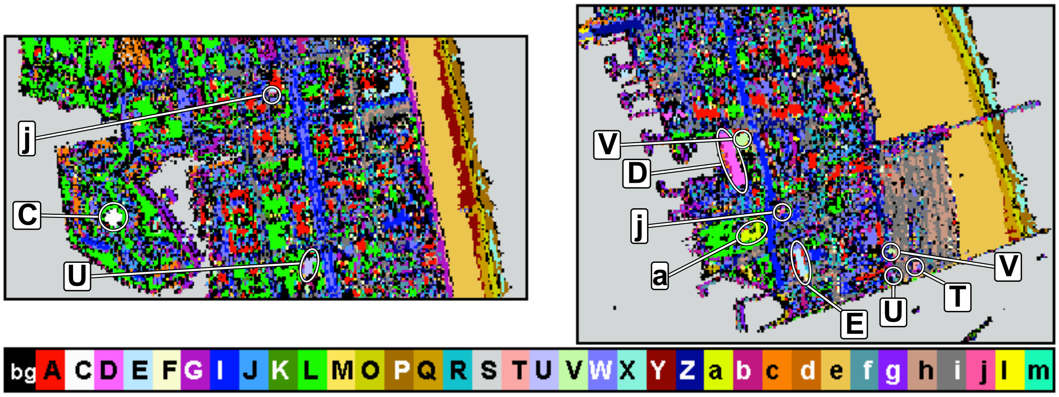
$\mathbf{PW}^{\mathbf{d}}$	$\mathbf{M}, \mathbf{l}_1, \mathbf{l}_2$	$\mathbf{M}, \mathbf{l}_1, \mathbf{l}_3$	$\mathbf{M}, \mathbf{l}_2, \mathbf{l}_3$
$\mathbf{CI}_{\mathbf{ED}}$	1.907	1.954	**1.025**
$\mathbf{CR}_{\mathbf{ED}}$	1.454	1.461	**1.004**
$\mathbf{CICR}_{\mathbf{ED}}$	1.312	1.343	**1.024**
$\mathbf{CI}_{\mathbf{SID}}$	3.114	3.266	**1.049**
$\mathbf{CR}_{\mathbf{SID}}$	2.026	2.080	**1.026**
$\mathbf{CICR}_{\mathbf{SID}}$	2.565	2.731	**1.065**

Table 2.6 : Distance (top table) and PW$^{\mathrm{d}}$ (bottom table) values for the three most similar library signatures to cluster signature M using each distance measure. Significantly higher PW$^{\mathrm{d}}$ scores are due to ambiguity between cluster signature $\mathbf{M}$ and library signatures $\mathbf{l}_2$ and $\mathbf{l}_3$ (bottom, bold), and a strong match to $\mathbf{l}_1$.

Another example where the PW$^{\mathrm{d}}$ may not reliably capture the accuracy of a given similarity measure is illustrated in a case described by van der Meer in [van der Meer, 2006], Figure 7. The author concludes that the $\mathrm{d}_{\mathrm{SID}}$ is more effective than the d_{ED} (SAM), according partly to an analysis of the PW$^{\mathrm{d}}$. The author considers both a synthetic data set consisting of 601-band field-measured spectra, and on AVIRIS imagery consisting of 50 bands in the 2.0 to 2.5 μm range, for material signatures montmorillonite (mont), kaolinite (kaol), quartz and alunite (alun). While our observations do show slight improvement in terms of visual scores over the $\mathrm{d_{ED}}$ when matching the Ocean City signatures, the $\mathrm{d_{SID}}$ performance is worse than the $\mathrm{d_{ED}}$ with the $\mathrm{d_{CR}}$ measures, and nearly equivalent with the $\mathrm{d_{CICR}}$ measures. Furthermore, in terms of mean $\mathrm{PW^d}$ scores, we see that the $\mathrm{d_{SID}}$-based measures appear less discriminatory than $\mathrm{CI_{ED}}$, $\mathrm{CR_{ED}}$, and $\mathrm{CICR_{ED}}$. The $\mathrm{d_{ED}}$ vs. $\mathrm{d_{SID}}$ $\mathrm{PW^d}$ values for alun-kaol, alun-mont, and kaol-mont, with quartz as the reference signature. However, this example is a somewhat pathological case for the $\mathrm{d_{ED}}$ because the alunite, kaolinite, and montmorilionite signatures are nearly equidistant from the quartz reference signature, thereby yielding high $\mathrm{PW^d}$ values for these three signatures.

2.4.5 Evaluation of Automatic Material Identification Results

We now provide an evaluation of whether the spectral matches produced by the best-performing measure, $\mathrm{CICR_{ED}}$, correspond to appropriate material labels. We categorize the spectral library into ten distinct material groups (loosely based on the taxonomy of urban materials given in [Herold et al., 2004]): Concrete materials, Asphalts, Composites (which largely consist of shingle materials), Metals, Vegetation, Coatings (i.e., paint), miscellaneous roofing materials (e.g., tile and wood shingles), Soil/Dirt, Water, and “Other” (“Other” refers to library signatures for which no material information is provided. In our library, this includes only the tennis and basketball court signatures). If the material group of the matching library signature corresponds well to the material group of the cluster signature, we consider the label assignment a success. For some cases, determining this correspondence requires the translation of an object label (for instance, “rooftop”) to a material group (“asphalt”) based on manual inspection of the cluster signature and expert interpretation, since the expert interpretations are sometimes given on the object, rather than on the material level.


j
C
U
V
D
j
a
E
U
T
V
bg A C D E F G I J K L M O P Q R S T U V W X Y Z a b c d e f g h i j l m

c	Expert Interpretation	Matched Library Material
A	Rooftop	Roof Comp Shingle Gray New
C	Tennis Court	Roof Wood Shingle
D	Rooftop	Roof Comp Shingle Gray New
E	Rooftop	Roof Wood Shingle
F	Unknown*	Paved Road Asphalt New
G	Rooftop	Roof Comp Shingle Dark Tan New
I	Road/Park/Walk	Paved Road Asphalt New
J	Road/Park/Walk	Paved Parking Lot Oil Old
K	Vegetation	Green Dry Mixed Grass
L	Vegetation	Paved Sidewalk Concrete New Shade
M	Vegetation	Green Dry Mixed Grass
O	Sand (Beach)	Dry Long Grass
P	Sand (Beach)*	Roof Wood Shingle
Q	Sand (Beach)	Soil
R	Road/Park/Walk	Paved Road Seal New
S	Water*	Roof Tile
T	Parking Lot	Paved Road Seal New
U	Rooftop	Concrete Rooftop

c	Expert Interpretation	Matched Library Material
V	Mini Golf/Rooftop	Roof Comp Shingle Lt Gray New
W	Road/Park/Walk	Paved Parking Lot Oil New
X	Water Fountain*	Roof Wood Shingle
Y	Sand (Beach)	Paved Sidewalk Concrete New
Z	Road/Park/Walk	Paved Parking Lot Asphalt Old
a	Rooftop*	Roof Comp Shingle Red
b	Rooftop	Paved Road Asphalt Old
c	Water/Rooftop*	Roof Metal Green Paint New
d	Unknown*	Roof Tile
e	Sand (Beach)	Paved Parking Lot Oil Old
f	Rooftop	Roof Comp Shingle Mixed New
g	Road/Park/Walk	Roof Tile
h	Road/Park/Walk	Paved Road Asphalt New
i	Road/Park/Walk	Paved Road Seal New
j	Water Tower	Coating Paint White Old Thick
l	Rooftop	Roof Comp Shingle Gray Old
m	Rooftop	Roof Comp Shingle Lt Gray New

Figure 2.10 : Automatic labeling results for all Ocean City cluster signatures. Cluster interpretations (from field knowledge) are given in black text (column 2) and the corresponding best match using the CICR$_{\mathrm{ED}}$ measure is given in column 3 (colored text). Cluster interpretations marked with an asterisk do not have representative material signatures in the spectral library, and are not included in the "Selected" measurements in Table 2.1. Matches in green text (in column 3) indicate that the material of the best library match corresponds well to the expert interpretation, red text indicates a mismatch, and black text indicates that the material composition for the cluster signature is unknown. Spectral matches are discussed in detail in Figures 2.11 to 2.15.

The automatic labeling results for each Ocean City SOM cluster using the CICR$_{\mathrm{ED}}$ measure are given in Figure 2.10. The CICR$_{\mathrm{ED}}$ measure successfully labeled 21 of the 25 clusters with adequate library representation. These 21 clusters comprise 67.6% of the image pixels with known material labels available in the library. Expert interpretations of clusters are given in plain text (column 2), and the $\mathrm{CICR}_{\mathrm{ED}}$ library matches are given in colored text (column 3). The text is colored green if the match is considered a success according to our library categorization, while mismatches are colored red. Clusters without clear expert interpretations of their materials are displayed in black text. Clusters with an asterisk by the expert interpretation (in column 2) lack representative material signatures in the spectral library; therefore these matches should be disregarded. Selected spectral matches, grouped according to their best matching library material label, are given in Figures 2.11 to 2.15. Even within these categories, there are often significant differences in spectral shape for similar materials. However, since our spectral library is sufficiently diverse, we find relevant matches in almost all cases.

Not surprisingly, incorporating CR signatures in the $\mathrm{CICR}_{\mathrm{ED}}$ measure does not improve discrimination between materials without significant absorption features. We observe this in the $\mathrm{CI}_{\mathrm{ED}}$ vs. the $\mathrm{CICR}_{\mathrm{ED}}$ matches for the asphalt signatures shown in Figure 2.11. The best library matches using both measures are the same, but with different ranking orders. Also, the visual scores (in Table 2.1) for the asphalt (h, i, T) and composite (G, I) signatures remain the same for both the d_{CI} and $\mathrm{d}_{\mathrm{CICR}}$ measures.

Two of the concrete matches are of particular interest. First, cluster signature L (Figure 2.13) is matched to a “shaded concrete” library signature. This library signature is described in detail in [Herold et al., 2004]), and is an example of an “intimate” mixture [Clark and Rousch, 1984] of concrete shaded by a tree canopy. The mixture of the flat concrete library signature does not significantly perturb the vegetation library signature and thus appears representative of vegetation. Consequently, after L^2

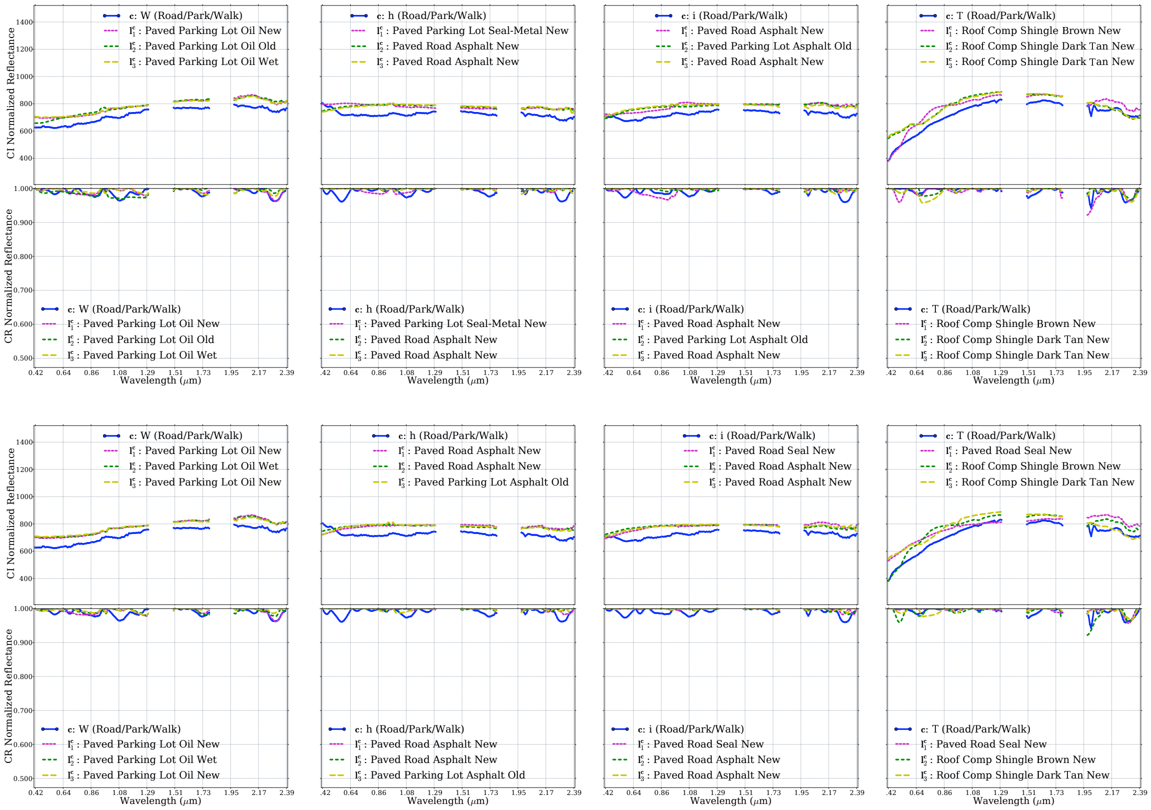


Figure 2.11 : Top: CI and corresponding CR library matches for category “Asphalt” using the CI_{ED} measure. Bottom: CI and corresponding CR library matches for the same clusters using the CICR_{ED} measure. Both measures yield nearly the same matches due to the lack of prominent absorption features in these signatures.

normalization, this signature matches well to cluster signature L (trees). Considering geometric albedo in post-processing (e.g., by using a similarity measure which considers spectral amplitude, such as [Nidamanuri and Zbell, 2011]) can potentially resolve such ambiguities, as the albedo of the concrete signature will generally differ greatly from the albedo the vegetation signature. The other concrete signature, U, corresponds well to several gray/dark gray-colored rooftop material signatures. According to recent aerial photographs, the smaller U signature (Figure 2.10, right image) is a viewing tower, with a small enclosed building on top, that likely is composed of a concrete roof and concrete base. The larger U signature (Figure 2.10, left image) appears to contain

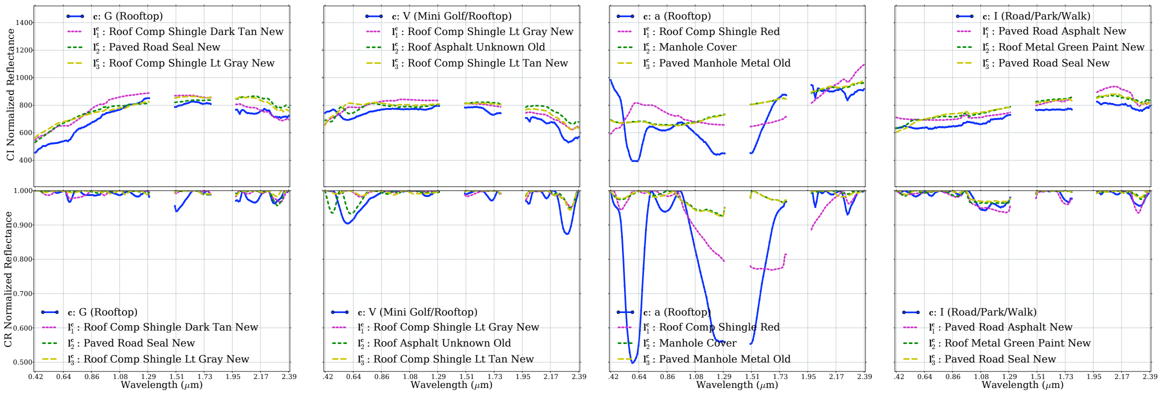


Figure 2.12 : CI and corresponding CR library matches for category "Composites" using CICR$_{\text{ED}}$. As observed by Herold et al. [2004], considerable spectral confusion exists between dark asphalt road and composite shingle rooftop signatures since the composite shingles often have a strong asphalt component, so material matches such as those observed for signature G are expected. The material content of cluster signature V (mini golf/rooftop) is unknown, but the marked similarity to other asphalt signatures suggests it may be dominated by asphalt as well. Signature a (a building rooftop consisting of a mixture of metal alloy and aluminum, painted blue) is a mismatch due to both signatures having dramatically different spectral shapes, indicating that there is not a representative signature present in the library.

concrete roof tiles. Also, since concrete materials generally consist of a mixture of cement, gravel and water, the match to the gravel rooftop is expected.

In some cases, translation between the expert interpretations of image segments and the labels provided in the spectral library is nontrivial. Figure 2.16 illustrates this issue. Here, the expert interpretation of cluster C, “tennis court” material, matches well to several of “wood shingle” library signatures, even though several tennis court material signatures exist in the library. Since the precise material composition of signature C is unknown, and the library metadata, in this case, does not provide a material label for the tennis court signatures, it is difficult to assess the accuracy of this labeling. Here, determining the correct labeling for C requires additional contextual information, since the wood shingle signatures are clearly stronger matches than the tennis court signatures (both in terms of spectral shape and absorption bands).

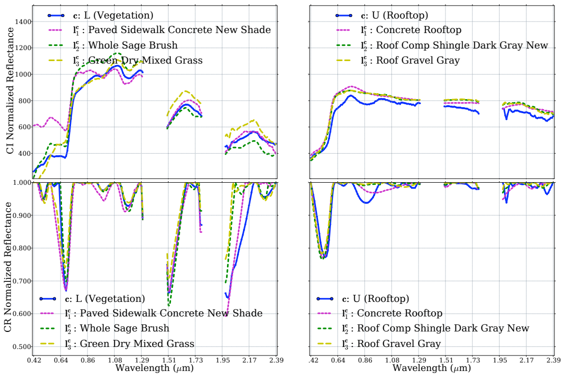


Figure 2.13 : CI and corresponding CR library matches for category ”Concrete” using CICR$_{\mathrm{ED}}$. The “shaded concrete” library signature has spectral shape typical of vegetation, due to intimate mixing effects caused by the shadow of a tree canopy on the concrete material, and closely matches cluster signature L (grass).

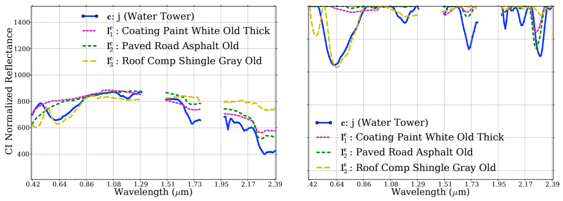


Figure 2.15 : CI and corresponding CR spectra for category “Coatings” using CICR$_{\mathrm{ED}}$. Cluster $\mathbf{j}$ corresponds to a water tower, painted light blue, for which the best match is a white paint signature.

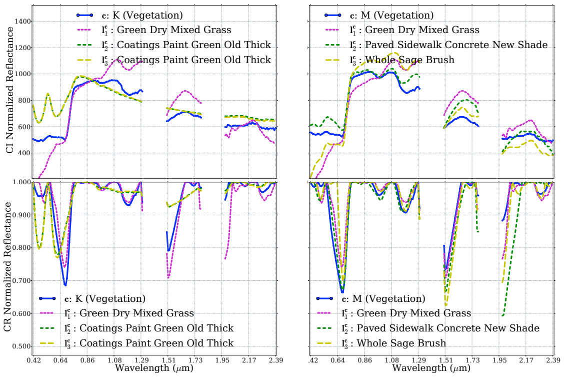


Figure 2.14 : CI and corresponding CR spectra for category “Vegetation” using CICR$_{\mathrm{ED}}$. The second and third “green paint” matches for cluster K are due to strong similarities in absorption features common to vegetation species, as observed in the CR signatures. As a result of these similarities, the measure would incorrectly label the vegetation spectra as green paint if the first match, “grass,” had not been present.

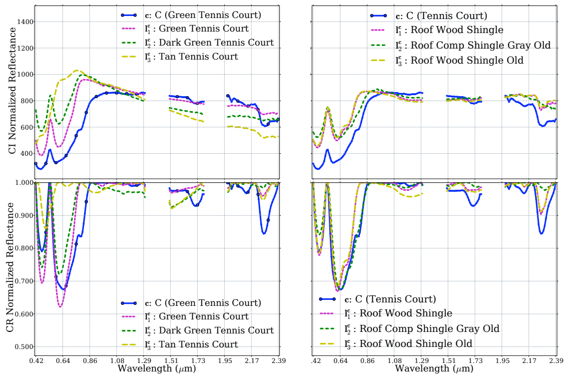


Figure 2.16 : Left plots: Tennis court cluster signature C matched to tennis court library signatures. Right plots: Hit lists for signature C using CICR$_{\mathrm{ED}}$. The “shingle” signatures are better matches, in terms of both CI and CR spectra, than the tennis court signatures.

These ambiguities are best resolved by employing more diverse spectral libraries with extensive metadata, complete with material descriptions.

The spectral matching results on the ISODATA clusters shown in Figure 2.17 exemplify the problem of identifying the materials of mixed or spectrally ambiguous signatures. In this case, ISODATA assigns pixels, corresponding to a clearly recognizable building (SOM cluster D shown in Figure 2.10 and Figure 2.7), into three separate clusters (K, L and M (not to be confused with the SOM clusters with the same labels), none of which represents the true signature of the building. There are two related issues here: (1) ISODATA fails to detect an area of a unique signature clearly delineated by the SOM, and (2) a number of spectrally similar materials correctly grouped together by the SOM are incorrectly assigned to several dissimilar ISODATA clusters. Consequently, these ISODATA clusters give no clear, or worse, misleading interpretations. Matches from a library — while they may be good matches to the mean cluster signatures — may not represent the species at the locations of the incorrectly delineated ISODATA clusters.

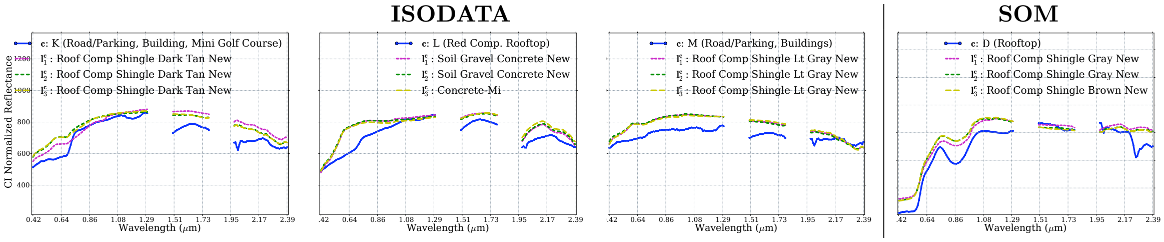


Figure 2.17 : Left three figures: Hit lists for ISODATA clusters K (comprising SOM clusters verified as rooftop shingles, roads/parking areas, and a mini golf course), L (various rooftop materials) and M (various rooftop and road materials). Right figure: Hit list for SOM cluster D (rooftop shingles). In this case, pixels that are delineated well by the SOM cluster D are misclustered by ISODATA into 3 of its clusters — K, L and M — none of which represent the true signature. Note that the spatial distributions as well as the labels of the ISODATA clusters are different from those of the SOM clusters (as shown in [Merényi et al., 2007]).

2.5 Summary and Discussion

In this chapter, we evaluated the performance of the widely-used Euclidean Distance, and the recently-proposed Spectral Information Divergence similarity measures, in terms their capabilities to discriminate between different materials using their CI and CR spectral representations. We began by measuring spectral matching performance according to information-theoretic criteria proposed in [Chang, 2000]. We showed that two of the proposed criteria, the mean PW^{d} and $\mathrm{SDE}^{\mathrm{d}}$, produced redundant rankings of similarity measure performance, and consequently proceeded with a PW^{d}-based analysis. The mean PW^{d} between hit lists of the top $m = 3$ matches for each cluster/reference signature indicated that the d_{ED} produced spectral matches that were more similar to the cluster signature than the $\mathrm{d}_{\mathrm{SID}}$. However, we showed that such criteria can be unreliable estimators of similarity measure performance. Specifically, small mean PW^{d} values indicate that the measure cannot distinguish between the reference signature and the hit list signatures. While this is a desirable property when the hit list signatures represent the same phenomena as the reference signature, it is quite undesirable when the hit list signatures represent different phenomena. Furthermore, large mean PW^{d} values may become skewed when less than m representative signatures are available for a given reference signature. Consequently, criteria such as the PW^{d} do not provide reliable estimates of material identification performance.

Due to the unreliability of the information-theoretic criteria, we manually assessed the quality of spectral matches by scoring them in terms of their visual similarity. Here, $\mathrm{d}_{\mathrm{SID}}$ slightly outperformed d_{ED} using CI signatures, but performed slightly worse than d_{ED} using CR signatures. The spectral matches on CI signatures suggest that using the $\mathrm{d}_{\mathrm{SID}}$ may be advantageous for identifying the materials of smoothly-varying spectral signatures, but may be at a disadvantage for jagged or discontinuous signatures. Further evaluation is necessary to confirm this hypothesis, but a number of studies observed similar results [Du and Chang, 2001; Du et al., 2004; Sobhan, 2007; Tarabalka et al., 2009a].

Based on our results using the d_{CI} and d_{CR}-based measures, we proposed a new similarity measure, $\mathrm{d}_{\mathrm{CICR}}$, which combines both the CI and CR distance measurements to account for differences in both spectral shape and absorption features. We showed improved matching accuracy using $\mathrm{d}_{\mathrm{CICR}}$ using both the d_{ED} and $\mathrm{d}_{\mathrm{SID}}$, with the d_{ED} outperforming the $\mathrm{d}_{\mathrm{SID}}$ in this case. We suspect that the improvement in performance by d_{ED} is a result of two factors: first, normalizing the d_{CI} and d_{CR} components of the $\mathrm{d}_{\mathrm{CICR}}$ distance by the variances of distances between spectral signatures may not properly balance the CI and CR terms for the $\mathrm{d}_{\mathrm{SID}}$ measure, whose distances vary more significantly than the d_{ED} (as indicated by the range of PW^{d} scores in Figure 2.6); and second, Table 2.5 indicates that the CI and CR distances for the $\mathrm{d}_{\mathrm{SID}}$ measure are more strongly correlated than they are for the d_{ED}, as evidenced by the smaller p-values for the ($\mathrm{CI}_{\mathrm{ED}}/\mathrm{CR}_{\mathrm{ED}}$) vs. the ($\mathrm{CI}_{\mathrm{SID}}/\mathrm{CR}_{\mathrm{SID}}$) measures. Consequently, the $\mathrm{d}_{\mathrm{SID}}$ CI vs. CR distances are more redundant with respect to each other than the d_{ED} distances, and thus less information is gained by combining them, an effect also observed by Lee et al. [2010].

Using the best-performing similarity measure, $\mathrm{CICR}_{\mathrm{ED}}$, we successfully identified the materials of 21 of the 25 SOM clusters with known material interpretations and representative library signatures. The remaining ten clusters could not be identified because either their material interpretations were unknown, or the library lacked representative material signatures for those clusters. Both of these issues could potentially be mitigated by augmenting the spectral library with additional, detailed metadata describing the exact material composition of all library spectra, or by including additional library spectra which include such metadata.

Our results show that capturing both the shape of the spectral continuum and the positions/widths of absorption bands is essential to accurately measure similarity between hyperspectral signatures, but the relative importances of these characteristics are data dependent. We expect to achieve improved performance by selecting α using an optimization procedure according to characteristics of input data. Furthermore, we can potentially improve the $\mathrm{d}_{\mathrm{CICR}}$ similarity measure by substituting measures into that apply data-dependant weightings to individual spectral bands such as the Mahalanobis [1936] distance, or measures that capture functional characteristics of spectral signatures such as the Sobolev [1963] distance. We investigate such measures in detail in Chapters 3 and 4.

The presence of mixed spectral signatures representing multiple distinct materials significantly complicates precise material identification. In spectral matching, a representative signature must exist in the library to properly identify the material species of an unlabeled signature. Thus, as we observed with the ISODATA cluster signatures and also in cases of intimate mixing (e.g., the shaded concrete library signature shown in Figure 2.13), the spectra we seek to identify have no clear material interpretations, and the resulting spectral matches are also, inevitably, inaccurate or misleading. Regrettably, automatically identifying all of the material constituents in a given image is limited by the availability of representative labeled spectra, and the lack of exhaustive and detailed ground-truth data makes the objective evaluation of automated labeling methods challenging. Since it is currently not feasible to acquire exhaustive material labels for large remote sensing surveys, synthetically-generated hyperspectral imagery may be of significant help. Alternatively, we can potentially use labeled spectra from other analyses of similar imagery when representative ground-truth spectra are not available. However, reconciling differences caused by differing capture conditions is necessary to use such data. We provide an evaluation of automated material identification techniques using both synthetic data, along with real image data from similar analyses, in Section 5.1.

2.A Appendix: Continuum-Removal Algorithm

Algorithm 2.1 RemoveContinuum

```
Input: Spectrum $\mathbf{x}$, wavelengths $\mathbf{w}$, number of bands $n$
Output: Continuum removed spectrum $\mathbf{cr}$, estimated continuum curve $\mathbf{cc}$
$\mathbf{cr} = \mathbf{1}^n$, $\mathbf{cc} = \mathbf{x}$, $\mathbf{h} = \mathbf{0}^n$, $\mathbf{h}' = \mathbf{1}^n$
for $i \in [1, n-1]$ do
    $\mathbf{h}_i = I(\mathbf{x}_i > \mathbf{x}_{i+1} \ \& \ \mathbf{x}_{i-1} < \mathbf{x}_i)$  # Initial hull = inflection points
end for
while $\mathbf{h}' \neq \mathbf{h}$ do
    $\mathbf{h} = \mathbf{h}'$                          # Update hull until no further changes
    $\mathbf{h}' = \text{SweepContinuum}(\mathbf{x}, \mathbf{w}, \mathbf{h}, n)$
end while
$\mathbf{h}_0 = \mathbf{h}_n = 1$                       # Endpoints always on hull
for $i \in [1, n]$ do
    if $h_i = 1$ then
        $j = i + 1$
        while $h_j = 1$ do
            $j = j + 1$                     # Find last band of current absorption feature
        end while
        $s = (\mathbf{x}_j - \mathbf{x}_i)/(\mathbf{w}_j - \mathbf{w}_i)$
        for $k \in [i, j]$ do
            $\mathbf{cc}_k = \mathbf{x}_i + (\mathbf{w}_k - \mathbf{w}_i)$
            $\mathbf{cr}_k = 1 - (\mathbf{x}_k/\mathbf{cc}_k)$         # $\mathbf{cr}_k > 0 \implies$ absorption feature
        end for
        $i = j$
    end if
end for
```

Algorithm 2.2 SweepContinuum

```
Input: Spectrum $\mathbf{x}$, wavelengths $\mathbf{w}$, current hull $\mathbf{h}$, number of bands $n$
Output: Updated hull $\mathbf{h}$
$l = 0$, $r = 1$
while $r < n$ do
    if $\mathbf{h}_r = 1$ then
        $s = (\mathbf{x}_r - \mathbf{x}_l)/(\mathbf{w}_r - \mathbf{w}_l)$
        for $j \in [l, r]$ do
            $\mathbf{h}_j = I(\mathbf{x}_j < \mathbf{x}_l + s(\mathbf{w}_j - \mathbf{w}_l))$
        end for
        $l = r$
    end if
    $r = r + 1$
end while
```

Part II

Adaptive Similarity Measures for Intra-domain Material Identification

Chapter 3

Hybrid Similarity Measures

Portions of this chapter are based upon the following publications:

- BD Bue, E Merényi, and B Csathó. "Automated Labeling of Segmented Hyperspectral Imagery via Spectral Matching". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* [Aug. 2009].
- BD Bue, E Merényi, and B Csathó. "Automated Labeling of Materials in Hyperspectral Imagery". *IEEE Trans. on Geoscience and Remote Sensing* 48.11 [2010], pp. 4059–4070.
- BD Bue and E Merényi. "An Adaptive Similarity Measure for Classification of Hyperspectral Signatures". *IEEE Geoscience and Remote Sensing Letters* 10.2 [2012], pp. 381–385.

We demonstrated in Chapter 2 that capturing both the shape of the spectral continuum and the positions/widths of absorption bands is essential to accurately measure similarity among hyperspectral signatures. However, the relative importances of these characteristics are data dependent. In this chapter* we demonstrate a technique to learn a convex weighting among several distinct similarity measures using a technique based on Linear Discriminant Analysis (LDA). We evaluate the performance of our adaptive CICR measure on AVIRIS spectra sampled from a well-studied urban scene and show that our technique yields improved classification accuracy in comparison to classification using CI or CR Euclidean distance measurements alone. Our LDA-based measure also yields competitive performance to brute-force computation of the CI vs. CR weight parameter, at much reduced computational cost. As we discussed earlier, a close relationship exists between finding a good set of features and choosing a good

*This work was done in collaboration with Erzsébet Merényi, with assistance from David Thompson, Kiri Wagstaff, Devika Subramanian, Bea Csathó, and Marika Kästner.

similarity function. Consequently, we demonstrate that classifying spectra using a classifier equipped with the CICR measure gives comparable or better results than several conventional feature selection and dimensionality reduction techniques. We then generalize our technique to exploit the functional nature of spectral data by calculating the weighted relevances of spectral derivates – i.e., derivatives of a spectral signature with respect to wavelength – using the Sobolev distance measure. We compare with the classification accuracy of the adaptive Sobolev measure to classification using per-derivate Euclidean distances on the Ocean City AVIRIS image described in Chapter 2. We provide an in-depth analysis of the empirical and asymptotic behavior of the Sobolev measure, and show improved performance over the Euclidean baseline when the higher-order derivatives of spectral signatures are uncorrelated.

3.1 The Adaptive CICR Measure

In this section, we present an adaptive version of the CICR similarity measure that automatically calculates a convex weighting between similarity measurements of Continuum Intact (CI) and Continuum Removed (CR) signatures. To achieve this, we reformulate the $\mathrm{d}_{\mathrm{CICR}}$ (Equation (2.2)) similarity measure as follows:

$$\mathrm{d}_{\mathrm{CICR}}(\mathbf{x}_i, \mathbf{x}_j, \alpha) = (1-\alpha)\mathrm{d}_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) + \alpha \mathrm{d}_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) \tag{3.1}$$

where

$$\mathrm{d}_{\mathrm{CI}}(\mathbf{x}_i, \mathbf{x}_j) = \left\| \frac{\mathbf{x}_i}{\|\mathbf{x}_i\|} - \frac{\mathbf{x}_j}{\|\mathbf{x}_j\|} \right\| \tag{3.2}$$

$$\mathrm{d}_{\mathrm{CR}}(\mathbf{x}_i, \mathbf{x}_j) = \left\| \frac{\mathrm{CR}(\mathbf{x}_i)}{\|\mathrm{CR}(\mathbf{x}_i)\|} - \frac{\mathrm{CR}(\mathbf{x}_j)}{\|\mathrm{CR}(\mathbf{x}_j)\|} \right\| \tag{3.3}$$

Here, $\alpha \in [0, 1]$ is a weighting parameter, and $\mathrm{CR}(\cdot)$ performs continuum removal. We estimate the continuum of a given spectrum by fitting a piecewise linear function to local maxima using the procedure described in Algorithm 2.1. Observations on the continuum are assigned values of zero, and absorption features (observations between local maxima) are assigned values in the $[0, 1]$ range, proportional to their relative distance from the estimated continuum. Because the continuum removal procedure is sensitive to spurious local maxima, we smooth each signature using a moving average filter before performing continuum removal. Although smoothing can mask small absorption features, such features are often close to the noise floor of the sensor, and we accept this loss in specificity in favor of noise reduction. In our experiments using AVIRIS data, smoothing windows ranging from three to five bands (0.03-0.05μm) have done well.

The $\mathrm{d}_{\mathrm{CICR}}$ measure described given above differs from our original formulation given by Equation (2.2) in two respects. First, the convex combination of CI and CR terms yields more consistent performance than applying α to only the CR term. Second, due to the nature of continuum estimation, CR signatures contain many values near zero, which provides little discriminating information among signatures when combined with the CI distance measure. We observed experimentally that scaling the CR signatures by their L^2 norms provides a greater degree of contrast between classes by allowing the most prominent absorption features to play a greater role in discrimination, in comparison to normalizing by the variance of CR distances (as we described in Chapter 2). Additionally, L^2 normalization has the benefit of mapping the CR signatures to the same range as the CI signatures, which enables fine-tuning of the weight parameter α according to input data. However, we note that L^2-normalization can exacerbate noise when CR signatures contain spurious absorption features, whereas a “global” scaling factor (such as the variance of CR distances) does not accentuate noise on individual signatures.

3.1.1 LDA for Hybrid Similarity Measures

Figure 3.1 gives an overview of the methodology we use to calculate the weight parameter α in Equation (3.1). Given a set of N vectors $\{\mathbf{x}_i\}_{i=1}^{N}$, $\mathbf{x}_i \in \mathbb{R}^n$ belonging to K classes, with labels $y_i \in [1, K]$, we calculate α using a method inspired by linear discriminant analysis (LDA) ([Fisher, 1936; Rao, 1948]). LDA computes the vector $\mathbf{w}$ that maximizes the Rayleigh quotient (using the formulation given in [Hastie et al., 2011])

$$S = (\mathbf{w}^T \mathbf{M}_B \mathbf{w})(\mathbf{w}^T \mathbf{M}_W \mathbf{w})^{-1}, \quad (3.4)$$

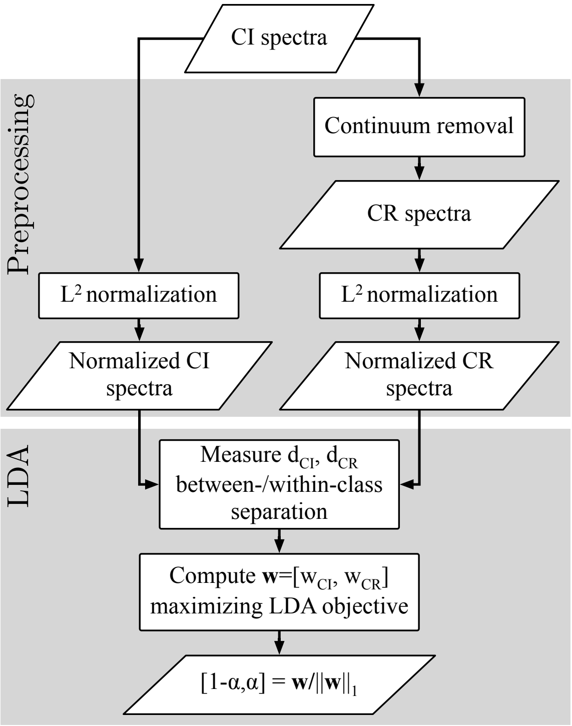


Figure 3.1 : Processing steps for calculating $\mathrm{d}_{\mathrm{CICR}}$ weight parameter α.

where $\mathbf{M}_B$ and $\mathbf{M}_W$ are (symmetric, positive-definite) between-class separation and within-class scatter matrices. We form the $\mathbf{M}_B$ and $\mathbf{M}_W$ matrices according to the capabilities of each of the $\{\mathrm{d}_{\mathrm{CI}}, \mathrm{d}_{\mathrm{CR}}\}$ measures in separating the given classes.

$$\mathbf{M}_B = \begin{bmatrix} s_b(\mathrm{d}_{\mathrm{CI}}, \mathrm{d}_{\mathrm{CI}}) & s_b(\mathrm{d}_{\mathrm{CR}}, \mathrm{d}_{\mathrm{CI}}) \\ s_b(\mathrm{d}_{\mathrm{CI}}, \mathrm{d}_{\mathrm{CR}}) & s_b(\mathrm{d}_{\mathrm{CR}}, \mathrm{d}_{\mathrm{CR}}) \end{bmatrix}, \quad (3.5)$$

$$\mathbf{M}_W = \begin{bmatrix} s_w(\mathrm{d}_{\mathrm{CI}}, \mathrm{d}_{\mathrm{CI}}) & s_w(\mathrm{d}_{\mathrm{CR}}, \mathrm{d}_{\mathrm{CI}}) \\ s_w(\mathrm{d}_{\mathrm{CI}}, \mathrm{d}_{\mathrm{CR}}) & s_w(\mathrm{d}_{\mathrm{CR}}, \mathrm{d}_{\mathrm{CR}}) \end{bmatrix}, \quad (3.6)$$

where $s_b(\mathrm{d}_1, \mathrm{d}_2)$ and $s_w(\mathrm{d}_1, \mathrm{d}_2)$ are the between-class and within-class separation, respectively, according to distance measures d_1 and d_2

$$
\begin{aligned}
s_b(\mathrm{d}_1, \mathrm{d}_2) &= \frac{1}{N} \sum_{j=1}^{K} N_j \mathrm{d}_1(\boldsymbol{\mu}_j, \overline{\boldsymbol{\mu}}) \mathrm{d}_2(\boldsymbol{\mu}_j, \overline{\boldsymbol{\mu}}), && (3.7)\\
&= s_b(\mathrm{d}_2, \mathrm{d}_1) && (3.8)\\
s_w(\mathrm{d}_1, \mathrm{d}_2) &= \frac{1}{N} \sum_{j=1}^{K} \sum_{i: y_i = j} \mathrm{d}_1(\mathbf{x}_i, \boldsymbol{\mu}_j) \mathrm{d}_2(\mathbf{x}_i, \boldsymbol{\mu}_j) && (3.9)\\
&= s_w(\mathrm{d}_2, \mathrm{d}_1). && (3.10)
\end{aligned}
$$

Here, $\left\{\boldsymbol{\mu}_j\right\}_{j=1}^{K}$ are the mean vectors of each of the K classes, $\overline{\boldsymbol{\mu}}$ is the mean of the $\boldsymbol{\mu}_j$, and N_j is the number of samples in class j.

The first (largest) eigenvector of $\mathbf{M}_W^{-1}\mathbf{M}_B$, $\mathbf{w}$, maximizes Equation (3.4), with separation S equal to the corresponding eigenvalue [Hastie et al., 2011]. The components of $\mathbf{w} = [w_{CI}, w_{CR}]$ provide a weighting of the CI and CR distances with good class separation on training data, but is not necessarily convex as we require in Equation (3.1), and may not generalize well to test data. Because Rayleigh quotients are invariant with respect to scaling of $\mathbf{w}$ (i.e., for any $c > 0$, $c\mathbf{w}$ also maximizes Equation (3.4)) [Horn and Johnson, 1985], we scale the components of $\mathbf{w}$ to a convex range by dividing each component by $\|\mathbf{w}\|_1$. This yields the convex pair $\{w_{CI}/\|\mathbf{w}\|_1, w_{CR}/\|\mathbf{w}\|_1\} = \{(1-\alpha), \alpha\}$, as desired.

As Equation (3.4) can become ill-posed, we regularize the within-class scatter matrix via a shrinkage operator:

$$
\mathbf{M}'_W = (1-\gamma)\mathbf{M}_W + \gamma\mathcal{I}, \qquad (3.11)
$$

where $\gamma \in [0, 1]$ is a (convex) regularization parameter, and $\mathcal{I}$ is the (2×2) identity matrix. In practice, we select γ via cross-validation, using the methodology described in the next section.

3.1.2 Complexity Analysis

Calculating α using our LDA-based method is significantly less computationally expensive than a brute force search over the range of α values. Quantitatively, assuming N samples of dimensionality D belonging to K classes, we first compute the continuum-removed representation of each spectrum using our piecewise linear continuum estimation procedure – an $\mathrm{O}(D)$ operation per spectrum. Then, given the set of (pre-computed) class means, a MinDist classifier must compare each signature to each class mean, an $\mathrm{O}(DNK)$ operation. Let α_{LS} be the current α value we consider, and let A be the number of values α_{LS} can take in the $[0, 1]$ range (in this work, we choose A=100). Using brute force search, we apply the $\mathrm{O}(DNK)$ MinDist classifier A times. With the LDA-based method, calculating the (symmetric) $\mathbf{M}_B$ involves three $\mathrm{O}(DK)$ operations and $\mathbf{M}_W$ involves three $\mathrm{O}(DN)$ operations, and calculating the eigendecomposition of the (2×2) $\mathbf{M}_W^{-1}\mathbf{M}_B$ matrix can be done in constant time. This amounts to roughly an A-fold improvement in performance by the LDA-based method over line search. Because A must be large enough to adequately cover the weight parameter space, our method is an order of magnitude faster than brute-force search.

3.1.3 Evaluation Methodology

We compare the performance of the adaptive $\mathrm{d}_{\mathrm{CICR}}$ measure to the d_{CI} and d_{CR} measures using a minimum distance to class means (MinDist) classifier with 5-fold random stratified sampling, using 50% for training and the remaining 50% for testing. In each scenario, we select at most N samples for training in each fold, and use N

samples for testing. When fewer than $2N$ samples are available for a given class, we randomly split the available samples evenly into training and testing sets in each fold. We calculate α by maximizing $\mathrm{S}(\alpha)$ as described in Section 3.1. We compare this α value to the α_{LS} value obtained by line search (LS) on a uniformly spaced range of 100 points, $\alpha_{\mathrm{LS}} \in (0, 1)$, that yields the highest *training* classification accuracy. We report accuracy on only test data according to accuracy $=$ (# of True Positives)/(# of Samples). Accuracies produced via line search are an approximate upper bound on achievable accuracy.

For the scenarios described below, we select the γ with the best classification accuracy on the training set over 10 uniformly spaced values in [0.001,0.1]. We chose this range because smaller γ values tended to yield ill-posed solutions, and larger values did not improve classification accuracy in any of the scenarios we consider – regardless of α. We calculate γ once for each scenario, and use the same value for each cross-validation fold. We also reject any γ values that produce solutions to $\mathbf{M}_W^{-1}\mathbf{M}_B$ with no positive eigenvalues, as such γ yield rank-deficient $\mathbf{M}'_w$ (Equation (3.11)).

3.1.4 Case Study: Ocean City AVIRIS Spectra

The starting point of the work described in this section is a set of reflectance spectra sampled across distinct material species from the AVIRISLA image of Ocean City, MD described in Section 2.4.3. The 35 SOM clusters resulting from [Merényi et al., 2007] guided the extraction of a trustworthy representative subset of spectra for this study, by stratified random sampling across 14 of those 35 clusters for which material identification was unambiguous and which served the methodology design for evaluating the adaptive $\mathrm{d}_{\mathrm{CICR}}$ measure. The experimental design is explained below. For this work the reflectance spectra were extracted from the already pre-processed

Ocean City image.

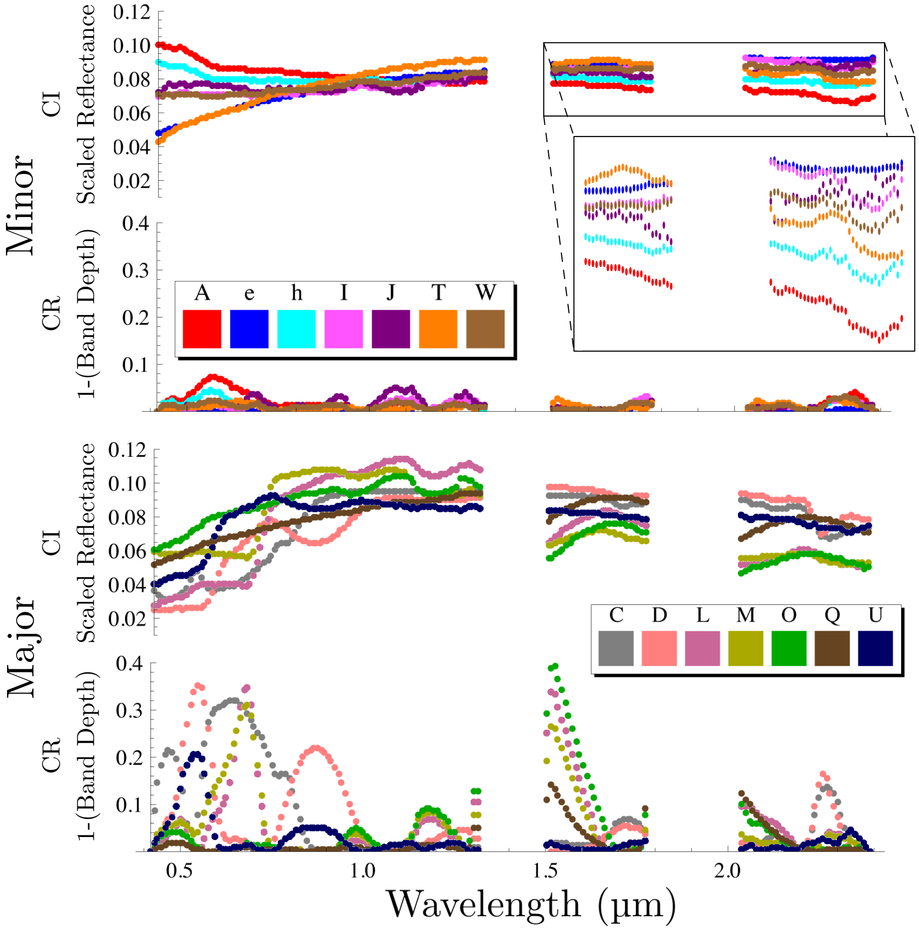


Figure 3.2 : Ocean City CI and CR mean signatures for Minor (top) and Major absorption classes (bottom). Top inset: detail view of Minor absorption signatures, wavelengths 1.5-2.5μm. The disconnected regions near 1.3-1.5 and 1.7-2.0 μm consist of noisy bands removed due to water saturation. CI signatures are scaled by their L^2 norms to compensate for varying illumination conditions.

We examine three different spectral scenarios specifically constructed to contrast the performance of the adaptive d_{CICR} measure. In the first scenario, all samples contain only *minor* absorptions, where we define a “minor” absorption as one with no CR band depths greater than threshold τ; we use $\tau = 0.1$ (10% absorption with respect to the continuum) in this work. In this case, we anticipate similar classification accuracies from the $\mathrm{d_{CICR}}$ and $\mathrm{d_{CI}}$ measures, since the CR signatures lack prominent absorption features (and therefore are flat and uninformative). The classes in this scenario consist of asphalt rooftop materials (class A), roads/parking lots (classes I, J, T, W, and h), and dry beach sand (class e). Figure 3.2 (top) shows the CI and CR mean signatures for these classes. In the second scenario, all signatures contain one or more *major* absorptions, where we define major absorptions as those with CR values greater than τ. Here, we anticipate a more significant boost in accuracy in comparison to the Minor absorption scenario, as the CR signatures are more informative. The subset of data in this scenario consists of vegetation (classes L and M), a tennis court (class C), wet sand (classes O and Q), and composite rooftop materials (classes D and U). Figure 3.2 (bottom) shows the CI and CR mean signatures for these classes. The 7 spectral species in each of the Major and Minor absorption categories are relatively "pure" representatives of their respective species. The last scenario, *Combined*, consists of all classes from both Major and Minor absorption scenarios. We anticipate notable performance gains with the $\mathrm{d_{CICR}}$ measure in this scenario, as both the CI and CR signatures provide information to discriminate between classes.

Figure 3.3 characterizes the relationship between the number of labeled samples provided for training vs. classification accuracy. In each of the three scenarios, the $\mathrm{d_{CICR}}$-based classifiers match or outperform the $\mathrm{d_{CI}}$-based classifier. Additionally, our LDA-based technique for calculating α performs nearly as well ≈0.5-1% difference in accuracy as brute force search when enough training samples (about 50 / class, for these scenarios) are available. We observe the most significant performance gains of the three scenarios in the Combined scenario, where the $\mathrm{d_{CICR}}$ measure can exploit absorption features to separate the classes belonging to the Major and Minor absorption scenarios and also can capitalize on the absorption characteristics of individual classes. $\mathrm{d_{CR}}$ performs the worst in all three cases, and is not shown in Figure 3.3 to emphasize the performance of the other measures.

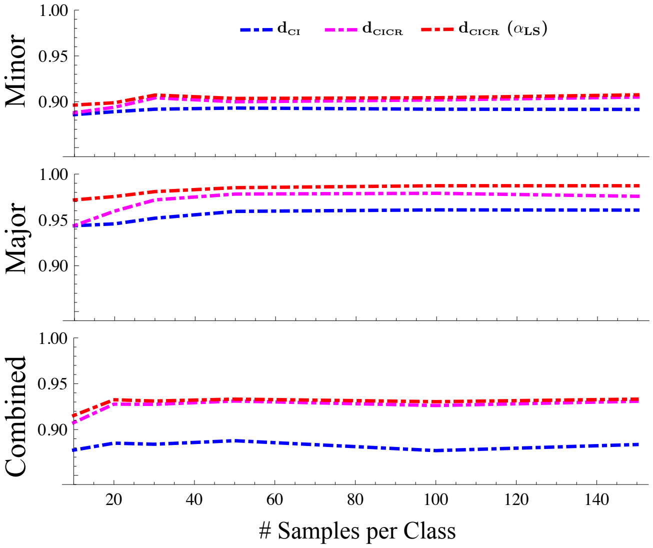


Figure 3.3 : Classification accuracy vs. number of (training) samples per class for Minor absorption (top), Major absorption (middle) and Combined (bottom) scenarios. In each scenario, the LDA-based d_{CICR} measure (magenta line) outperforms the baseline d_{CI}-based classifier (blue line), and with enough training samples (∼20-30, scenario dependant) achieves classification accuracy comparable to line search (red line). d_{CR}-based classification accuracy not shown above due to significantly lower accuracies (∼65-77%) in comparison to the d_{CI} and d_{CICR}-based classifiers.

We now consider a "typical" classification problem consisting of 100 training samples per class. Figure 3.4 gives the overall and per-class classification accuracies for $\alpha \in [0, 1]$. The vertical magenta dashed line marks the α value determined by maximizing Equation (4.2), and the black vertical line gives α_{LS}. Table 3.1 provides average accuracies for each measure. In all three scenarios, small alpha values (< 0.3) yield the highest classification accuracies (though we do not constrain the search to this range). This indicates that, for this data set, CI signatures are more robust descriptors than CR signatures for classification. This is particularly obvious in the Minor absorption scenario (Figure 3.4, top), where the CR signatures lack discriminative features. Here, classification accuracy using d_{CI} is close to d_{CICR}, and both our method and the line search produce α values near zero.

For the Major absorption classes (Figure 3.4, middle), note that the rate of decrease in classification accuracy is less dramatic as α approaches one, by comparison to the Minor (top) and Combined (bottom) absorption scenarios. This indicates that the CR signatures provide additional discriminating information, which increases the α values

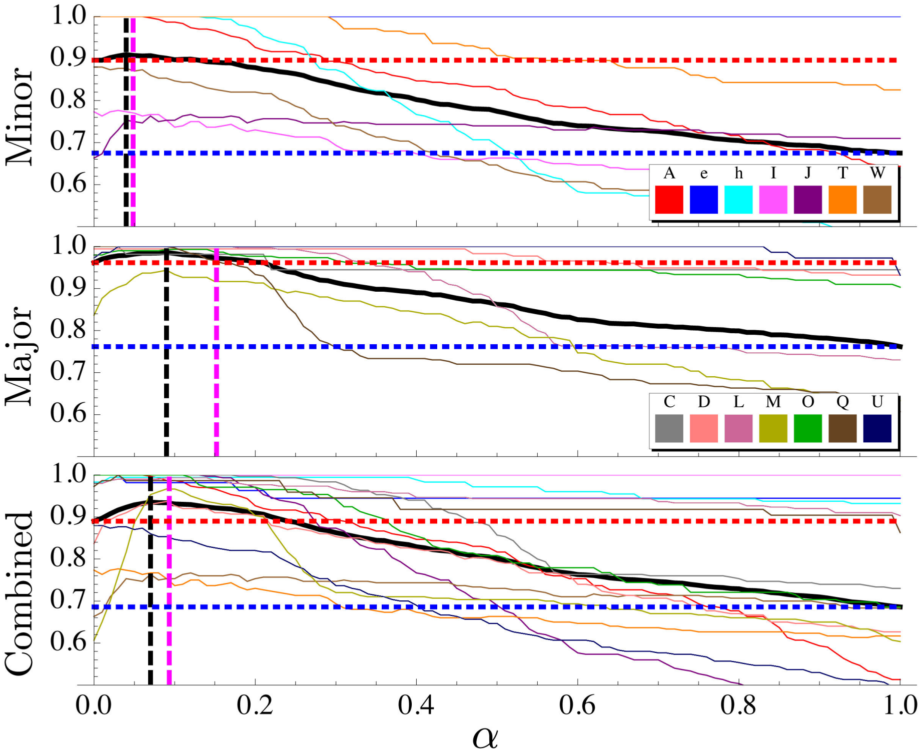


Figure 3.4 : α vs. per-class d_{CICR} classification accuracy for Minor (top), Major (middle) and Combined (bottom) absorption classes. Colored lines indicate per-class accuracies, and the black solid line gives the overall classification accuracy. The black vertical bar gives α_{LS}, and the magenta vertical bar gives α. The horizontal lines give the CI (red, $\alpha = 0$) and CR (blue, $\alpha = 1$) classification accuracies. Because the CI representation is generally more informative than CR, α values tend towards zero, but larger values occur in cases when the CR representation provides additional discrimination information (as in the Major and Combined absorption class scenarios).

yielding higher classification accuracy. Correspondingly, the maximum separation also shifts towards larger α values. Although α and α_{LS} differ the most in this scenario, their corresponding classification accuracies are not far apart (97.4% vs. 98.4%). Both are improvements over the baseline d_{CI} accuracy (1.5% and 2.5% relative improvements for our LDA-based α and α_{LS}, respectively).

In the Combined scenario (Figure 3.4, bottom), due to potentially increased class confusion among signatures (compared to the other two scenarios), locating a compromise between the CI and CR terms is challenging. As we see in Figure 3.4,

	$\mathbf{d_{CI}}$	$\mathbf{d_{CR}}$	$\mathbf{d_{CICR}}$ $(\alpha = 0.5)$	$\mathbf{d_{CICR}}$ $(\alpha \pm \sigma_\alpha)$	$\mathbf{d_{CICR}}$ $(\boldsymbol{\alpha}_{\mathbf{LS}} \pm \sigma_{\boldsymbol{\alpha}_{\mathbf{LS}}})$
Minor	88.5	66.1	66.7	**90.1** (0.0510±0.0375)	90.4 (0.0485±0.0003)
	0.83	1.10	0.72	0.71	0.74
Major	92.9	76.7	81.3	**97.4** (0.1493±0.0041)	98.4 (0.0770±0.0195)
	0.98	1.51	1.67	0.98	0.32
Combined	88.2	66.4	70.6	**91.4** (0.0903±0.0006)	92.6 (0.0670±0.0057)
	1.28	1.12	0.64	0.35	0.78

Table 3.1 : Mean and standard deviation of classification accuracy obtained with each of the $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$, $\mathrm{d_{CICR}}$ measures shown in Figure 3.4. Results using the unweighted $\mathrm{d_{CICR}}$ measure ($\alpha = 0.5$) are also provided. Mean and standard deviation (σ) of α values for $\mathrm{d_{CICR}}$ measures are given in parentheses. The most accurate measure for each scenario (excluding the α_{LS}-based measure) is given in bold text.

the mean classification accuracy for this scenario generally falls between the mean accuracies of the Minor and Major absorption scenarios. However, we see the most significant improvement, over the baseline $\mathrm{d_{CI}}$ method, in classification accuracy in this scenario (4.5%, by comparison of the thick black line to the horizontal red dashed line in Figure 3.4, bottom), vs. the other two scenarios, since both the CI and CR representations provide complimentary information to discriminate the classes. This is noteworthy given that the CI and CR classification accuracies in the Combined scenario are close to those of the Minor absorption scenario (88.5% vs. 88.2% and 66.1% vs. 66.4%, in the Combined vs. Minor scenarios, respectively), yet the relative improvement in the Minor scenario is, not surprisingly, lower (1.7%).

3.2 Comparisons to Related Work

The problem of combining multiple similarity measures is closely related to the problem of combining predictions produced by multiple classifiers. Hansen and Salamon [1990] and later Dietterich [2000] showed that a Combined set of classifiers can be more accurate than the best of the individual classifiers if and only if each classifier produces better than random error (i.e., each classifier is *accurate*) and the classifiers produce uncorrelated errors with respect to one another (i.e., the classifiers are *diverse*). As the accuracy of a classifier is coupled with the quality of the similarity measure used to compare samples, a similar theory holds when combining similarity functions. Lee et al. [2010] empirically illustrated this connection by measuring classification accuracy on separate vs. combined Euclidean distance measures on separate representations of identical samples. They observed that as the distances produced by each measure became less correlated, classification accuracy using the combined measure typically improved.

We may also view a hybrid measure as a form of feature weighting where the features of input samples are scaled according the discriminative capabilities of a particular representation. For instance, the $\mathrm{d}_{\mathrm{CICR}}$ measure may be viewed as a weighted form of the distance measure $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ whose outputs are scaled according the overall discriminative utility of the absorption features. This is similar in many ways to applying a feature selection or dimensionality reduction procedure to determine which spectral features are most relevant to the classification task. Here, we compare our results using the adaptive $\mathrm{d}_{\mathrm{CICR}}$ measure to several conventional feature selection / dimensionality reduction techniques. We consider univariate χ^2_p feature selection, where we select the top $p\%$ of features by discarding statistically independent features according to the χ^2 criterion [Manning et al., 2008, Eq. 136], and Recursive Feature Elimination (RFE, [Guyon et al., 2002]), which iteratively removes features that contribute the least to a decision function of a generalized linear model. We also consider the feature weighting approach where we compute weight vector $\mathbf{w}$ applied to each sample $\mathbf{x}$ as $\mathbf{x} = [w_1 x_1, \ldots, w_n x_n]$ using a L^1-penalized generalized linear model (GLM) ([Guyon and Elisseeff, 2003]). We remove any features whose corresponding weight is zero, thereby reducing the dimensionality of the feature space from n to $n_{\mathbf{w}\neq 0}$, where $n_{\mathbf{w}\neq 0}$ is the number of nonzero values of $\mathbf{w}$. Additionally, we consider the dimensionality reduction techniques principal components analysis (PCA), selecting the top m principal components that explain 99% of the observed variance, and $\mathrm{LDA_{FW}}$, where we map features to a $K-1$ dimensional space using regularized *feature-weighted* Linear Discriminant Analysis (described in detail later in Chapter 4).

We mimic the methodology described in Section 3.1.3, and evaluate the classification accuracy in each of the Ocean City scenarios using five cross-validation folds. As before, we evenly split the data from each scenario into training and test sets, and compute the vector of feature weights $\mathbf{w}$, or, in the case of PCA and LDA, the transformation $T(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^{K-1}$, using the (CI) training set, and then classify the weighted/transformed test spectra using the MinDist classifier. For the RFE, L^1 and $\mathrm{LDA_{FW}}$ algorithms, we select the scalar regularization parameter γ from the set $\{10^{-10}, \ldots, 10^{-2}, 0.2, \ldots, 0.8, 1-10^{-2}, \ldots, 1-10^{-10}\}$ that yields the highest accuracy on the training data.

	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

Table 3.2 : Mean (shaded rows) and standard deviation (unshaded rows) of $\mathrm{d_{CICR}}$ results in comparison to feature selection methods. The best and second-best performing techniques (excluding $\mathrm{d_{CICR}}$ LS) for each scenario are given in red and blue italics, respectively. $\mathrm{LDA_{FW}}$ yields the best overall performance, though $\mathrm{d_{CICR}}$ LDA performs competitively at lower computational cost.

Table 3.2 gives the classification accuracies for the baseline $\mathrm{d_{CI}}$, $\mathrm{d_{CR}}$, and $\mathrm{d_{CICR}}$ measures in comparison to the feature selection and dimensionality reduction techniques. Perhaps unsurprisingly, $\mathrm{LDA_{FW}}$ gives the best overall performance across the three scenarios, as it can exploit discriminative characteristics of individual spectral features. However, the performance of $\mathrm{d}_{\mathrm{CICR}}$ is competetive to $\mathrm{LDA}_{\mathrm{FW}}$, and can be achieved at significantly lower computational cost. Of the remaining feature selection algorithms, only RFE yields an improvement over the baseline d_{CI} measure in the Major absorption scenario, and it is still 3-5% less accurate than both $\mathrm{LDA}_{\mathrm{FW}}$ and $\mathrm{d}_{\mathrm{CICR}}$ with LDA. One caveat is that both RFE and L^1 select features according to the weights of a GLM, and thus, their selected features may be suboptimal for a MinDist classifier. However, this suggests that such techniques are limited in that they are tied to a specific classification technique, whereas $\mathrm{d}_{\mathrm{CICR}}$ can be substituted into any similarity-based classification algorithm.

3.3 The Adaptive Sobolev Measure

Thus far, we have shown that our hybrid LDA framework can efficiently and accurately combine CI and CR-based Euclidean distances. In this section, we describe how we extend our hybrid LDA framework to exploit the functional properties of spectral data using a distance measure based upon the Sobolev distance. Specifically, we define a convex weighted form of the parameterized Sobolev distance proposed by Villmann and Hammer [2009] that automatically weights distances between spectral derivates according to their relevance to the classification problem.

The form of the Sobolev distance we consider measures the distance between spectral signatures $\mathbf{x}_i$ and $\mathbf{x}_j$ according to

$$\mathrm{d}_{S^\kappa}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{l=0}^{\kappa} \gamma_l \alpha_l \ \mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j) \tag{3.12}$$

$$\mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)}\| \tag{3.13}$$

where $\mathrm{d}^{(l)}$ is the Euclidean distance between the l^{th} spectral derivatives of $\mathbf{x}_i$ and $\mathbf{x}_j$, γ_l

are scaling factors applied to each derivate (described below), and α_l are convex weight parameters (i.e., $\alpha_l \in [0, 1]$, $\sum_l \alpha_l = 1$) determining the contribution of each derivate in the hybrid measure. When $\kappa = 1$, d_{S^κ} reduces to the Euclidean distance (i.e., $\mathrm{d}^{(0)}$). We equalize the contribution of the derivates by setting $\gamma_l = 1/\sqrt{\mathrm{var}\left(\mathrm{d}^{(l)}\right)}$, where $\mathrm{var}\left(\mathrm{d}^{(l)}\right)$ is the sample variance with respect to derivate l. This maps the derivates of each sample to at most unit variance, and allows us better fine-tune the α_l weight parameters according to data-specific characteristics. As with the $\mathrm{d}_{\mathrm{CICR}}$ distance, we are faced with the problem of estimating the relevances of each of the $\mathrm{d}^{(l)}$ distances to maximize classification accuracy. To achieve this goal, we turn to our LDA-based hybrid metric learning method from Section 3.1.1. We calculate the vector of weights $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_\kappa]$ by extending the $\mathbf{M}_W$ and $\mathbf{M}_B$ matrices (Equations (3.5) and (3.6)) to measure the within and between class separation for each of the κ derivates, as follows:

$$\mathbf{M}_B = \begin{bmatrix} s_b(\mathrm{d}^{(1)}, \mathrm{d}^{(1)}) & \ldots & s_b(\mathrm{d}^{(\kappa)}, \mathrm{d}^{(1)}) \\ \vdots & \ddots & \vdots \\ s_b(\mathrm{d}^{(1)}, \mathrm{d}^{(\kappa)}) & \ldots & s_b(\mathrm{d}^{(\kappa)}, \mathrm{d}^{(\kappa)}) \end{bmatrix} \tag{3.14}$$

$$\mathbf{M}_W = \begin{bmatrix} s_w(\mathrm{d}^{(1)}, \mathrm{d}^{(1)}) & \ldots & s_w(\mathrm{d}^{(1)}, \mathrm{d}^{(\kappa)}) \\ \vdots & \ddots & \vdots \\ s_w(\mathrm{d}^{(1)}, \mathrm{d}^{(\kappa)}) & \ldots & s_w(\mathrm{d}^{(\kappa)}, \mathrm{d}^{(\kappa)}) \end{bmatrix}, \tag{3.15}$$

where s_b and s_w are computed using Equations (3.7) and (3.9), respectively.

3.3.1 Evaluation Methodology

We mimic the evaluation methodology described in Section 3.1.3, with two key differences. First, we consider a wider range of γ values than in our previous evaluation.

Specifically, we select γ from the $\{10^{-10}, \ldots, 10^{-2}, 0.2, \ldots, 0.8, 1-10^{-2}, \ldots, 1-10^{-10}\}$. Second, because we now must select a vector of α_i values of the form $\boldsymbol{\alpha}_{\mathrm{LS}} = \left[\frac{\alpha_1}{\sum_{l=1}^{\kappa} \alpha_i}, \ldots, \frac{\alpha_\kappa}{\sum_{l=1}^{\kappa} \alpha_i}\right]$, we limit the size of the line search space (LS) by allowing each α_l to take $A = 15$ (rather than $A = 100$, as before) uniformly spaced values in the $[0, 1]$ range. This is due to the fact the search space is of size $O(A^{\kappa-1})$, which becomes too large to search exhaustively for large values of A and κ. We evaluate the classification accuracy using the $\mathrm{d}^{(l)}$ measure for $l \in \{0, 1, 2, 3\}$, and with the Sobolev measure with $\kappa \in \{1, 2, 3\}$. We consider the unweighted (i.e., $\alpha_i = 1/\kappa$) Sobolev measure (UW), the LDA-based measure (LDA) and the $\boldsymbol{\alpha}_{\mathrm{LS}}$-based measure (LS). In each scenario, we consider the same set of 100 training and 100 testing samples per class as we used in Section 3.1.4.

3.3.2 Evaluation on Ocean City AVIRIS Imagery

We evaluated the performance of our adaptive Sobolev measure on the Minor, Major, and Combined absorption scenarios sampled from the Ocean City AVIRIS image, as described in Section 3.1.4. Table 3.3 gives the classification accuracy for each of the Ocean City scenarios using the $\mathrm{d}^{(l)}$ and d_{S^κ} measures. We typically observe 2-4% improvements in classification accuracy using the Sobolev measures over the baseline Euclidean distance ($\mathrm{d}^{(0)}$) and the remaining $\mathrm{d}^{(l)}$ measures. The LDA-based Sobolev measure generally produces slightly (1-2%) more accurate results than the unweighted Sobolev measure. However, the improvements in classification accuracy between the LDA and unweighted Sobolev measures are not nearly as significant as previously observed using our LDA-based technique with the $\mathrm{d}_{\mathrm{CICR}}$ measure. Moreover, we observe that both the unweighted and LDA-based Soboev measures decrease in accuracy as κ increases. This reduction in accuracy with respect to increasing κ is partially explained by the poor performance by the $\mathrm{d}^{(l)}$-based measures for the larger l values, indicating that the higher-order derivates are more ambiguous than the lower-order derivates. This is not surprising, as the derivatives become more smooth with increasing κ, eventually becoming completely flat as $\kappa \to \infty$. However, despite the ambiguity of the higher-order derivates, the LS-based Sobolev measure becomes more accurate with increasing κ values by exploiting the additional degrees of freedom provided with larger κ values in conjunction with the true class labels.

	$\mathbf{d}^{(l)}$				$\mathbf{d}_{S^\kappa},\ \kappa=1$			$\mathbf{d}_{S^\kappa},\ \kappa=2$			$\mathbf{d}_{S^\kappa},\ \kappa=3$		
	$\mathbf{d}^{(0)}$	$\mathbf{d}^{(1)}$	$\mathbf{d}^{(2)}$	$\mathbf{d}^{(3)}$	**UW**	**LDA**	**LS**	**UW**	**LDA**	**LS**	**UW**	**LDA**	**LS**
Minor	0.8863	0.7717	0.6362	0.6364	0.9047	*0.9108*	0.9210	0.8808	*0.9090*	0.9254	0.8627	0.9052	0.9268
	0.0134	0.0128	0.0067	0.0104	0.0134	0.0052	0.0043	0.0090	0.0077	0.0025	0.0084	0.0121	0.0066
Major	0.9256	0.9299	0.8857	0.8759	0.9616	0.9659	0.9707	0.9630	*0.9703*	0.9830	0.9543	*0.9688*	0.9804
	0.0108	0.0093	0.0104	0.0110	0.0087	0.0120	0.0063	0.0088	0.0178	0.0022	0.0085	0.0049	0.0031
Combined	0.8698	0.8276	0.7219	0.7082	*0.9123*	*0.9121*	0.9257	0.8976	0.9011	0.9300	0.8925	0.8967	0.9321
	0.0056	0.0137	0.0077	0.0090	0.0071	0.0070	0.0021	0.0079	0.0081	0.0045	0.0087	0.0075	0.0052

Table 3.3 : Mean (shaded rows) and standard deviation (unshaded rows) of classification accuracy on Ocean City spectra obtained with the $\mathrm{d}^{(l)}$ measures for derivates $l \in \{0, 1, 2, 3\}$, and with the d_{S^κ} measure for $\kappa \in \{1, 2, 3\}$. The best and second-best of the UW and LDA-based accuracies in each scenario are given in red and blue italics, respectively.

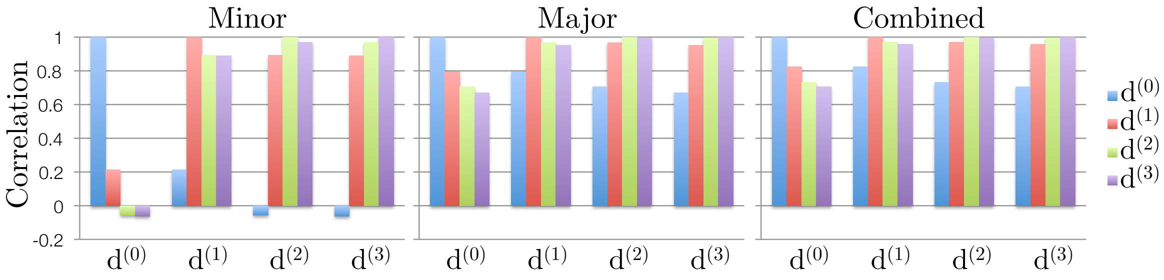


Figure 3.5 : Correlation coefficients for $\mathrm{d}^{(l)}$-based distances from each labeled sample to its class mean. Only $\mathrm{d}^{(0)}$ is substantially uncorrelated to the remaining $\mathrm{d}^{(l)}$.

The reduced classification accuracy using the unweighted and LDA-based Sobolev measures with large κ values can be explained by considering both the performance of the $\mathrm{d}^{(l)}$ measures and the correlation between the outputs produced by each of the $\mathrm{d}^{(l)}$

measures. Figure 3.5 gives the correlation coefficients for $\mathrm{d}^{(l)}$-based distances from each labeled sample to its class mean. As we can clearly see, distances produced by the $\mathrm{d}^{(0)}$ measure exhibit much lower correlation values to the $\mathrm{d}^{(l)}$, $l > 0$ measures in each scenario. The high correlations between the $\mathrm{d}^{(l)}$, $l > 0$ distances is unsurprising, as each derivate is simply a linear transformation of each spectral signature. The intuition here is that a classifer trained using one of two distinct similarity measures will produce increasingly similar errors as the correlation between the outputs each measure increases, an effect also observed by Lee et al. [2010]. Consequently, a hybrid measure consisting of several correlated distance measures will produce similar errors as the individual distances. This fact, combined with the decreasing performance of the $\mathrm{d}^{(l)}$ measures with increasing l, explains the increased accuracy using both the unweighted and LDA-based Sobolev measures with $\kappa = 1$ over $\mathrm{d}^{(0)}$, and the decrease in accuracy between the $(\kappa = 1)$-based and $(\kappa > 1)$-based unweighted and LDA-based Sobolev measures. This effect did not occur for our previous experiments with the CICR measure, where the CI and CR distaces were relatively uncorrelated. The LS-based measure does not suffer from these issues to the same degree in that it can explicitly select the α_i weights which maximize classification accuracy on both training and test data, at significantly higher computational cost.

An additional side-effect of the high correlation between the higher-order $\mathrm{d}^{(l)}$ is that the within-class scatter matrix (Equation (3.9)) does not provide useful discriminating information, and consequently we favor γ values near one. Table 3.4 gives the γ values with respect to κ for the LDA-based Sobolev measure. Note that the smallest γ value occurs in the Major absorption scenario for $\kappa = 1$, where we observe (in Table 3.3) similarly high classification accuracies for $\mathrm{d}^{(0)}$ and $\mathrm{d}^{(1)}$. As κ increases, γ also increases as the ambiguity between the within-class distances for higher order $\mathrm{d}^{(l)}$ increases.

		$\kappa = 1$		$\kappa = 2$			$\kappa = 3$			
		α_1	α_2	α_1	α_2	α_3	α_1	α_2	α_3	α_4
Minor	$\boldsymbol{\alpha}$	*0.8088*	*0.1912*	*0.7435*	*0.1797*	0.0768	*0.6769*	*0.1693*	0.0737	0.0801
		0.0038	0.0038	0.0080	0.0039	0.0052	0.0082	0.0021	0.0036	0.0039
	$\boldsymbol{\alpha}_{\mathbf{LS}}$	*0.6260*	*0.3740*	*0.6787*	*0.1693*	0.1520	*0.7451*	0.0000	0.0458	*0.2090*
		0.0646	0.0646	0.0725	0.0813	0.1017	0.0482	0.0000	0.0651	0.0549
Major	$\boldsymbol{\alpha}$	*0.6882*	*0.3118*	*0.6365*	0.1558	*0.2077*	*0.5360*	0.1712	0.0960	*0.1969*
		0.0437	0.0437	0.0702	0.0403	0.0474	0.1674	0.0716	0.0897	0.0481
	$\boldsymbol{\alpha}_{\mathbf{LS}}$	*0.6235*	*0.3765*	*0.6805*	0.1591	*0.1604*	*0.7021*	0.0799	*0.1113*	0.1067
		0.0687	0.0687	0.0698	0.1544	0.1188	0.1060	0.1058	0.0776	0.0783
Combined	$\boldsymbol{\alpha}$	*0.5013*	*0.4987*	*0.3552*	*0.3418*	0.3030	*0.2911*	*0.2606*	0.2210	0.2273
		0.0027	0.0027	0.0062	0.0031	0.0043	0.0045	0.0019	0.0027	0.0028
	$\boldsymbol{\alpha}_{\mathbf{LS}}$	*0.5637*	*0.4363*	*0.6394*	*0.2427*	0.1179	*0.6934*	0.0586	0.0624	*0.1856*
		0.0692	0.0692	0.0548	0.0483	0.0663	0.0339	0.0752	0.1036	0.1178

Table 3.5 : Mean (shaded rows) and standard deviation (unshaded rows) of α and α_{LS} values for Ocean City spectra obtained with the $\mathrm{d}_{S^{\kappa}}$ measure for $\kappa \in \{1, 2, 3\}$. The first and second largest α_i for $i \in \{1, \ldots, \kappa\}$ values for both $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}_{\mathrm{LS}}$ are given in red and blue italics, respectively.

Examining the differences between the LDA-based $\boldsymbol{\alpha}$ vs. α_{LS} values for the Ocean City scenarios is also instructive (shown in Table 3.5). We see that, for $\kappa \in \{1, 2\}$, the LDA-based $\boldsymbol{\alpha}$ closely approximates α_{LS} in both of the Major and Minor scenarios, but in the Combined scenario only the $\kappa = 1$ $\boldsymbol{\alpha}$ aligns well with α_{LS}. The $\boldsymbol{\alpha}$ estimates differ from α_{LS} most significantly for $\kappa = 3$ in all three scenarios, though arguably less so for the Minor and Major scenarios. These results are no surprise, as the $\mathrm{d}^{(l)}$ distances become more highly correlated to one another for increasing values of l, and are most correlated in the Combined scenario. Consequently, the $\boldsymbol{\alpha}$ predictions become less stable when the distances are highly-correlated.

	$\kappa = 1$	$\kappa = 2$	$\kappa = 3$
Minor	1.0000	1.0000	1.0000
	4.00e-7	4.89e-7	4.89e-7
Major	0.9520	0.9900	0.9920
	0.0760	0.0000	0.0040
Combined	0.9999	0.9999	0.9999
	0.0000	0.0000	0.0000

Table 3.4 : Mean (shaded rows) and standard deviation (unshaded rows) of γ values with respect to κ for the LDA-based Sobolev measure.

3.3.3 Asymptotic Behavior

While theory suggests that the asymptotic behavior of the Sobolev distance given in Equation (3.12) converges to the Sobolev distance consisting only of derivatives 0 and κ [Villmann, 2007], our observations show that in practice, data characteristics, errors induced in differentiation and combining redundant $\mathrm{d}^{(l)}$ distances may decrease the accuracy of the Sobolev measure for large κ. A solution is to avoid summing all derivates $l \in \{1, \ldots, \kappa\}$, but instead to select a single derivate $l > 1$ that contributes to the Sobolev distance, as follows:

$$\begin{aligned} \mathrm{d}'_{S^\kappa}(\mathbf{x}_i, \mathbf{x}_j) &= \gamma_0 \alpha'_0 \mathrm{d}^{(0)}(\mathbf{x}_i, \mathbf{x}_j) + \gamma_l \alpha'_l \mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j) & (3.16) \\ &= \gamma_0 (1 - \alpha'_l) \mathrm{d}^{(0)}(\mathbf{x}_i, \mathbf{x}_j) + \gamma_l \alpha'_l \mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j). & (3.17) \end{aligned}$$

Once again, $\{\alpha'_0, \alpha'_l\} \in [0, 1]$ are scalar weight parameters that emphasize the contribution of their respective derivate (described in detail below). We now need to select the l that produces the best classification accuracy for a given data set. The obvious approach implied from theory is to simply assign $l = \kappa$. We refer to this approach as l_κ. However, this approach may yield poor performance for large κ when the higher-order derivates become uninformative for classification. Thus, we also select the l corresponding to the maximum LDA-based weight according to $l = \underset{\ell,\ \ell>0}{\operatorname{argmax}}\ \alpha_\ell \in \boldsymbol{\alpha}$. We call this approach l_{LDA}. Given this l value, we form the convex combination $\{\alpha'_0, \alpha'_l\} = \{1 - \alpha'_l, \alpha'_l\}$ by re-weighting the $\{\alpha_0, \alpha_l\}$ from our previous estimate of $\boldsymbol{\alpha}$ as follows

$$\alpha'_l = \frac{\alpha_l}{\alpha_0 + \alpha_l}, \ \{\alpha_0, \alpha_l\} \in \boldsymbol{\alpha}. \qquad (3.18)$$

For comparison, we also evaluate the classification accuracy using the unweighted (i.e., $\alpha'_l = 0.5$) version of Equation (3.17). As before, we use the notation UW to refer to the unweighted versions of the l_κ and l_{LDA}-based d'_{S^κ} measures, respectively.

Table 3.6 gives the classification results for the d_{S^κ} vs the d'_{S^κ} measures for $\kappa \in \{2, 3\}$. In most cases, the weighted versions of l_κ and l_{LDA} outperform their unweighted counterparts, though the differences are not particularly significant in the Minor and Major scenarios. In the Combined scenario, we see a small ($\approx 1\%$) improvement using the weighted d'_{S^κ} measure for $\kappa = 2$, but observe only a slight increase for $\kappa = 3$ using the weighted d'_{S^κ} measure vs. its unweighted counterpart. It is interesting to note that the l_κ and l_{LDA} accuracies, along with their corresponding unweighted versions, produce very similar results. This observation suggests that in cases where $l_{LDA} \neq l_\kappa$ and the classification accuracies of $\mathrm{d}^{(\kappa)}$ and $\mathrm{d}^{(l_{LDA})}$ are comparable, using either method to select l produces similar results with the d'_{S^κ} measure.

More interesting perhaps is the comparison between d_{S^κ} and d'_{S^κ}. In all three scenarios, both the weighted and unweighted d'_{S^κ} measures yield accuracies comparable to the LDA-based d_{S^κ}. However, while the d_{S^κ} versus the d'_{S^κ} accuracies in the Minor and Major scenarios do not differ significantly, d'_{S^κ} consistently outperforms d_{S^κ} in the Combined scenario by $0.5 - 1.5\%$. The Combined scenario results indicate that in cases where it is unclear which derivates are the most informative, a good strategy is to select the most relevant of the $\mathrm{d}^{(l)}$ $(l > 0)$ measures based upon our initial estimate of $\boldsymbol{\alpha}$, and re-weight $\mathrm{d}^{(0)}$ and $\mathrm{d}^{(l)}$ in Equation (3.17) according to Equation (3.18).

Although the results given in Table 3.6 show only a small (0.5-1.5%) difference in accuracies between the d_{S^κ} and the d'_{S^κ} measures, it is important to note that the l_κ and l_{LDA} measures produce roughly the same accuracies independent of the value of κ, whereas the accuracies produced using the LDA-based d_{S^κ} measure decrease slightly with increasing κ. While this effect is not particularly dramatic, with less than a 0.5% decrease for the LDA-based d_{S^κ} measure, we expect larger κ values to

		$\mathbf{d}_{S^\kappa}$			$\mathbf{d}'_{S^\kappa}$			
		UW	**LDA**	**LS**	$\mathbf{UW}l_\kappa$	l_κ	$\mathbf{UW}l_{LDA}$	l_{LDA}
$\kappa = 2$	**Minor**	0.8808	*0.9090*	0.9254	0.8930	0.9058	*0.9111*	0.9087
		0.0090	0.0077	0.0025	0.0147	0.0103	0.0106	0.0089
	Major	0.9630	*0.9703*	0.9830	0.9645	0.9652	0.9645	*0.9659*
		0.0088	0.0178	0.0022	0.0129	0.0166	0.0129	0.0152
	Combined	0.8976	0.9011	0.9300	0.9058	*0.9147*	0.9058	*0.9163*
		0.0079	0.0081	0.0045	0.0070	0.0093	0.0070	0.0041
		UW	**LDA**	**LS**	$\mathbf{UW}l_\kappa$	l_κ	$\mathbf{UW}l_{LDA}$	l_{LDA}
$\kappa = 3$	**Minor**	0.8627	0.9052	0.9268	0.9006	*0.9087*	0.9000	*0.9067*
		0.0084	0.0121	0.0066	0.0050	0.0062	0.0039	0.0065
	Major	0.9543	*0.9688*	0.9804	0.9638	0.9645	0.9620	*0.9659*
		0.0085	0.0049	0.0031	0.0044	0.0061	0.0044	0.0050
	Combined	0.8925	0.8967	0.9321	0.9105	*0.9144*	0.9142	*0.9165*
		0.0087	0.0075	0.0052	0.0058	0.0063	0.0053	0.0052

Table 3.6 : Mean (shaded rows) and standard deviation (unshaded rows) of classification accuracy on Ocean City spectra obtained with the d_{S^κ} and the d'_{S^κ} measures for $\kappa \in \{2, 3\}$. The best and second-best accuracies (excluding the LS-based measure) for each value of κ in each scenario are given in red and blue italics, respectively.

decrease classification accuracy more substantially using the LDA-based d_{S^κ} measure. Specifically, based upon our results from Table 3.5 for $\kappa = 3$, we see that the most second most-relevant α_i weights according to $\boldsymbol{\alpha}$ and α_{LS} differ in all three scenarios. Additionally, the scales of the most-relevant weights (i.e., α_1) according to $\boldsymbol{\alpha}$ and α_{LS} also differ substantially, particularly in the Combined scenario. These issues suggest that the weights computed using the LDA-based d_{S^κ} measure may not accurately reflect the relevances of the individual distance measures to the classification task when an increasing number of the independent measures are highly correlated. The d'_{S^κ}-based measures are less susceptible to this issue, as they discard the least-informative $\mathrm{d}^{(l)}$ distances.

Chapter 4

Feature-Weighted Similarity Measures

Portions of this chapter are based upon the following publications:

- BD Bue, DR Thompson, MS Gilmore, and R Castaño. "Metric Learning for Hyperspectral Image Segmentation". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* [2011].
- BD Bue. *Low-rank Mahalanobis Metric Learning for Hyperspectral Image Classification: A Comparative Survey.* Tech. rep. Rice University (in preparation), 2013.

In Chapter 3 we considered the problem of learning a hybrid similarity measure consisting of a weighted combination of several distinct similarity measures. Such techniques are well-suited to scenarios where multiple similarity measures are available, each capturing uncorrelated notions of similarity. However, such notions of similarity are not always straightforward to define, and often require detailed a priori knowledge of the problem domain. An alternative, but complimentary, approach to hybrid metric learning is to learn the relevances of the individual features each sample represents with respect to the classification task. In this chapter[*] we consider the problem of *low-rank* Mahalanobis metric learning, where the objective is to learn a linear transformation matrix from $\mathbb{R}^n$ to $\mathbb{R}^m$, $m << n$, that induces a Mahalanobis distance measure. We provide a comprehensive evaluation of several Mahalanobis metric learning algorithms on the Ocean City, MD AVIRIS spectra described in Chapter 3, in addition to three well-studied, high-dimensional hyperspectral images captured by the Compact Reconnaissance Imaging Spectrometer for Mars (CRISM) instrument. We show empirically that, when properly regularized, multiclass LDA

[*] The work described in this chapter was performed in collaboration with David Thompson, Martha Gilmore and Rebecca Castaño, with assistance from Erzsébet Merényi and Marika Kästner.

is not only significantly more efficient, but also produces more stable and accurate results than several widely-used Mahalanobis metric learning algorithms. We then propose a methodology to improve hyperspectral image segmentation results using learned Mahalanobis metrics, and compare the performance of metrics learned using multiclass LDA vs. the state-of-the-art Information Theoretic Metric Learning (ITML) algorithm. We demonstrate our methodology by segmenting the three aforementioned CRISM images and show that segmentations produced using learned metrics are both visually and quantitatively superior to those produced using the Euclidean distance.

4.1 Mahalanobis Metric Learning

The goal in Mahalanobis metric learning is to compute a $n \times n$ symmetric, positive semi-definite matrix $\mathbf{M}$ that induces a Mahalanobis distance

$$\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j) \tag{4.1}$$

that best separates N labeled samples $(X, Y) = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, $\mathbf{x}_i \in \mathbb{R}^n$ representing K classes with labels $y_i \in \{1, \dots, K\}$. Because any $n \times n$ positive semidefinite matrix can be decomposed into the product of a $n \times m$ matrix $\mathbf{A}$ with its transpose $\mathbf{M} = \mathbf{A}\mathbf{A}^T$, the Mahalanobis metric learning problem is often framed in terms of learning an linear transformation $\mathbf{A}$ by optimizing an objective function $f(\mathbf{A})$ with respect to the labeled data.

Within the past decade, a number of approaches to learn Mahalanobis metrics have been proposed (e.g.,[Davis et al., 2007; Globerson and Roweis, 2006; Goldberger et al., 2005a; Tsang et al., 2005; Weinberger et al., 2006; Weizman and Goldberger, 2009; Xing et al., 2003]). Although the theoretical properties of several Mahalanobis metric learning algorithms have been compared [Yang and Jin, 2006], their relative performances for high-dimensional, multiclass classification tasks have not been systematically evaluated. Moreover, the current literature does not adequately address a number of challenges inherent to application domains where such classification problems arise, such as in hyperspectral image classification. Hyperspectral image data are high-dimensional, often contain many, potentially nonlinearly-separable, classes ($K > 10$), and in many cases, limited labeled data is available for training. Considering each of these issues is essential to demonstrate the effectiveness of Mahalanobis metric learning techniques for hyperspectral data, yet many existing studies only provide results on data sets of relatively low dimensionality or on classification problems with few classes (e.g., [Davis et al., 2007; Globerson and Roweis, 2006; Goldberger et al., 2005a; Sugiyama, 2007; Tsang et al., 2005; Weizman and Goldberger, 2009; Xing et al., 2003]). Additionally, several previous works [Davis et al., 2007; Jain et al., 2009; Weinberger et al., 2006] propose applying a feature selection algorithm or transforming data via Principal Components Analysis (PCA) as a preprocessing step before learning the Mahalanobis metric. However, such preprocessing discards important functional relationships between adjacent spectral bands, and often limits the classification sensitivity to discriminate between spectrally-similar classes in hyperspectral data [Merényi, 2000].

Current Mahalanobis metric learning techniques can be grouped into two categories: *LDA-based* algorithms, and *gradient-descent* algorithms. LDA-based algorithms learn the Mahalanobis matrix $\mathbf{M} = \mathbf{A}^T\mathbf{A}$ in closed-form by solving some formulation of the multiclass LDA objective function (Equation (4.2)). LDA-based algorithms have the advantage of speed, but may produce degenerate transformations when the number of features is greater than the number of available training samples. In contrast, the computation time of the gradient-descent algorithms varies with the complexity of the learning problem, but such algorithms typically rely upon weaker assumptions than the LDA-based algorithms, and do not necessarily produce degenerate transformations when the number of features outnumber the number of training samples. Thus, it is often argued that the benefits of gradient-descent algorithms in terms of classification robustness outweigh their computational costs. Indeed, several works demonstrate that such algorithms outperform LDA-based algorithms in various classification tasks [Globerson and Roweis, 2006; Weinberger et al., 2006; Weizman and Goldberger, 2009]. However, recent results demonstrate that *regularized* versions of LDA perform significantly better than the classical, unregularized form of LDA, often achieving accuracies comparable to state-of-the-art gradient-descent algorithms [Alipanahi et al., 2008] and more sophisticated classifiers such as SVMs [Bandos et al., 2009].

4.1.1 Low-rank Mahalanobis Metric Learning for Hyperspectral Image Classification

The goal of this work is to provide a comparative study of several state-of-the-art Mahalanobis metric learning algorithms evaluated on hyperspectral image classification tasks. We focus on the problem of *low-rank* Mahalanobis metric learning, where our objective is to learn a $n \times m$ transformation matrix $\mathbf{A}$, where $m << n$. Applying such a transformation reduces the dimensionality of the feature space, and potentially allows for convenient visualization of high-dimensional data. We consider both LDA-based and gradient-descent algorithms, and evaluate their performance on several hyperspectral image classification tasks of varying complexity. We characterize the performance of each algorithm in terms of its classification accuracy, computation time, and its sensitivity to tuning parameters and the size of the training set. We demonstrate that in most cases, multiclass LDA, combined with a simple and computationally efficient regularization procedure, performs as well or better than state-of-the-art techniques for low-rank Mahalanobis metric learning, with significantly lower computation time.

We now review the main details of the algorithms we consider in this work, along with information regarding their software implementations and our strategies for computing their respective free parameters. For as direct comparison as possible to previous works, we evaluate the K Nearest Neighbor (KNN) classification accuracy using the Mahalanobis matrix $\mathbf{M}$ calculated by each algorithm. We fix the number of neighbors for the KNN classifier to 3. All experiments are performed using 64-bit Matlab v7.12 on a Macbook Pro with 2.66GHz Intel Core i7 processor with 4GB memory. All implementations are pure Matlab implementations using native linear algebra routines – no pre-compiled (e.g., Matlab mex) functions are used. For more detailed information regarding each algorithm, we refer the reader to the corresponding references.

LDA-based Algorithms

Multiclass Linear Discriminant Analysis: Multiclass Linear Discriminant Analysis (LDA, Fisher [1936]) is a classical approach for classification and dimensionality reduction which has recently been applied in metric learning contexts (e.g., [Ghodsi et al., 2008; Hayden et al., 2011]). To learn the transformation matrix $\mathbf{A}$, we employ a regularized version of multiclass LDA. Multiclass LDA calculates the transformation matrix $\mathbf{A}$ which maximizes the ratio of between-class vs. within-class separation

$$f(\mathbf{A}) = \frac{\det(\mathbf{A}^T\mathbf{M}_B\mathbf{A})}{\det(\mathbf{A}^T\mathbf{M}_W\mathbf{A})}, \tag{4.2}$$

where $\det(\mathbf{M})$ is the determinant of matrix $\mathbf{M}$ and $\mathbf{M}_W$ and $\mathbf{M}_B$ are the within and between class scatter matrices, respectively, calculated according to

$$\mathbf{M}_W = \frac{1}{N}\sum_{j=1}^{K}\sum_{i:y_i=j}(\mathbf{x}_i - \boldsymbol{\mu}_j)(\mathbf{x}_i - \boldsymbol{\mu}_j)^T \tag{4.3}$$

$$\mathbf{M}_B = \frac{1}{N}\sum_{j=1}^{K} N_j(\boldsymbol{\mu}_j - \overline{\boldsymbol{\mu}})(\boldsymbol{\mu}_j - \overline{\boldsymbol{\mu}})^T, \tag{4.4}$$

Here, $\{\boldsymbol{\mu}_j\}_{j=1}^{K}$ are the mean vectors of each of the K classes, $\overline{\boldsymbol{\mu}}$ is the mean of the $\boldsymbol{\mu}_j$, and N_j is the number of samples in class j. By forming $\mathbf{A}$ from the top K-1 eigenvectors of

$$\mathbf{M}_W^{-1}\mathbf{M}_B \tag{4.5}$$

we define a projection into a K-1 dimensional subspace that captures the variability between features with respect to training data [Fisher, 1938].

When the number of training samples is less than the number of features, Equation (4.2) may become ill-posed. To prevent this, we regularize $\mathbf{M}_W$ using the shrinkage operator

$$\mathbf{M}'_W = (1-\gamma)\mathbf{M}_W + \gamma\mathbf{I}^n, \tag{4.6}$$

where $\mathbf{I}^n$ is the $n \times n$ identity matrix and $\gamma \in [0,1]$ is a regularization parameter that controls the influence of the within-class scatter matrix in the objective function. We use our LDA implementation from [Bue et al., 2011b] in this work [†].

Local Fisher Discriminant Analysis: Local Fisher Discriminant Analysis (LFDA, Sugiyama [2007]) combines LDA with an unsupervised dimensionality reduction technique known as Locality Preserving Projections (LPP, Niyogi [2003]). LFDA uses

[†] Available at: http://www.ece.rice.edu/~bdb1/#code

local within-class and between-class scatter matrices $\widetilde{\mathbf{M}}_W$ and $\widetilde{\mathbf{M}}_B$, defined as

$$\widetilde{\mathbf{M}}_W = \sum_{i,j=1}^{N} \widetilde{\mathbf{W}}_{ij}\ (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T \tag{4.7}$$

$$\widetilde{\mathbf{M}}_B = \sum_{i,j=1}^{N} \widetilde{\mathbf{B}}_{ij}\ (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T, \tag{4.8}$$

where the matrices $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{B}}$ weight the pairwise *local affinities* $\mathbf{G}_{i,j}$ between $\mathbf{x}_i$ and $\mathbf{x}_j$, according to

$$\widetilde{\mathbf{W}}_{ij} = \begin{cases} \mathbf{G}_{ij}/N_\ell & y_i = y_j = \ell \\ 0 & y_i \neq y_j \end{cases} \tag{4.9}$$

$$\widetilde{\mathbf{B}}_{ij} = \begin{cases} \mathbf{G}_{ij}(1/N - 1/N_\ell) & y_i = y_j = \ell \\ 1/N & y_i \neq y_j. \end{cases} \tag{4.10}$$

Here, the local affinities are computed according to $\mathbf{G}_{ij} = \exp\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}\right)$, where σ approximates the width of the Gaussian $\mathbf{G}_{ij}$ as the Euclidean distance between $\mathbf{x}_i$ and its k^{th} nearest neighbor. As with LDA, LFDA computes $\mathbf{A}$ by maximizing Equation (4.2), substituting $\widetilde{\mathbf{M}}_W$ and $\widetilde{\mathbf{M}}_B$ for $\mathbf{M}_W$ and $\mathbf{M}_B$, respectively. However, rather than forming $\mathbf{A}$ from the top $K-1$ eigenvectors of Equation (4.5), the author suggests using one of the following two methods (1) weighting the top m generalized eigenvectors of Equation (4.5), $\{\psi_i\}_{i=1}^m$ according to

$$\mathbf{A} = \left(\sqrt{\lambda_1}\psi_1 | \cdots | \sqrt{\lambda_m}\psi_m\right), \tag{4.11}$$

where λ_i is the eigenvalue associated with ψ_i; or (2) orthonormalizing the top m eigenvectors via the QR decomposition. We experimented with both methods and found that the orthonormalized $\mathbf{A}$ produced significantly better results than the weighted $\mathbf{A}$.

The primary benefit of using LFDA over traditional LDA is that sample pairs that are far apart within the same class have less influence on $\widetilde{\mathbf{M}}_W$ and $\widetilde{\mathbf{M}}_B$. This better accounts for classes with multimodal structure, in comparison to the original LDA formulation that assumes that each class is well-represented by its class mean. Sample pairs in different classes are not weighted by LFDA since the objective is to separate them regardless of their similarities in the original space. Additionally, while the between-class scatter matrix in the original LDA formulation (Equation (4.4)) has maximum rank $K - 1$, the rank of the LFDA between-class scatter matrix is generally much larger. Thus, LFDA permits dimensionality reduction to more than K-1 dimensions. We use the LFDA implementation provided by the author[‡], and assign $k = 3$, using the same number of nearest neighbors we use with the kNN classifier, as we describe in Section 4.1.2.

Discriminative Components Analysis: Discriminative Components Analysis (DCA, Hoi et al. [2006]) is a metric learning technique closely related to LDA. The primary difference between the two algorithms is in the form of class labels. While LDA assumes the class labels for training samples are known, DCA assumes a set of similarity/dissimilarity constraints between examples is provided, where each constraint indicates whether a pair of samples are similar (positive constraint) or dissimilar (negative constraint). DCA groups all of the samples with positive constraints together into *chunklets* – groups of samples belonging to the same, but potentially unknown,

[‡] Available at: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LFDA/index.html

class. DCA then defines a *discriminative set* for each chunklet by identifying the remaining chunklets that contain at least one negative constraint between them. Given the N samples in C chunklets, each sample is then labeled according to their discriminative sets $\{\hat{y}_i\}_{i=1}^{N}$, $\hat{y}_i \in [1, C]$. Then, DCA calculates the within-class and between-class scatter matrices using the chunklets as follows:

$$\widehat{\mathbf{M}}_W = \sum_{j=1}^{C} \sum_{i:\hat{y}_i=j} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_j)(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_j)^T \tag{4.12}$$

$$\widehat{\mathbf{M}}_B = \sum_{j=1}^{C} C_j(\hat{\boldsymbol{\mu}}_j - \hat{\bar{\boldsymbol{\mu}}})(\hat{\boldsymbol{\mu}}_j - \hat{\bar{\boldsymbol{\mu}}})^T \tag{4.13}$$

where $\left\{\hat{\boldsymbol{\mu}}_j\right\}_{j=1}^{C}$ are the mean vectors of each of the C chunklets, $\hat{\bar{\boldsymbol{\mu}}}$ is the mean of the $\hat{\boldsymbol{\mu}}_j$, and C_j is the number of samples in chunklet j. DCA forms $\mathbf{A}$ using the same method as LDA, i.e., from the top m eigenvectors of Equation (4.2), substituting $\widehat{\mathbf{M}}_W$ and $\widehat{\mathbf{M}}_B$ for $\mathbf{M}_W$ and $\mathbf{M}_B$, respectively. When the similarity and dissimilarity constraints defining the chunklets are constructed using all of the labeled samples in the training set, the DCA objective reduces to the classical, unregularized LDA described above (Equation (4.5)). However, we emphasize that DCA uses a subset of the training samples based upon the number of classes to define the chunklets, rather than the entire training set. We describe the method we use to select the similarity/dissimilarity constraints that form the set of DCA chunklets in Section 4.1.2.

We use the DCA implementation provided by Yang[§]. We note that this implementation uses the optimization technique proposed in [Yu and Yang, 2001] to solve the LDA objective function, and can be viewed as an alternative to the regularization-based approach we apply for multiclass LDA (Equation (4.6)).

[§] Available at: `http://www.cs.cmu.edu/~liuy/dca.zip`

Gradient Descent Algorithms

Neighbourhood Components Analysis: The Neighborhood Components Analysis algorithm (NCA, Goldberger et al. [2005b]) learns a Mahalanobis distance metric that minimizes an approximation of the leave-one-out (LOO) cross-validation error of nearest-neighbor classification. Specifically, let $Y_i = \{j | y_i = y_j\}$ be the indices of samples with the same label as $\mathbf{x}_i$, and $p^{\mathbf{A}}(j|i) = p_{ij}$ be the probability that $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighbors after applying transformation matrix $\mathbf{A}$, defined as follows:

$$p^{\mathbf{A}}(j|i) = p_{ij} = \frac{\exp\left(-\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_j\|\right)^2}{\sum\limits_{k \neq i} \exp\left(-\|\mathbf{A}\mathbf{x}_i - \mathbf{A}\mathbf{x}_k\|\right)^2}, \; p_{ii} = 0. \tag{4.14}$$

The probability of classifying $\mathbf{x}_i$ correctly can be expressed as $p_i = \sum_{j \in Y_i} p_{ij}$, and thus, the criterion we wish to maximize is the expected classification accuracy after applying transformation $\mathbf{A}$

$$f(\mathbf{A}) = \sum_{i=1}^{N} p_i. \tag{4.15}$$

Differentiating f with respect to $\mathbf{A}$ yields the gradient rule:

$$\frac{\delta f}{\delta \mathbf{A}} = 2\mathbf{A} \sum_i \left(p_i \sum_k p_{ik} x_{ik} x_{ik}^T - \sum_{j \in Y_i} p_{ij} x_{ij} x_{ij}^T \right) \tag{4.16}$$

where $x_{ij} = \mathbf{x}_i - \mathbf{x}_j$. Since Equation (4.15) is non-convex, it is not guaranteed to converge to the global optimum and may overfit to training data, particularly in high-dimensional feature spaces with few training samples [Yang and Jin, 2006]. To prevent overfitting, Singh-Miller et al. [2007] suggest regularizing $f(\mathbf{A})$ as follows:

$$f(\mathbf{A}) = \frac{1}{N} \sum_i p_i - \gamma \sum_{j,k} \mathbf{A}_{j,k}^2 \tag{4.17}$$

for $\gamma \geq 0$, selected via cross-validation.

Computationally, NCA minimizes Equation (4.15) using a conjugate gradient method that recomputes p_i for all of the training samples in each iteration, and thus NCA incurs a rather high computational cost. Although some recent work addresses this issue [Yang et al., 2012], we consider the original formulation of NCA in this study. We allow NCA to run for a maximum of 50 iterations, where each iteration consists of a single pass over all of the training samples. We consider the NCA implementation provided in the Matlab Toolbox for Dimensionality Reduction [van der Maaten, 2007].

Maximally Collapsing Metric Learning: Maximally Collapsing Metric Learning (MCML, Globerson and Roweis [2006]) is a convex extension of NCA that seeks to map all samples with the same class label to a single point, while pushing the samples from the other classes infinitely far apart. To do so, MCML selects $\mathbf{A}$ that minimizes the Kullback-Leibler divergence $\mathrm{KL}(p_0||p^{\mathbf{A}})$ (Equation (1.10)) between $p^{\mathbf{A}}(j|i)$ (Equation (4.14)) and the distribution p_0, which represents the distribution of optimally separated samples:

$$p_0(j|i) \propto \begin{cases} 1 & y_i = y_j \\ 0 & y_i \neq y_j \end{cases} \tag{4.18}$$

$\mathrm{KL}(p_0||p^{\mathbf{A}})$ is minimized using the objective function

$$f(\mathbf{A}) = -\sum_{i,j:y_j=y_i} \log p^{\mathbf{A}}(j|i) = \sum_{i,j:y_j=y_i} d_{ij}^{\mathbf{A}} + \sum_i \log Z_i, \tag{4.19}$$

where $d_{ij}^{\mathbf{A}} = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{A}^T \mathbf{A}(\mathbf{x}_i - \mathbf{x}_j)$, and Z_i is a convex function of affine functions in $\mathbf{A}$. The authors solve Equation (4.19) by taking a small step in the gradient direction during each iteration, and then by taking the eigendecomposition of $\mathbf{A}$

while removing (i.e., zeroing) the negative eigenvectors. This procedure projects the solution to the positive semidefinite cone of matrices $\mathbf{A}$. Due to the convexity of the objective function, this approach is guaranteed to converge to the globally optimal solution. However, since each iteration involves computing an eigendecomposition of an m-dimensional matrix, MCML is computationally expensive, as observed in, e.g., [Sugiyama, 2007]. Moreover, the MCML optimization problem becomes non-convex in low-rank settings. To resolve this issue, the authors propose solving for the full-rank matrix, and then using the spectral decomposition of that matrix to determine a low rank projection based on its top m eigenvalues. As with NCA, we allow MCML to run for a maximum of 50 iterations, using the implementation provided in the Matlab Toolbox for Dimensionality Reduction [van der Maaten, 2007].

Large Margin Nearest Neighbors: The Large Margin Nearest Neighbors (LMNN, Weinberger et al. [2006]) algorithm learns the Mahalanobis distance by finding a transformation that separates samples from different classes by a large margin, while simultaneously reducing the distances between each training sample to its k_{LMNN} nearest neighbors. To achieve this, LMNN computes $\mathbf{A}$ by solving a piecewise linear, convex function of the elements in the matrix $\mathbf{M}$ using the following semidefinite program (SDP):

$$\min_{\mathbf{A}} \left[(1-\gamma) \sum_{ij} \eta_{ij} d_{ij}^{\mathbf{A}} + \gamma \sum_{ijl} \eta_{ij} (1 - y_{il}) \xi_{ijl} \right] \tag{4.20}$$

$$s.t. \begin{cases} d_{il}^{\mathbf{A}} - d_{ij}^{\mathbf{A}} \geq 1 - \xi_{ijl} \\ \xi_{ijl} \geq 0 \\ \mathbf{A}^T \mathbf{A} \succeq 0, \end{cases} \tag{4.21}$$

where $d_{ij}^{\mathbf{A}} = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{A}^T \mathbf{A}(\mathbf{x}_i - \mathbf{x}_j)$. Here, $y_{ij} \in \{0, 1\}$ and $\eta_{ij} \in \{0, 1\}$ are binary indicator variables that specify whether $y_i = y_j$ and whether $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighbors, respectively. $\gamma \in [0, 1]$ is a regularization parameter that controls the influence of the penalty terms with respect to slack variables ξ_{ijl}, which are nonzero iff $\mathbf{x}_i$, $\mathbf{x}_j$, and $\mathbf{x}_l$ have different labels. The left term of Equation (4.20) penalizes large distances between each input and its neighbors, while the right term penalizes small distances between examples with different labels. We note that as with the NCA and MCML algorithms, obtaining low-rank transformations using LMNN requires solving a non-convex optimization problem, but the authors claim that the objective function does not appear to suffer from poor local minima [Weinberger and Saul, 2009].

LMNN often outperforms other baseline Mahalanobis metric learning algorithms such as NCA and MCML due to its maximum-margin formulation [Jin et al., 2009; Kulis et al., 2009; Yang et al., 2010]. However, it is also quite computationally expensive, sometimes scaling quadratically with the number of input dimensions, as shown empirically later in this work, and also in Shen et al. [2009]. In Weinberger and Saul [2009], the authors note that this may be a result of the poor performance of many SDP solvers, and thus developed a special-purpose solver¶ for LMNN. While this new version may be more efficient than a general SDP solver, it also this requires tuning several additional free parameters. Consequently, for as direct comparison as possible to the other Mahalanobis metric learning algorithms, we consider the implementation provided in the Matlab Toolbox for Dimensionality Reduction [van der Maaten, 2007]. We allow LMNN to run for a maximum of 10^5 iterations, where a single iteration involves computing the distances between the training samples with the current Mahalanobis matrix $\mathbf{M}$ to update the slack variables (Equation (4.20)) and

¶Available at: http://www.cse.wustl.edu/~kilian/code/page21/page21.html.

performing a gradient update on the k_{LMNN} nearest neighbors corresponding to those slack variables. We set $k_{\text{LMNN}} = 3$ to mirror the k parameter of our kNN classifier as described in Section 4.1.2.

Information Theoretic Metric Learning: The Information Theoretic Metric Learning (ITML, Davis et al. [2007]) algorithm exploits a bijection between the set of Mahalanobis distances and the set of multivariate Gaussians. This allows them to formulate the problem of learning $\mathbf{M} = \mathbf{A}^T\mathbf{A}$ as one of minimizing the Kullback-Leibler divergence (KL) divergence between two multivariate Gaussians: one that represents the Mahalanobis distance constrained by a set of similar/dissimilar samples, and one that represents a known Mahalanobis distance for regularization. Specifically, they express a Mahalanobis distance $\mathrm{d}_{\mathbf{M}}$ parametrized by $\mathbf{M} = \mathbf{A}^T\mathbf{A}$ as $p(\mathbf{x};\mathbf{M}) = \frac{1}{Z}\exp(-\frac{1}{2}d_{\mathbf{M}}(\mathbf{x},\boldsymbol{\mu}))$, where Z is a normalizing constant (without loss of generality, they assume the Gaussians share the same mean $\boldsymbol{\mu}$). The KL divergence between the Gaussians parametrized by Mahalanobis matrix $\mathbf{M}$ and regularization matrix $\mathbf{M}_0$ is expressed as the convex function

$$\mathrm{KL}(p(\mathbf{x};\mathbf{M}_0)||p(\mathbf{x};\mathbf{M})) \propto \mathrm{D}_{\mathrm{ld}}(\mathbf{M},\mathbf{M}_0) = \mathrm{tr}(\mathbf{M}\mathbf{M}_0^{-1}) - \log\det(\mathbf{M}\mathbf{M}_0^{-1}) - n. \quad (4.22)$$

Given a set of similarity $S_{ij} \in \{0,1\}$ and dissimilarity constraints $D_{ij} \in \{0,1\}$, $S_{ij} \neq D_{ij}$ between sample pairs $(\mathbf{x}_i, \mathbf{x}_j)$, ITML solves the following optimization problem:

$$\min_{\mathbf{M}} \mathrm{D}_{\mathrm{ld}}(\mathbf{M},\mathbf{M}_0) + \gamma \mathrm{D}_{\mathrm{ld}}(\mathrm{diag}(\boldsymbol{\xi}), \mathrm{diag}(\boldsymbol{\xi}_0)) \quad (4.23)$$

$$s.t. \begin{cases} \mathrm{d}_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \leq \boldsymbol{\xi}_{C_{ij}}, & S_{ij} = 1 \\ \mathrm{d}_{\mathbf{M}}(\mathbf{x}_i,\mathbf{x}_j) \geq \boldsymbol{\xi}_{C_{ij}}, & D_{ij} = 1, \end{cases}, \quad (4.24)$$

where $\gamma \in [0, 1]$ is a regularization parameter controlling the influence of slack variables $\{\boldsymbol{\xi}\}_{i=1}^{N_C}$, and C_{ij} gives the index of constraint (i, j). The entries of $\boldsymbol{\xi}$ are initially assigned the value u if $S_{ij} = 1$, and the value l if $D_{ij} = 1$. The values of u and l are estimated as the 5th and 95th percentiles, respectively, of the distribution of pairwise distances between training samples. The optimization procedure used to solve Equation (4.24) repeatedly computes projections of the current solution $\mathbf{M}_t$ onto a randomly-selected constraint $C_{ij} \in \{S \cup D\}$ according to

$$\mathbf{M}_{t+1} = \mathbf{M}_t + \beta \mathbf{M}_t(\mathbf{x}_i \mathbf{x}_j)(\mathbf{x}_i \mathbf{x}_j)^T \mathbf{M}_t \tag{4.25}$$

where β is a Lagrange multiplier corresponding to C_{ij}. Unlike the other gradient-descent algorithms we consider in this work, the ITML-based solution to the low-rank Mahalanobis metric learning problem is convex.

We use the ITML implementation provided by the authors[‖], and allow the algorithm to run for a maximum of 10^5 iterations, where a single iteration involves a single constraint projection (Equation (4.25)). We follow the methodology of Davis et al. [2007], and use the n-dimensional identity matrix $\mathbf{I}^n$ as the regularization matrix $\mathbf{M}_0$, and thus $\gamma = 1$ yields the squared Euclidean distance (i.e., the Mahalanobis distance parametrized by $\mathbf{I}^n$). We choose the set of similarity/dissimilarity constraints using the methodology described in Section 4.1.2, below.

4.1.2 Experimental Methodology

Rank of Projection Matrix A: We choose the dimensionality of the low-rank projection matrix $\mathbf{A}$ to be $\mathbb{R}^{n \times K-1}$, where K is the number of classes. We select

[‖] Available at: `http://www.cs.utexas.edu/~pjain/itml/`

$m = K - 1$ because it is the theoretically optimal value for LDA, but also because it yields stable performance using the remaining algorithms.

Performance Assessment: We measure the effects of training set size vs. classification accuracy by selecting a maximum of $N_j \in \{25, 50, 100, 200, 250\}$ samples from each class, and use two-fold cross-validation folds to balance the amount of computation time necessary while providing some detail on the generalization performance of each algorithm. In each fold, we evenly split the samples into training and test sets via stratified random sampling, and use the training samples to learn the Mahalanobis metric using each of the aforementioned algorithms. We report the mean and standard deviation of test accuracy over the two folds for each value of N_j.

Regularization: For those algorithms requiring regularization, we select their respective values of the regularization parameter γ via cross-validation. We evenly split the *training* data into train$_{CV}$ and test$_{CV}$ sets using stratified random sampling. We then learn a metric on the train$_{CV}$ set for each value of $\gamma \in$ {0, 0.001, 0.1, .25, .5, .75, 0.99, 0.999, 1}, and compute the accuracy on the test$_{CV}$ set using the metric produced using each γ value. We repeat this process twice and return the value of γ yielding the highest average accuracy over the two cross-validation splits.

DCA/ITML Similarity/Dissimilarity Constraints: To form the similarity and dissimilarity constraint sets used by DCA and ITML, select $N_C = C_f K^2$ $(C_f > 0)$ pairs of samples from the training set via random uniform sampling using the method implemented by Davis et al. [2007]. We add pairs of points in the same class to the set of similarity constraints, and pairs with different class labels to the set of dissimilarity constraints. Identical samples to one another are discarded from the constraint sets. Davis et al. [2007] found that ITML was generally robust to $C_f \geq 20$, but smaller values increased the variance in accuracy between folds. We experimented with different values of C_f and found that $C_f = 40$ produced generally stable results, and larger values did not significantly improve classification accuracy.

Gradient-descent Convergence Tolerance Parameter τ : Each of the gradient-descent algorithms test for convergence by determining if the value of their respective objective function is within some tolerance τ of the objective value at the previous iteration. A gradient-descent algorithm converges faster for large τ than for small τ, but small values of τ allow the algorithm to fine-tune its parameters, and are typically more accurate than large τ. Because the best value of τ depends on both the algorithm and the data, we vary $\tau \in \{25, 5, 1, 0.5, 0.1\}$ and report the accuracy and computation time corresponding to most accurate value of τ. We discuss each algorithm's sensitivity with respect to τ in greater detail in Section 4.2.1.

4.2 Case Studies: Ocean City AVIRIS and Mars CRISM Imagery

Ocean City, MD AVIRIS Image: We first evaluate the performance of each of the above metric learning algorithms on the minor, major and combined scenarios described in Section 3.1.4. With respect to Mahalanobis metric learning, the minor absorption scenario represents the most significant challenge due to the lack of discriminative spectral features to exploit. In contrast, the classes in the major absorption scenario are each distinguished by dramatic differences in absorption features, and are consequently better separated than the minor absorption classes, as suggested by our results in Chapter 3. In the combined scenario, each algorithm must find a compromise between the discriminative spectral features of the major absorption classes, and the spectrally similar and largely featureless classes of the minor absorption scenario.

CRISM Images 3e12, 3fb9, 863e: We also consider three well-studied CRISM [Murchie et al., 2007] images. The CRISM instrument captures spectral measurements over the $[1, 4]$ μm range, with over 400 measurement channels with spatial resolution of approximately 18 meters / pixel. The images we consider, 3e12, 3fb9, and 863e (omitting the frt0000 catalog prefix), originally studied by Thompson et al. [2010], are typical of planetary science data, with high noise and relatively low spatial resolution, and contain diverse spectra consistent with olivine, phyllosilicate, carbonate and sulfate minerals. The images were calibrated using the Brown CRISM Analysis Toolkit [Morgan et al., 2009], and noisy bands in the extreme short and long wavelengths were removed in previous work [Thompson et al., 2010], leaving a total of 231 bands in the [1.06, 2.58] μm range for analysis. An expert geologist (M. Gilmore) identified the primary material constituents in each of the images, along with the pixels containing the purest examples of each mineral, and defined class maps for the materials using the ENVI spectral angle mapper (SAM) function [RSI, 2008]. As a final step, the geologist examined the spectral angles for each class to filter out ambiguous or mixed materials. We exclude such pixels from the following performance evaluation. Our final preprocessing step is to normalize each spectrum by its Euclidean norm, to compensate for linear illumination effects [Pouch and Campagna, 1990]. See [Thompson et al., 2010] for further details regarding these images and their constituent materials. False color images of each image and the locations of labeled classes, along with the corresponding L^2 normalized mean spectra of each class are shown in Figure 4.1. The "dark" class in image 863e consists of mostly absorption-free spectra of dark materials, and has been used in previous work to enhance certain geologic features of interest [Mandrake et al., 2010; Thompson et al., 2010]. However, after L^2 normalization, the dark class appears quite similar to several of the other material classes in the scene. We stress that we do *not* exclude the dark pixels to provide a more comprehensive evaluation of our methodology.

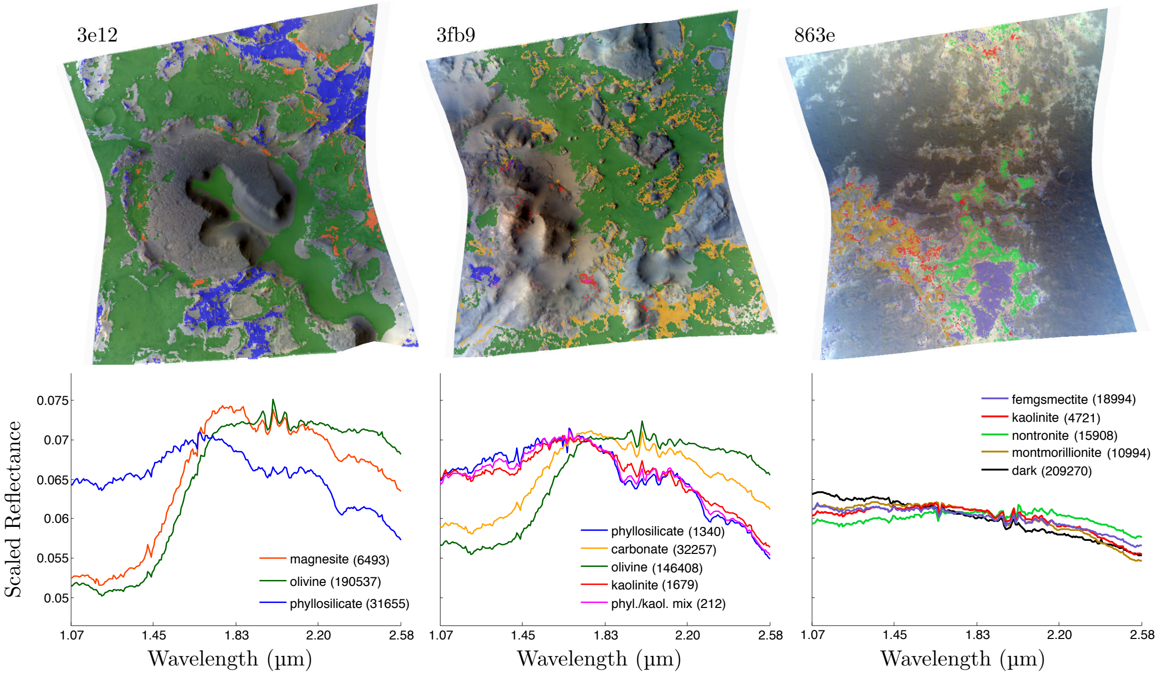


Figure 4.1 : Top: False color images with locations of labeled classes for CRISM images 3e12 (left), 3fb9 (middle) and 863e (right). Bottom: Corresponding class means and sample counts for each image. Due to varying capture conditions, spectra representing the same material species often have dramatically different spectral representations in each image.

While the CRISM images contain fewer classes than the Ocean City scenarios, with a total of 231 spectral bands they are of nearly twice their dimensionality and have a lower signal to noise ratio due to instrument artifacts and calibration errors that often occur in planetary imagery. Also, each image poses a distinct set of problems to the metric learning techniques. The first of the images, image 3e12, represents a fairly simple classification task involving a pair of similar olivine and magnesite classes vs. a spectrally dissimilar phyllosilicate class. In contrast, the classes in image

3fb9 are the most challenging to classify of the three images, with two pairs of similar classes – specifically: (phyllosilicate, kaolinite) and (carbonate, olivine) – along with a mixed class consisting of phyllosilicate and kaolinite minerals. The last of the three images, image 863e, also poses some interesting challenges, with four spectrally similar classes with comparable absorption features, distinguished primarily by differences in continuua.

4.2.1 Experimental Results

Accuracy vs. Training Set Size

Table 4.1 provides the average cross-validated accuracies over all values of N_j for each algorithm on each data set. The best and second-best performing algorithm on each data set are given in red and blue italics, respectively. We see that LDA yields the highest overall accuracy (97.72%) for these six data sets, with MCML (96.35%) and ITML (96.32%) following closely behind. LMNN yields comparable accuracy (96.23%) to MCML and ITML, but falls slightly behind due to poor performance on the Ocean City Combined data set. The remaining algorithms (LFDA, DCA and NCA) produce overall accuracies near the Euclidean baseline.

To give a more detailed view of the results summarized in Table 4.1, we display the cross-validation accuracy vs. the number of samples per class N_j in Figure 4.2 for each of the Ocean City scenarios (top three plots), and on the CRISM images (bottom three plots). Here, we see that LDA consistently matches or outperforms the Euclidean baseline on each data set. In contrast, MCML and ITML yield good performance for some training set sizes, but occasionally perform worse than the Euclidean distance. For instance, for the Minor data set, ITML yields the best accuracies of all of the algorithms for $N_j \in \{50, 100\}$, but produces the worst accuracy

			EUC	**LDA**	**LFDA**	**DCA**	**ITML**	**NCA**	**LMNN**	**MCML**	**Mean**
Ocean City	**Minor**	**(mean)**	94.6630	*95.7437*	94.9773	95.0654	95.1356	91.7682	95.6077	*95.7893*	94.8438
		(std)	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)	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)	0.2166	0.2945	0.4008	0.4513	0.5952	0.4230	0.3406	0.4147	0.3921
	Average	**(mean)**	96.9500	*97.3901*	97.0102	96.1596	97.2101	94.8496	97.0817	*97.3852*	96.7546
		(std)	0.7953	0.6241	0.7615	0.6989	0.7632	1.0045	0.7284	0.8330	0.7761
CRISM	**3e12**	**(mean)**	98.3581	98.6337	98.4915	*99.0391*	98.5892	98.2425	98.6089	*98.8337*	98.5996
		(std)	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)	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)	1.0842	0.8344	1.4661	1.6358	1.2115	0.4997	1.1785	1.1361	1.1308
	Average	**(mean)**	94.5550	*96.0682*	94.6294	94.4128	95.4483	*95.6445*	95.3771	95.3225	95.1822
		(std)	1.2476	1.3817	1.5132	1.1004	1.2788	1.2527	1.1325	1.2728	1.2725
	Overall	**(mean)**	95.7525	*96.7292*	95.8198	95.2862	96.3292	95.2470	96.2294	*96.3539*	95.9684
		(std)	1.02145	1.0029	1.1374	0.8997	1.0210	1.1286	0.9305	1.0529	1.0243

Table 4.1 : Mean and standard deviation of classification accuracies averaged over training set sizes $N_j \in \{25, 50, 100, 200, 250\}$ using each metric learning algorithm. Average accuracies for the Ocean City and CRISM data sets, and the overall accuracy across the data sets (bottom row), and across the algorithms (last column) are also provided. The top two most accurate algorithms for each data set are given in red and blue italics, respectively.

for $N_j = 25$. MCML, on the other hand, produces the best and second-best accuracies for $N_j = 25$ and $N_j = 50$, respectively, but produces poor accuracy for $N_j = 100$. We note that the high standard deviations in the Minor scenario using all of the algorithms for $N_j \in \{25, 50, 100\}$ indicates that the Minor absorption scenario is particularly challenging for Mahalanobis metric learning, as the classes do not contain significant distinguishing absorption features. However, the somewhat inconsistent performance of MCML and ITML on the Minor data set suggest that they are particularly sensitive to the choice of training set in scenarios involving fairly similar, perhaps nonlinearly-separable, classes and few training samples. The third best-performing algorithm overall, LMNN, while it does not always outperform ITML and MCML, typically produces more stable results than the other algorithms, as observed by its low overall standard deviation in classification accuracy (0.93%). This is most likely due to its

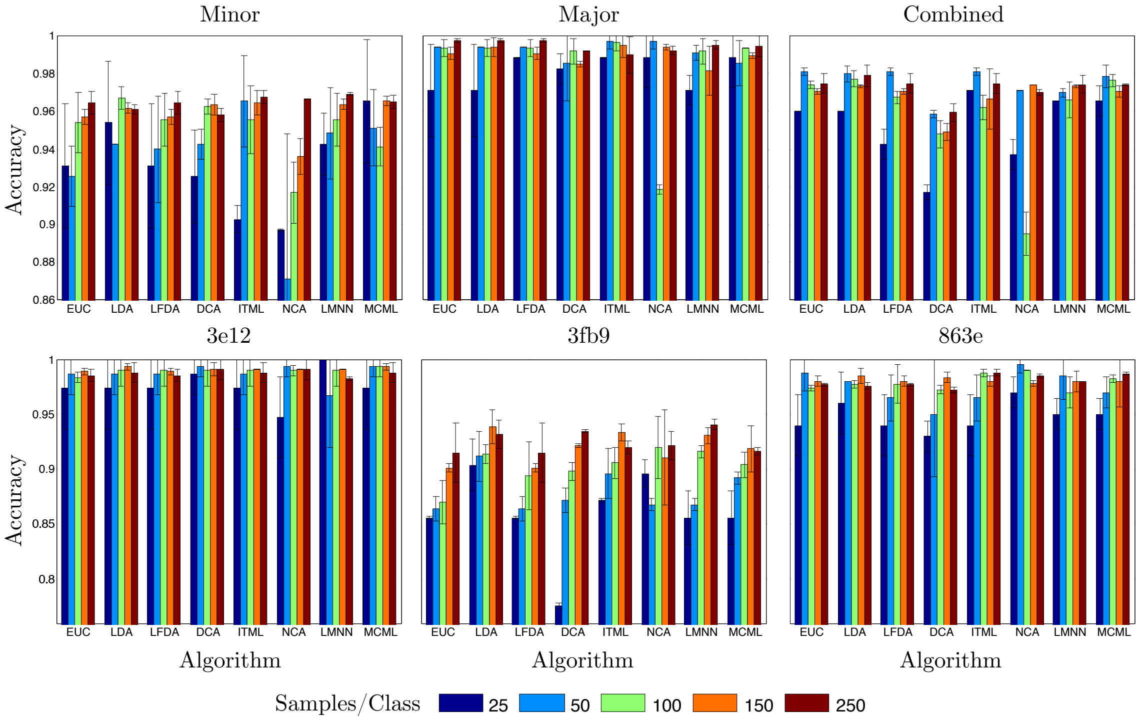


Figure 4.2 : Average kNN classification accuracy vs. training/testing samples/class on the Ocean City Minor (top left), Major (top center) and Combined (top right) data sets, and on the CRISM image 3e12 (bottom left), 3fb9 (bottom center) and 863e (bottom right) data sets. Error bars give the standard deviation of the cross validation folds.

large-margin formulation. In fact, only DCA produces consistently more stable (albeit less accurate) results than LMNN (std. dev. 0.90%). This is not surprising, as DCA's constraint-based formulation is computed using a smaller training set whose size varies with the number of classes, rather than the number of samples N_j. Conseqently, DCA typically produces similar Mahalanobis distances for different values of N_j, with the exception of the smaller N_j values where classification accuracy is less stable. It is also interesting to recall that DCA is, in essence, an unregularized version of LDA, and thus, DCA's poor overall performance reflects the problem of applying LDA without an appropriate regularization procedure. This issue is also evidenced in the poor overall accuracies of LFDA, which generally produces accuracies comparable to the Euclidean distance. Despite the lack of regularization, both DCA and LFDA outperform the worst-performing gradient-descent algorithm, NCA, which generates unusually inconsistent results. More specifically, in the Minor and Combined scenarios, NCA performs up to 6% worse the Euclidean baseline. In others, such as on image 863e, NCA produces fairly stable (std. dev. 0.5%) and accurate (98.37%) results. This inconsistent performance may be attributed to the non-convex nature of the low-rank projection technique employed by NCA (and similarly, but to a lesser degree, MCML). However, such a wide range of accuracies is rather alarming, and suggests that NCA may not be well-suited for learning low-rank Mahalanobis metrics for hyperspectral image classification tasks.

CPU Time vs. Training Set Size

We now evaluate the computation time used by each algorithm with respect to the number of samples for each cross-validation fold. For those algorithms requiring regularization, we include the cost of searching over the values of γ specified in Section 4.1.2 in each fold. Figure 4.3 gives the CPU time vs. training set size for the cross-validation folds that produced the results shown in Figure 4.2. Not surprisingly, we see a clear dichotomy between the CPU times of the LDA-based algorithms vs. the gradient-descent algorithms. The somewhat atypical ITML runtimes are a result of the fact that ITML employs a constant number of similarity constraints (proportional to the number of classes), and thus ITML produces comparable CPU times regardless of training set size. For the remaining gradient-descent algorithms (i.e., NCA, LMNN, and MCML), the differences between the two paradigms are particularly severe for the larger sample sizes (i.e, $N_j \in \{150, 250\}$), whose CPU times are several orders-of-magnitude greater than the LDA-based algorithms. In particular, LMNN and MCML take over an hour to converge on the larger training sets. For the smaller sample sizes (i.e., $N_j \in \{25, 50\}$), the gradient-descent algorithms (with the exception of ITML) converge in a matter of minutes, but are still far more expensive than the closed-form LDA-based algorithms.

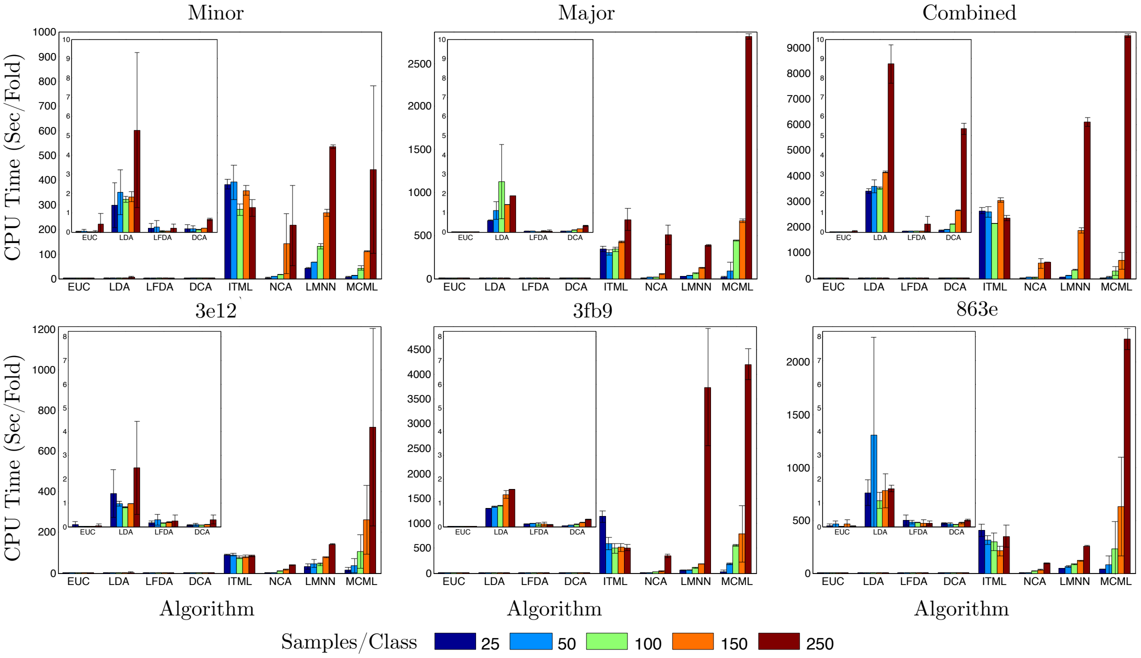


Figure 4.3 : Average CPU time (seconds / fold) vs. training/test samples/class on the Ocean City Minor (top left), Major (top center) and Combined (top right) data sets, and on the CRISM image 3e12 (bottom left), 3fb9 (bottom center) and 863e (bottom right) data sets. Y-axis scales differ to reflect the relative relationships between algorithms for each data set. The inset box in each figure gives a zoomed view of the EUC, LDA, LFDA and DCA CPU times. Error bars give the standard deviation of the cross validation folds.

Gradient-Descent Algorithms: Sensitivity to Tolerance Parameter

Our observations from Section 4.2.1 show that the gradient-descent algorithms exhibit different rates of convergence. We now investigate the sensitivity of each of the gradient-descent algorithms to tolerance parameter τ for both small (50 samples / class) and large (250 samples / class) training sets. Here, we limit our discussions to the Ocean City data sets, as we observed similar trends on the CRISM data sets.

Figure 4.4 gives the average classification accuracy with respect to the tolerance parameter τ for the Ocean City scenarios using 50 (top plots) and 250 (bottom plots) training samples/class. When few training samples are available, we observe that classification accuracy generally improves with smaller tolerance values, with $\tau \in \{1, 0.5\}$ and occasionally 0.1 producing the most accurate results. This is expected, as the additional fine-tuning imposed with a small value of τ can potentially compensate for ambiguities resulting from limited training data. Perhaps more interesting, however, are the accuracies observed with large training sets. While we note that classification accuracy does not vary widely ($\approx \pm 0.05\%$) with τ in all three scenarios, we observe a slight upward trend in the Minor scenario for each algorithm except NCA (which exhibited particularly unstable performance in this scenario), with $\tau \in \{1, 0.1\}$ producing the most accurate results, while we observe slightly downward trends in the Major and Combined Scenarios, where $\tau \in \{5, 1\}$ yield the most accurate results. As the classes in the Minor scenario are distinguished by subtle differences in features, the results suggest that the additional fine-tuning imposed by smaller τ can potentially improve accuracy on such data. In cases where the classes are already well-separated, such as in the Major scenario (and, to some extent, the Combined scenario), our observations suggest that smaller τ may produce minor overtraining effects. This may be a result of the algorithm attempting to find a transformation matrix that separates samples from non-linearly separable classes, and, as a consequence, reduces the separability of samples near the decision boundaries.

Figure 4.5 gives the average computation time per fold with respect to τ for the small and large training set cases shown in Figure 4.4. We reiterate that ITML employs a training set of size proportional to the number of classes and not the number of samples per-class, and thus produces similar CPU times regardless of the number

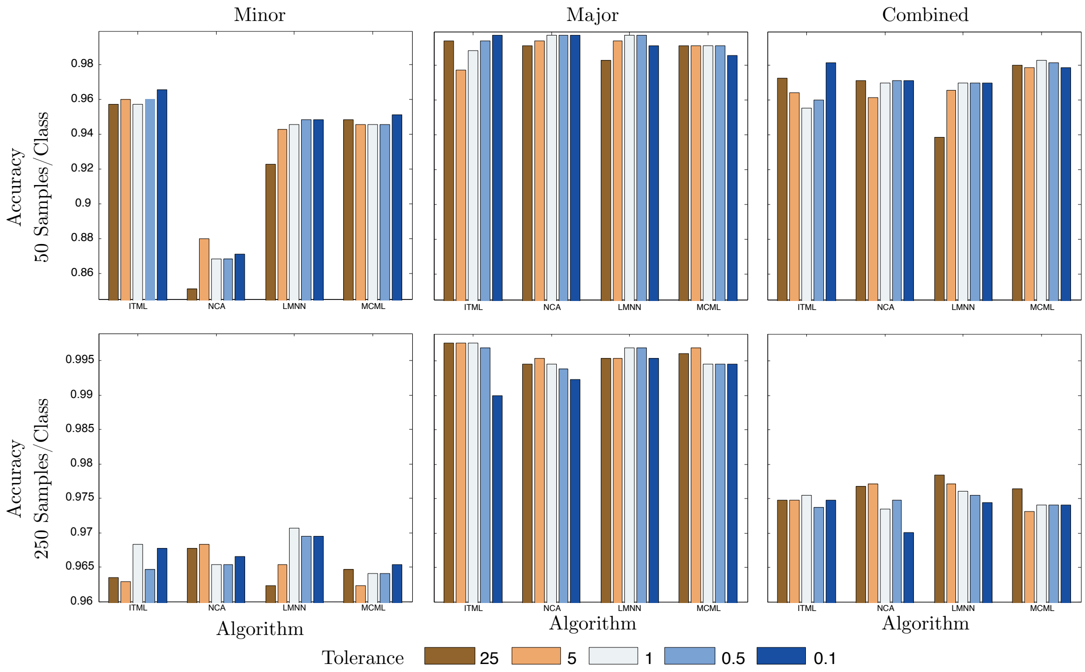


Figure 4.4 : Average classification accuracy vs. tolerance parameter τ for 50 samples / class (top plots) and 250 samples / class (bottom plots) on the Ocean City Minor (left), Major (middle) and Combined (right) data sets.

of samples / class. In general, computation time increases in inverse proportion to τ for both small and large training sets. For small training sets, the computation times for the most accurate τ are relatively short, converging in $30 - 60$ seconds per-fold in the Minor and Major scenarios, and about $60 - 120$ seconds per-fold in the combined scenario – owing to the fact that the Combined scenario contains twice as many samples as the other two scenarios. With large training sets, we observe the smallest variance in CPU times for the selected values of τ in the Minor scenario, where each algorithm must rely upon detailed fine-tuning to achieve the most accurate results. In contrast, we observe significantly *longer* computation times in the Major and Combined scenarios for the smaller τ values, which produce the *least* accurate results. This supports our hypothesis that the fine-tuning induced with small τ in these two scenarios is attempting to separate nonlinearly-separable samples, as each algorithm takes very small steps during each iteration in a ill-defined gradient direction. However, it is crucial to note that the computation time required for the τ values achieving the best accuracies are generally several orders-of-magnitude greater than the computation times produced by the LDA-based algorithms.

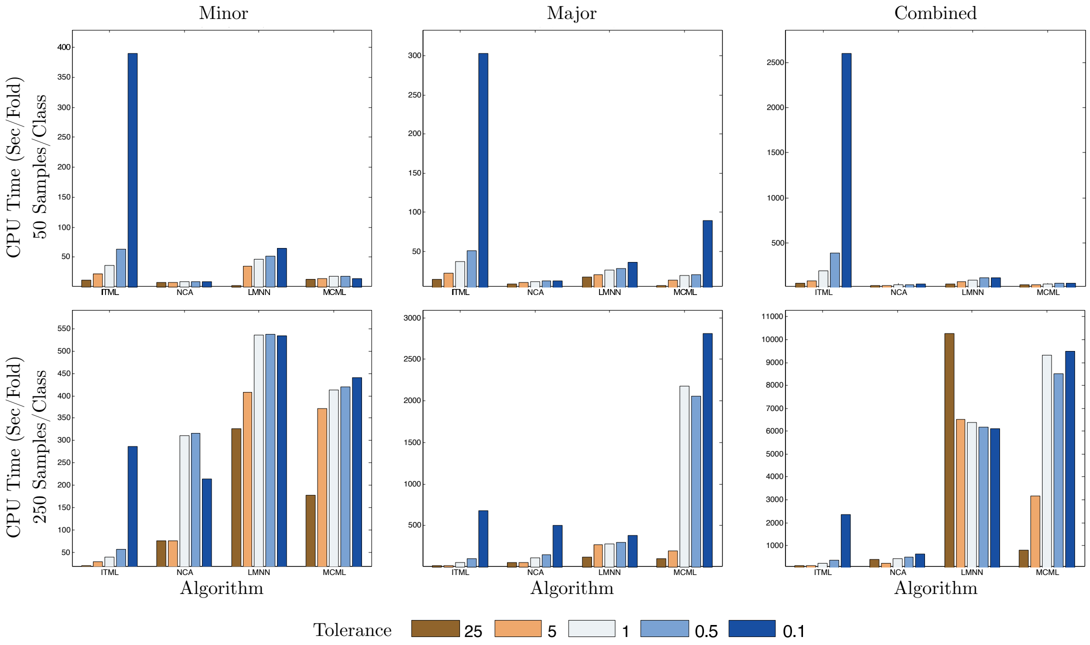


Figure 4.5 : Average CPU time (seconds / fold) vs. tolerance parameter τ for 50 samples / class (top plots) and 250 samples / class (bottom plots) on the Ocean City Minor (left), Major (middle) and Combined (right) data sets. Figure scales are different to reflect relative relationships between algorithms for each of the data sets.

Comparisons Between Learned Mahalanobis Matrices

Examining the characteristics of the Mahalanobis matrices computed by each algorithm is also instructive. Here, we focus on the diagonal entries of each matrix, which can be interpreted as a vector of weights applied to each spectral band. Figure 4.6 gives the diagonal vector of the Mahalanobis matrix for each algorithm for CRISM image 3fb9, in comparison to the mean signatures for each of the five mineral classes. We

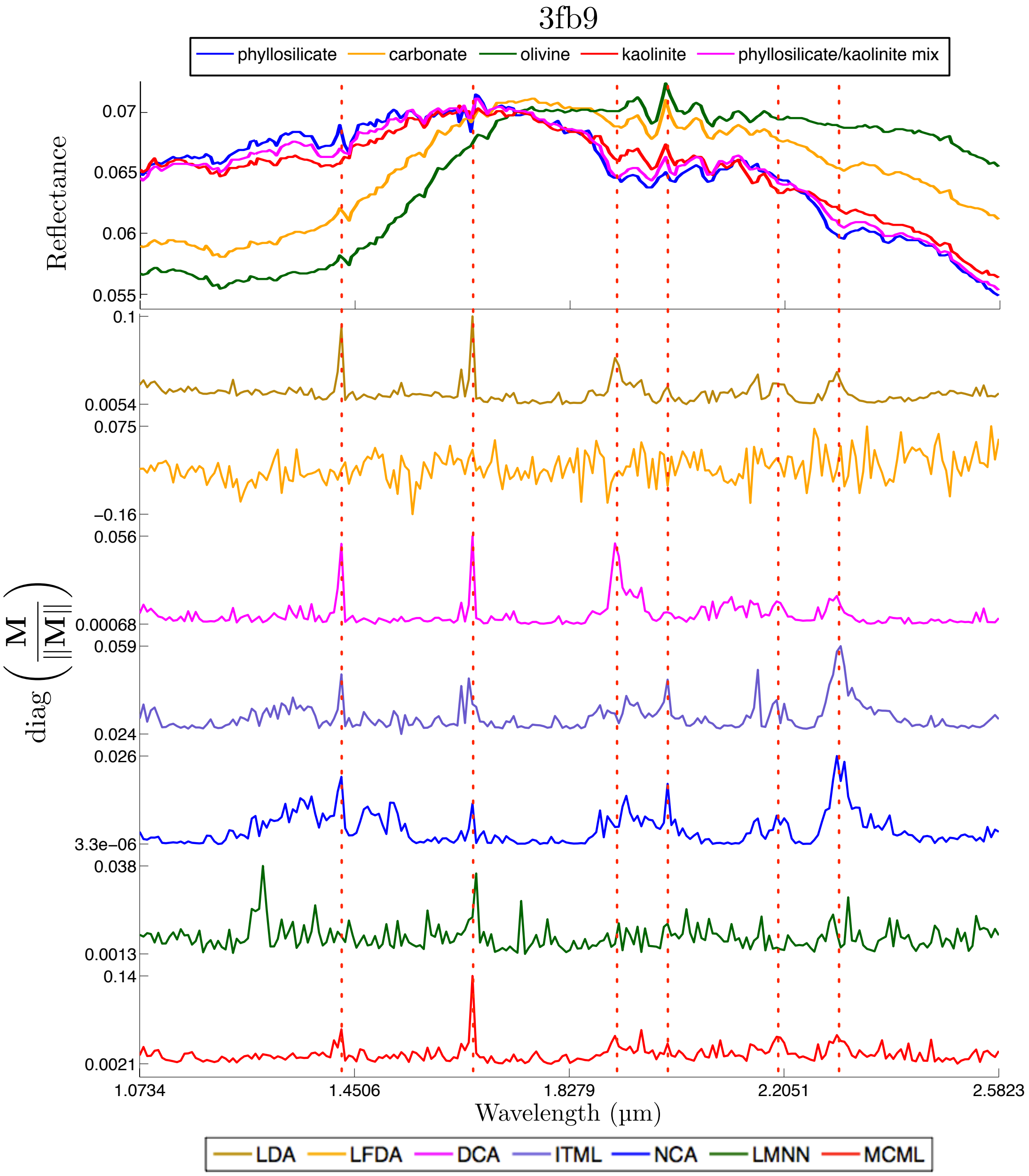


Figure 4.6 : Class means for CRISM image 3fb9 (top) vs. diagonals of Mahalanobis matrices computed by each metric learning algorithm for $N_j = 100$ samples/class. Several prominent peaks which occur for multiple algorithms are indicated by red dotted vertical lines.

scale each matrix by its L^2 norm $\|\mathbf{M}\|$ to map the entries of the diagonal vectors to a common range, and flag prominent peaks in the diagonal vectors which occur for multiple algorithms with red dotted vertical lines. Perhaps unsurprisingly, the most prominent peaks occur for spectral bands where absorption features differ the most among the classes, which, to some extent, explains the comparable accuracy of the CICR measure to the feature-weighted LDA Mahalanobis metric described in Section 3.2. We also observe that several of the algorithms produce relatively similar diagonal vectors. The similarity between LDA and DCA is expected, as DCA is effectively a constraint-based version of LDA. More interesting is the similarity between the ITML and NCA diagonals, with peaks occurring at nearly identical positions, with the exception of the differing double peaks at ≈ 1.63 and ≈ 2.27 μm. The MCML diagonal also bears some resemblance to LDA/DCA, particularly in terms of the peaks near 1.63μm and for wavelengths $\geq 2.0\mu$m, but the peaks at the remaining wavelengths differ in their relative amplitudes. Of the remaining algorithms, both LFDA and LMMN produce diagonals that differ substantially from the other techniques. These differences are expected, as both LFDA and LMNN place specific emphasis on *local* relationships between samples, and thus, the weights characterize the relationships between samples in close proximity to one another more so than samples that distant from each other in the original feature space. Consequently, their Mahalanobis matrices differ from the other algorithms, which emphasize the *global* relationships among classes. Despite their differences, however, both LFDA and LMNN produce reasonably accurate results, with LMNN producing the 2nd best accuracy of all the algorithms, and while LFDA does not perform as well, it still outperforms the Euclidean distance by $\approx 3\%$.

Figure 4.7 shows the pairwise differences between the L^2 normalized Mahalanobis matrices computed by each algorithm for $N_j \in \{25, 100\}$ samples per class. A value of zero indicates the matrices are identical, while a value of 2 indicates the (L^2 normalized) matrices are maximally dissimilar from one another. To emphasize the sometimes

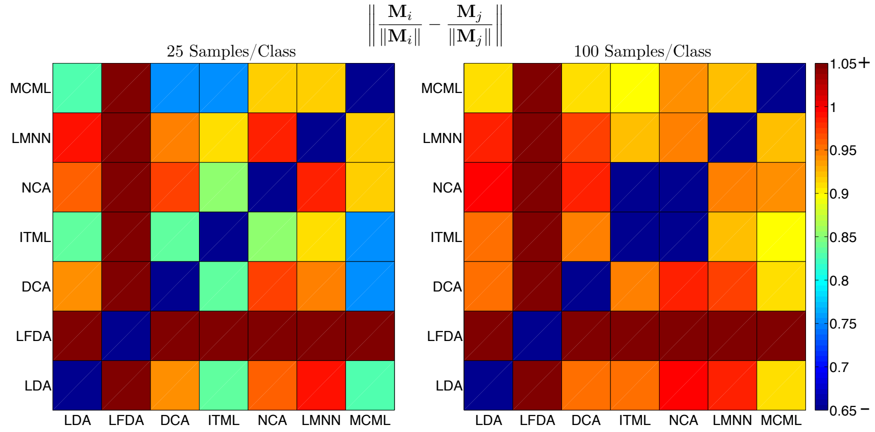


Figure 4.7 : Differences between L^2-normalized Mahalanobis matrices computed using each algorithm for $N_j = 25$ samples/class (left) $N_j = 100$ samples/class(right). Values less than 0.65 and greater than 1.05 are shown in dark blue and dark red, respectively.

subtle differences between the matrices, we clip the range of the values we display to $[0.65, 1.05]$, where values less than 0.65 and greater than 1.05 are shown in dark blue and dark red, respectively. We can see that the similarities in the diagonals shown in Figure 4.6 are largely reflected in the differences between the matrices. As before, due to its local nature, the matrix computed using LFDA is substantially different from those produced by the other algorithms. Interestingly, the NCA and ITML matrices become increasingly similar to one another with increasing quantities of training samples, with distances of about 0.8554 for $N_j = 25$ and 0.4467 for $N_j = 100$. This trend continued for $N_j \in \{150, 250\}$, where the difference between the ITML and NCA matrices are typically 15% more similar than the mean similarity between the remaining algorithms. We also see that the matrices produced by both LDA and DCA are most similar to ITML and MCML, and to a lesser degree, NCA. Interestingly, we observe that LDA and DCA are relatively dissimilar to one another, in comparison to the matrices produced by the other algorithms, despite the visual similarity of their diagonal vectors shown in Figure 4.6. This is primarily a result of a difference in the dynamic range of the entries of their respective matrices – which causes a slight shift in scale after L^2 normalization. The relatively low values between LMNN and ITML/MCML suggest a small degree of structural similarity between their matrices, but additional examination is necessary to verify if this is truly the case, as their diagonals appear (visually) dissimilar from one another.

4.2.2 Summary and Discussion

Table 4.2 summarizes the performance each metric learning algorithm on the Ocean City and CRISM scenarios. We assign a score of one when the algorithm yields good performance, and a score of three when the algorithm performs poorly, on average, over the set of classification scenarios we consider. We consider the performance of each algorithm with respect to three variables: the number of training samples N_j, and, for the gradient-descent algorithms, the tolerance parameter for small sample sizes (i.e., $N_j = 50$) vs. large sample sizes ($N_j = 250$). For each variable, we consider the following criteria: *Acc.*: measures the performance in terms of the overall classification accuracy produced using each algorithm, on average, for each value of the given variable (i.e., the number of samples or tolerance) and for all of the scenarios we consider; *Degen.*: measures how often the algorithm produces degenerate results (i.e., below the Euclidean baseline) – a score of 1 indicates the algorithm falls subtantially below (i.e., by roughly $> 10\%$ of the range between the minimum and maximum accuracies for a particular scenario) the baseline accuracy less than twice (total), for each value of the variable in all of the scenarios; *Stable*: gives the variability in classification accuracy for each algorithm with respect to the baseline Euclidean distance and the other algorithms; *CPU*: scores the variability in computation time / fold for each algorithm in comparison to the other algorithms;. In the case of the tolerance parameter, scores are based upon relative comparisons between the gradient-descent algorithms only. We omit the Acc. and Degen. criteria from the table for the tolerance parameter, as all of the gradient-descent algorithms, with the exception of NCA, produced comparable results (as shown in Figure 4.4).

	# Samples				**Tol** ($N_j = 50$)		**Tol** ($N_j = 250$)		
	Acc.	**Degen.**	**Stable**	**CPU**	**Stable**	**CPU**	**Stable**	**CPU**	**Mean**
LDA	1	1	1	1	n/a	n/a	n/a	n/a	*1.000*
LFDA	2	2	2	1	n/a	n/a	n/a	n/a	1.750
DCA	2	3	3	1	n/a	n/a	n/a	n/a	2.250
ITML	1	1	2	2	1	2	2	2	*1.625* (*1.500*)
NCA	3	3	3	2	2	1	2	2	2.250 (2.750)
LMNN	1	1	2	3	2	1	2	3	1.875 (1.750)
MCML	1	1	2	3	1	2	1	3	1.750 (1.750)

Table 4.2 : Summary of performance of each metric learning algorithm on Ocean City and CRISM scenarios. 1=good performance, 3=poor performance. Values indicated as *n/a* are not included in the mean calculation, and values in parenthesis give the mean scores for the gradient-descent algorithms with respect to the # Samples criteria alone. The best and second-best performing algorithms are given in red and blue italics, respectively.

Our results demonstrate that low-rank metrics learned using several metric learning techniques employing computationally expensive gradient-descent methods produce results comparable to those generated by several LDA-based techniques. In particular, when appropriately regularized, LDA produces the most accurate low-rank Mahalanobis matrices of all of the algorithms we considered, on a diverse set of classification problems, each with training sets of varying size. While these results may seem somewhat sobering, we stress that they are limited to the case of learning Mahalanobis matrices of rank $K-1$. This distinction is crucial, as the rank $K-1$ solution produced by LDA is optimal if all class distributions are Gaussian with a single shared covariance. Although this condition rarely holds in practice, when the class distributions are well-approximated by Gaussians of this form, a LDA-based formulation has an advantage over competing algorithms. However, this also limits the transformations that can be produced by the classical LDA algorithm to at most $K-1$ dimensions. In contrast, the gradient-descent techniques, and additionally, the LFDA algorithm, can produce transformation matrices of rank up to n. Indeed, the NCA, MCML, and LMNN algorithms can be solved in convex form only when $\mathbf{M}$ is full-rank (i.e., when the rank of the transformation matrix is equal to the number of features). Learning a transformation matrix of rank larger than $K-1$ using such techniques allows for additional degrees of freedom to separate samples from different classes, and can potentially produce more accurate and stable results than in the $K-1$ rank case, at the cost of extra computation time and higher dimensionality.

We also note that the techniques described in this work learn *linear* transformations to separate samples from different classes. When classes are nonlinearly separable in their original n-dimensional feature space, it is often advantageous to employ nonlinear techniques to separate such data. Several kernel-based approaches have been proposed to extend the techniques we describe in this work to learn transformations that can separate nonlinearly separable classes ([Alipanahi et al., 2008; Davis et al., 2007; Globerson and Roweis, 2006; Hoi et al., 2006; Sugiyama, 2007; Weinberger et al., 2006]). The high-dimensional kernel spaces employed by these techniques are often more informative than the original feature space, and thus, low-rank transformations can be learned efficiently with more accurate results, as demonstrated in, e.g., [Globerson and Roweis, 2006].

Our results also indicate that the simple shrinkage-based regularization procedure we apply to the classical LDA algorithm produces better results in low-rank scenarios than the alternative formulations of LDA employed by the DCA and LFDA algorithms. To some extent, this supports the claims reported in previous works that applying LDA without regularization performs poorly. However, we could potentially apply the same type of regularization as in Equation (4.6) to either of the LDA or LFDA techniques. Regularizing the LFDA objective is of particular interest, as it would allow us to potentially find a compromise between the local affinities on the manifold, and the global smoothness of the regularized objective function.

4.3 Mahalanobis Metric Learning for Hyperspectral Image Segmentation

In this section, we focus on the problem of hyperspectral image segmentation, where the goal is to partition an image into disjoint, spectrally homogeneous groups of spatially adjacent pixels called *segments*. A good segmentation not only reveals spatial trends that show the physical structure of a scene to an analyst, but also dramatically reduces the number of effective spectra to be analyzed. However, many segmentation algorithms employ unweighted similarity measures to quantify the relationships between spectral signatures. Such measures are often confused by noise, instrument artifacts, or spectral variations that are irrelevant to the classes of interest. Here, we propose a methodology to improve hyperspectral image segmentation results using task-specific similarity/distance measures.

4.3.1 Felzenszwalb Segmentation Algorithm

We consider the Felzenszwalb segmentation algorithm for its simplicity and computational efficiency [Felzenszwalb and Huttenlocher, 2004; Thompson et al., 2010]. Figure 4.8 shows the main concepts of the Felzenszwalb algorithm, which we describe in detail below.

The Felzenszwalb algorithm employs an agglomerative clustering approach that groups spatially-adjacent pixels $\mathbf{x}_i$ and $\mathbf{x}_j$ based on a pairwise distance $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$. The algorithm represents the image as an 8-connected grid of nodes where each node corresponds to a single pixel. The edges between the nodes in the graph are weighted according to $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$. All pixels are initially treated as separate segments and iteratively joined into larger groups. The maximum internal edge weight of a

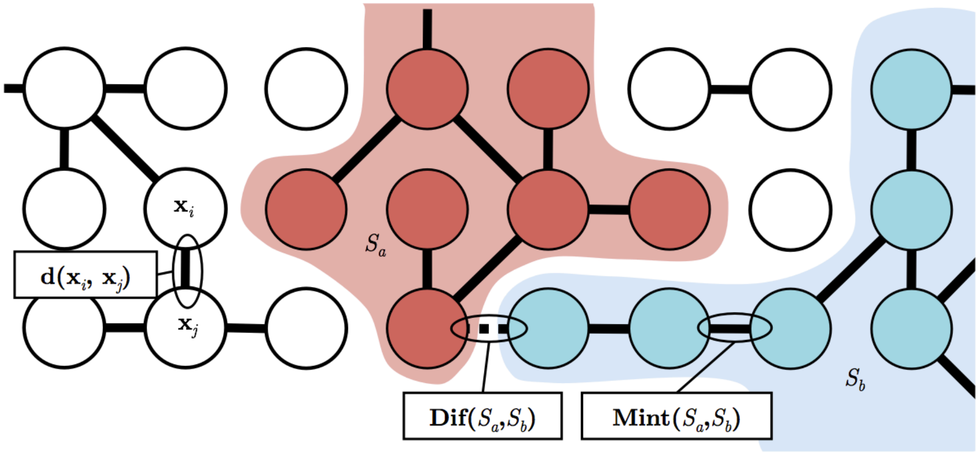


Figure 4.8 : Conceptual details of the Felzenszwalb segmentation algorithm. Pixels are represented as nodes in an 8-connected graph. Weights in the graph are determined by the distance $\mathrm{d}(\mathbf{x}_i, \mathbf{x}_j)$ between spatially-adjacent pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Segments S_a and S_b are indicated by red and blue regions, respectively. Segment boundaries are determined by the difference between their neighbors $\mathrm{Dif}(S_a, S_b)$ vs. their internal weights: $\mathrm{MInt}(S_a, S_b)$, respectively.

segment S, $\mathrm{Int}(S)$, is defined as the largest edge weight in its minimum spanning tree, $\mathrm{MST}(S)$.

$$\mathrm{Int}(S) = \max_{\mathbf{x}_i,\mathbf{x}_j} \mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) \;\forall\; \mathbf{x}_i \in S,\; \mathbf{x}_j \in S, \quad (\mathbf{x}_i, \mathbf{x}_j) \in \mathrm{MST}(S) \tag{4.26}$$

The smallest edge weight that joins two neighboring segments S_a and S_b (i.e. the most similar pixel pair on their border) defines the cross-segment distance:

$$\mathrm{Dif}(S_a, S_b) = \min_{\mathbf{x}_i,\mathbf{x}_j} \mathrm{d}(\mathbf{x}_i, \mathbf{x}_j) \;\forall\; \mathbf{x}_i \;\in S_a, \mathbf{x}_j \in S_b, \quad (\mathbf{x}_i, \mathbf{x}_j) \in E \tag{4.27}$$

Two adjacent segments are merged when the cross-segment distance $\mathrm{Dif}(S_a, S_b)$ is larger than Mint – the minimum of both internal weights weighted by an internal bias b and inversely proportional to the area of the segment $|S|$.

$$\mathrm{Mint}(S_a, S_b) = \min\left(\mathrm{Int}(S_a) + \frac{b}{|S_a|}, \mathrm{Int}(S_b) + \frac{b}{|S_b|}\right) \tag{4.28}$$

Larger b values cause a preference for larger segments, but is not a minimum segment size – smaller segments are allowed when there is a sufficiently large difference between spatially neighboring segments. However, in some cases, a minimum segment size is desirable, so we merge small segments below a user-defined threshold $t \geq 1$ with their spectrally-closest neighbors.

We attempt a *superpixel* segmentation in which the image is conservatively over-segmented; that is, we accept that single surface features may be split into multiple segments, but try to ensure that each individual segment - or superpixel - has homogeneous mineralogy [Thompson et al., 2010]. Figure 4.9 gives example superpixels produced by coarse vs. fine segmentations. By analyzing superpixels rather than individual pixels, we reduce the number of effective spectra to analyze in a given image, and potentially mitigate issues caused by instrument noise and intraclass variability.

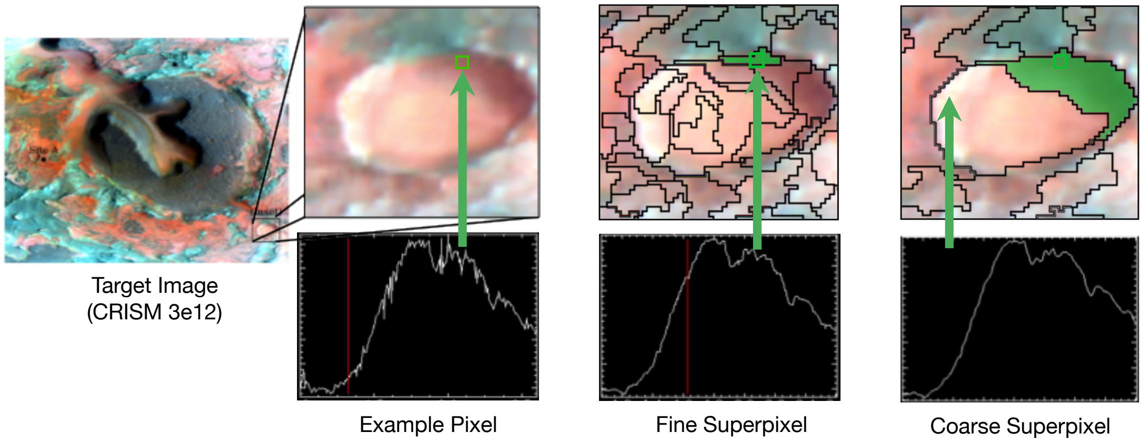


Figure 4.9 : Segmentation of an image patch from CRISM image 3e12 (described in detail in Section 4.2). Fine segmentations capture distinctions between materials better than coarse segmentations, but are more susceptible to noisy features. Coarse segmentations are less susceptible to noise and produce fewer segments to analyze, but may blur important class distinctions. Figure adapted from [Thompson et al., 2010].

Figure 4.10 outlines the main steps of the methodology we use to learn the Mahalanobis metric and apply it using the Felzenszwalb segmentation algorithm. We consider both multiclass LDA and the state-of-the-art Information-Theoretic Metric Learning (ITML) algorithm [Davis et al., 2007]. Both algorithms and their respective parameters are described in detail in Section 4.1.

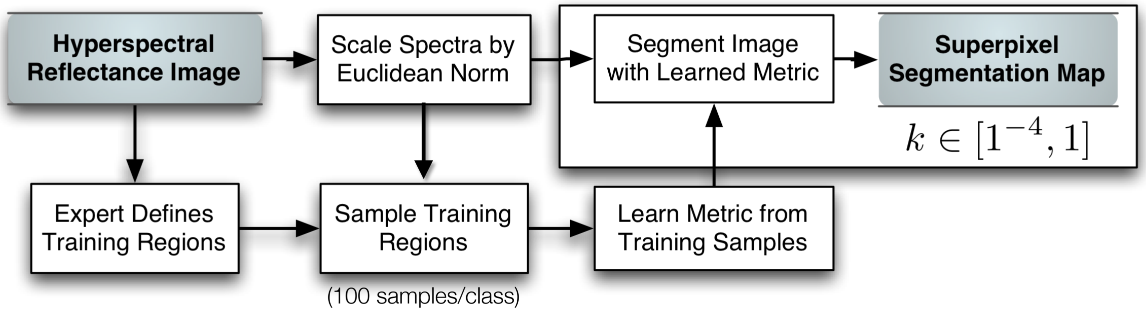

Hyperspectral Reflectance Image
Scale Spectra by Euclidean Norm
Segment Image with Learned Metric
Superpixel Segmentation Map
$k \in [1^{-4}, 1]$
Expert Defines Training Regions
Sample Training Regions
(100 samples/class)
Learn Metric from Training Samples

Figure 4.10 : Methodology for hyperspectral image segmentation using learned similarity measures.

4.3.2 Measuring Segmentation Quality

We measure the quality of the segmentations produced by each distance measure according to the homogeneity of the segments with respect to the class labels. However, because each superpixel segmentation is an *oversegmentation* of a given image, each expert-labeled class will be split into multiple segments. We expect the resulting segments to be better separated with respect to the training classes – i.e., pixels in each segment will belong to a single training class, rather than multiple classes – when we use a learned metric to segment the image, in comparison to metrics which do not account for class relationships. We define two measures to quantify the degree to which the resulting segments partition distinct classes. The first measure is the conditional entropy of the class map given the segmentation map, H(class|segment). H(class|segment) quantifies the remaining uncertainty for a random variable – in our case, the distribution of material classes – given the value of another random variable – the partitions produced by segmentation algorithm. In the case of a perfect segmentation of the classes, H(class|segment) will be zero, as the segmentation perfectly reconstructs the class map. Thus, we prefer smaller values of H(class|segment). Our second measure of segmentation quality, the *impurity ratio,* is the ratio of *impure* vs. *pure* segments with respect to the class labels. A pure segment consists of pixels belonging to a single class, whereas an impure segment consists of pixels belonging to multiple classes. Because segment size can bias this score, we scale the impurity ratio for each segment by its pixel area. As with H(class|segment), smaller impurity ratios are better.

4.4 Case Study: CRISM Image Segmentation

We now evaluate the quality of segmentations produced by the Euclidean distance vs. the LDA and ITML metric learning algorithms on the CRISM images described in Section 4.2. We proceed by splitting each image into two spatially contiguous halves, sampling 100 spectra from each class from the first half of the image (subsequently referred to as the "train" image), and use these points to train each metric learning algorithm. We then segment the train image and the remaining half of the image (the "test" image), using the (baseline) Euclidean distance and the LDA and ITML-based Mahalanobis distances. To objectively compare results between several metrics, we must compare segmentations that produce a similar number of superpixels. Because both the distance metric and the internal bias b (Equation (4.28)) alter the size – and subsequently the quantity – of the resulting superpixels, we describe results for segmentations produced by each distance measure using a range of $b \in \{10^{-}4$, $\ldots$, $10^{1}\}$. We choose this range because the number of superpixels produced by each distance measure followed a similar trend for all of the images we studied. We focus on segmentations that produce 200-1250 superpixels, as segmentations with few superpixels tend to inadequately capture morphological characteristics of the imagery we study, while segmentations with large quantities of superpixels are more sensitive to noise and insignificant differences in spectra. We ignore superpixels consisting of less than $t = 50$ pixels, as they tend to be unstable and noisy with respect to the training classes. Ignoring these small superpixels allows for a more consistent evaluation of the resulting segmentation maps between different distance measures.

4.4.1 Experimental Results

Figure 4.11 gives a set of segmentation maps for image 863e where the Euclidean and LDA/ITML-learned metrics produced a comparable number of segments. The number of segments for the train/test images are provided for each segmentation. Visually, the LDA-based segmentation tends to produce segments that better match the underlying morphology of the image data. This is particularly evident in the Fe/Mg-smectite class (light blue region) shown in the zoomed images. The Euclidean-based segmentation, and to a lesser degree, the ITML-based segmentation, both suffer from column striping artifacts as the noisy spectral bands are not adequately attenuated using these metrics. The LDA-based segments also tend to follow class boundaries slightly better than the other two algorithms, as evidenced by the tightness of the segment boundaries to the colored regions. These differences

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

Table 4.3 : Average pure pixels / segment for Euclidean, LDA and ITML-based segmentations of image 863e (Figure 4.11). Best and worst average per-class accuracy given in green and red font, respectively.

are also reflected in the percentages of pure pixels / segment given in Table 4.3. Both learned metrics outperform the baseline, with LDA improving over the Euclidean metric for material classes FeMg Smectite, Montmorillonite and Nontronite. ITML gives comparable performance to LDA for most materials, but the gains are not as significant for the very similar Montmorillionite and Nontronite classes.

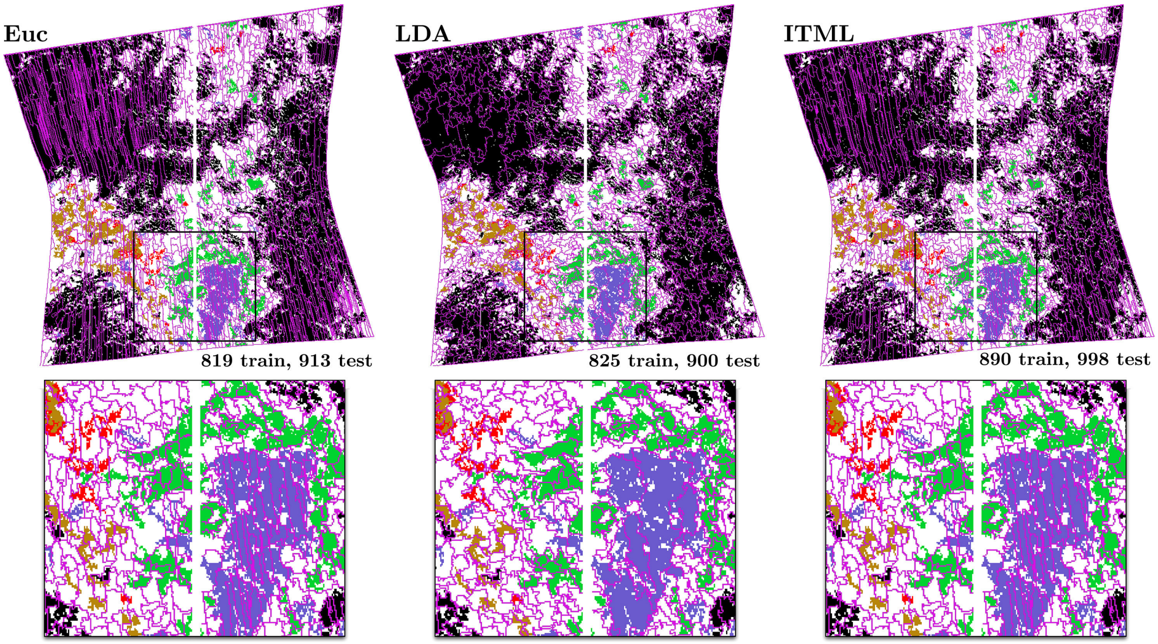


Figure 4.11 : Image 863e class locations and segments produced using the Euclidean (left), LDA (middle) and ITML (right) measures. Top images: Training/testing regions indicated by the white vertical line. Bottom images: zoom of rectangular region shown in each of the top images. Segments are indicated by purple lines, class locations colored according to the legend shown in Figure 4.1. The total number of segments produced in each of the training/test images using each measure is given in text above the zoomed regions. Visually, we see that the LDA-based segmentation does not split spatially-adjacent pixels with identical class labels as frequently as the EUC/ITML-based segmentations. This is particularly evident in the Fe/Mg-smectite class (light blue region) shown in the zoomed images, where column-striping artifacts split the region up into numerous segments in the Euc and ITML-based segmentations.

Figures 4.12 and 4.13 give the H(class|segment) and impurity ratios vs. the number of segments produced using each metric. LDA outperforms both the Euclidean metric and ITML, sometimes dramatically (e.g. on images 863e and 3fb9). The Euclidean metric gives the worst performance of the three distance measures, which is not surprising since it is more susceptible to noise that a learned metric can often suppress. ITML yields similar performance to the Euclidean distance for training images 3e12 and 3fb9, which is likely because the quantity of training samples is small for these two images. On image 863e, with training samples belonging to 5 material classes, ITML approaches the performance of LDA. This is also reflected in the summary statistics per-image for each segmentation given in Tables 4.4 and 4.5. Note that the performance improvements on testing data over training data on the 863e image are due to the fact that the test image contains a smaller number of Kaolinite (670) and Montmorillionite (93) pixels than in the training image, which are easily confused with other training classes (e.g., Kaolinite vs. FeMg Smectite).

H(class|segment) (Train/Test)

	EUC	**LDA**	**ITML**
3e12	0.017/0.068	0.015/0.059	0.019/0.066
3fb9	0.088/0.380	0.050/0.242	0.097/0.354
863e	0.047/0.004	0.018/0.001	0.031/0.002

Table 4.4 : Average H(class|segment) for each image and similarity measure. Green and red fonts indicate the best and worst performing metrics, respectively.

Impurity (Train/Test)

	EUC	**LDA**	**ITML**
3e12	0.018/0.062	0.012/0.057	0.020/0.060
3fb9	0.066/0.296	0.037/0.195	0.075/0.294
863e	0.068/0.032	0.040/0.012	0.061/0.027

Table 4.5 : Average impurity ratios for each image and similarity measure. Green and red fonts indicate the best and worst performing metrics, respectively.

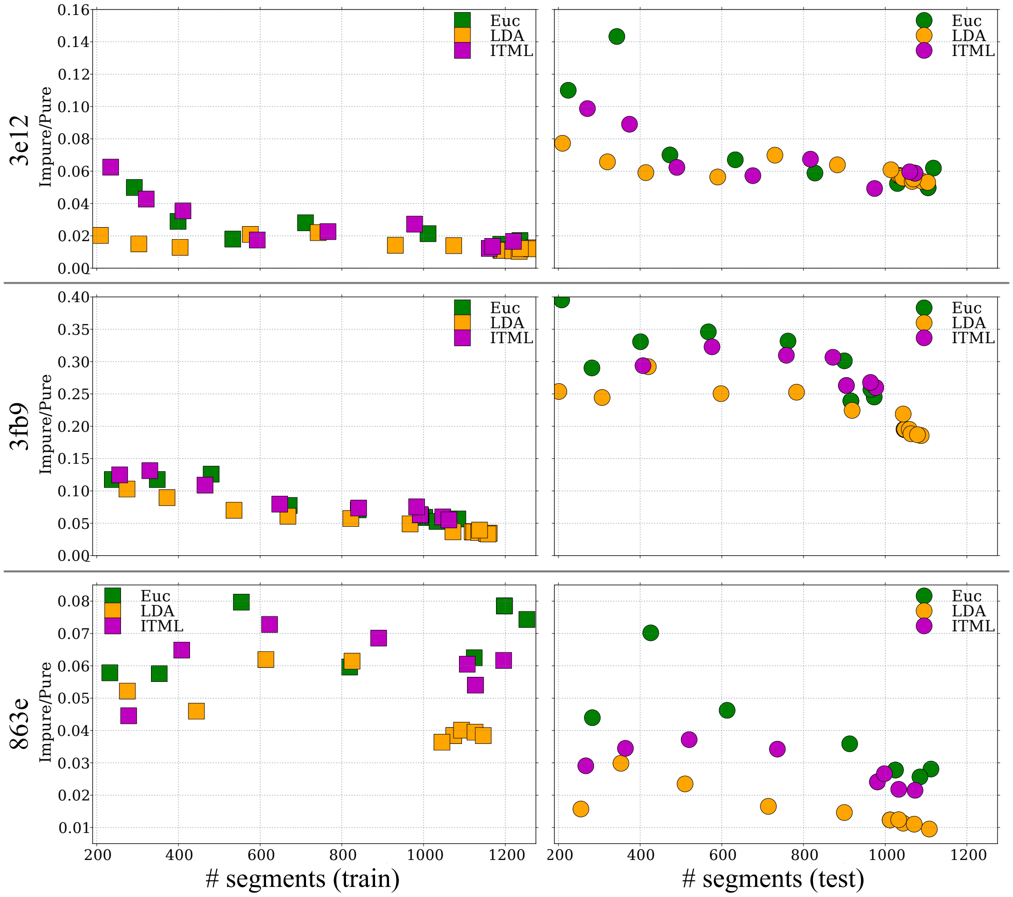


Figure 4.12 : Impurity ratios for EUC (green), LDA (yellow) and ITML (magenta) segmentations vs. number of segments on training (left) and testing (right) images. LDA produces the smallest number of impure superpixels, followed by ITML and EUC.

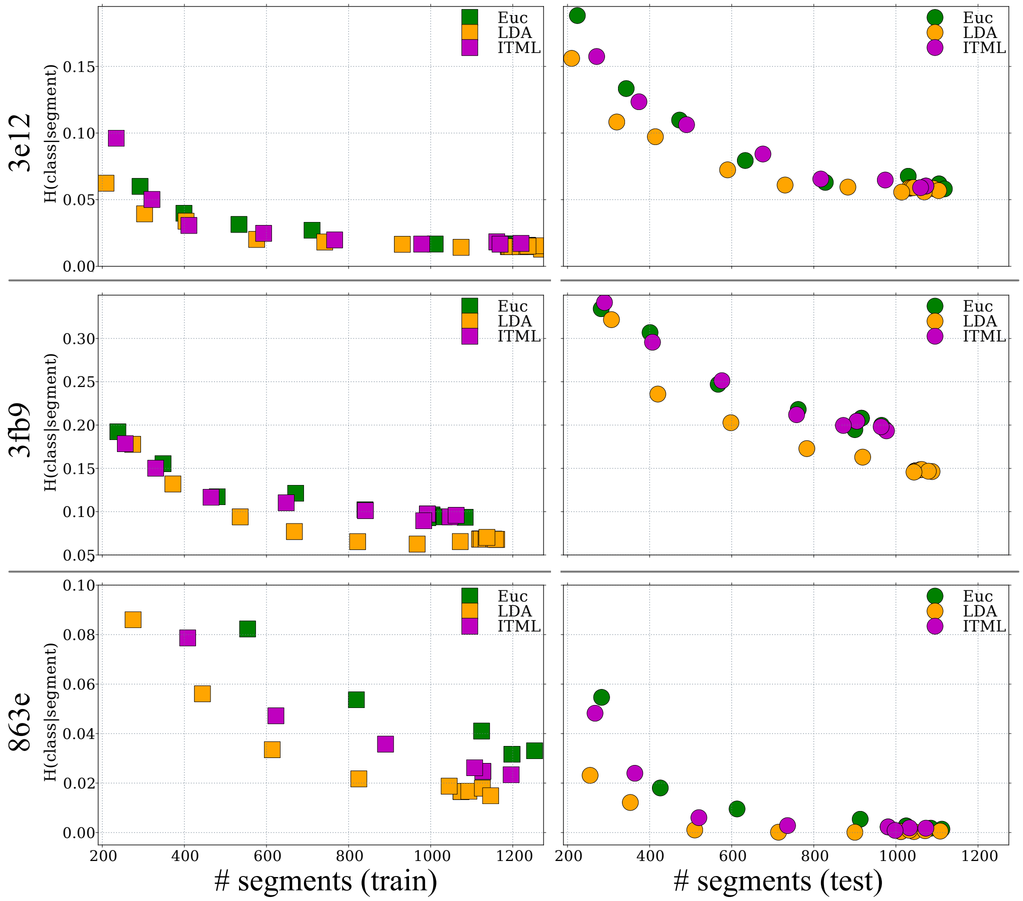


Figure 4.13 : H(class|segment) values for EUC (green), LDA (yellow) and ITML (magenta) segmentations vs. number of segments on training (left) and testing (right) images. As with Figure 4.12, LDA produces the most informative superpixels, followed by ITML and EUC.

Part III

Adaptive Similarity Measures for Inter-domain Material Identification

Chapter 5

Supervised Domain Adaptation

Portions of this chapter are based upon the following publications:

- BD Bue and E Merényi. "Using spatial correspondences for hyperspectral knowledge transfer: evaluation on synthetic data". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* [June 2010].
- BD Bue, E Merényi, and B Csathó. "An Evaluation of Class Knowledge Transfer from Real to Synthetic Imagery". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* [June 2011].

5.1 Inter-domain Material Identification

We have illustrated that hyperspectral image spectra provide ample signal content to distinguish spectrally similar but distinct materials. However, in many practical remote sensing scenarios, we do not have a sufficient quantity of representative samples to train a classifier to reliably classify all materials in a given scene. In such situations, leveraging labeled data captured under similar conditions can be a great resource, but poses significant challenges. In particular, the spectral representations of identical materials differ when they are captured under different conditions (e.g., by different sensors, at different spatial locations, or at different capture times). Consequently, reconciling differences between training (or *source domain*) and test (*target domain*) spectra captured under different conditions is crucial to accurately transfer our existing knowledge of the source domain to predict the materials of spectra in the target domain. These *inter-domain* material identification problems are the focus of this chapter. We begin* with a formal definition of the inter-domain material identification problem, and describe the classification settings we consider. We then introduce our similarity-based domain adaptation framework, RelTrans (**Rel**ational class knowledge **Trans**fer), which calculates a mapping between domains using a set of source spectra to a set of target spectra representing identical materials in both domains. This mapping, applied as a similarity measure, captures structured, relative relationships between classes shared between the source and target domains, allowing us to apply a classifier trained using labeled source domain samples to classify samples from the target domain.

5.2 Domain Adaptation for Multiclass Knowledge Transfer

We apply *domain adaptation* techniques to reconcile the differences between the source and target domains. Formally, we assume we have N^S labeled examples $(X^S, Y^S) = \left\{(\mathbf{x}_i^S, y_i^S)\right\}_{i=1}^{N^S}$, $y_i^S \in \{1, \ldots, K^S\}$ drawn from a source distribution $\mathrm{p}^S(\mathcal{X}, \mathcal{Y})$ to train a predictor to classify N^T samples $X^T = \left\{\mathbf{x}_i^T\right\}_{i=1}^{N^T}$ with unknown labels Y^T drawn from a target distribution $\mathrm{p}^T(\mathcal{X}, \mathcal{Y})$. We assume the unlabeled target samples are available at training time. In some cases, we have a small quantity of labeled target samples available to guide the domain adaptation procedure. We refer to such problems as *supervised domain adaptation* problems. When no labeled target data is available, we refer to the problem as an *unsupervised domain adaptation* problem. Our objective in both cases is train a classifier using the available labeled and unlabeled data to predict labels for the unlabeled target examples.

*The work described in this chapter was performed in collaboration with Erzsébet Merényi, David Thompson and Beáta Csathó, with assistance from Bill Farrand, John Kerekes, Mike Mendenhall, David Pogorzala, Devika Subramanian and Kiri Wagstaff.

In practical inter-domain material identification settings, the number of source classes K^S will often differ from the number of target classes K^T. For instance, if our source and target data are drawn from images of different geospatial regions, there will likely be classes in the target domain that are not present in the source domain, and vice-versa. Even when the source and target data represent the same geospatial region, the underlying scenery itself may have changed between image capture times. Figure 5.1 summarizes the domain adaptation settings we consider in this thesis. In the first setting, we assume that all K^S classes present in the source data are represented in the target data (i.e., $K^S = K^T$). Thus, we can potentially predict accurate labels for all of the target classes, assuming the differences between source and target feature spaces can be adequately reconciled. We refer to this setting as *domain adaptation* (abbreviated *DA*). A special case of the DA setting occurs when the source data contains several classes not present in the target data (i.e., $K^S > K^T$). Here, the extraneous source classes may increase misclassifications if they closely resemble any of the source classes. In the second setting, the target domain data contains samples from all of the source classes, and also contains a number of classes not present in the source domain (i.e., $K^S < K^T$). We refer to this setting as *outlier detection* (*OD*), as it is necessary to detect which target samples represent the *outlier* or *unknown* classes that are absent from the source domain to minimize the number of misclassifications.

5.3 The RelTrans Framework

We now introduce the **Rel**ational class knowledge **Trans**fer, or RelTrans, framework, which allows us to reconcile differences between spectra captured under similar, but not identical, conditions. Our framework is inspired by the Structural Correspondence

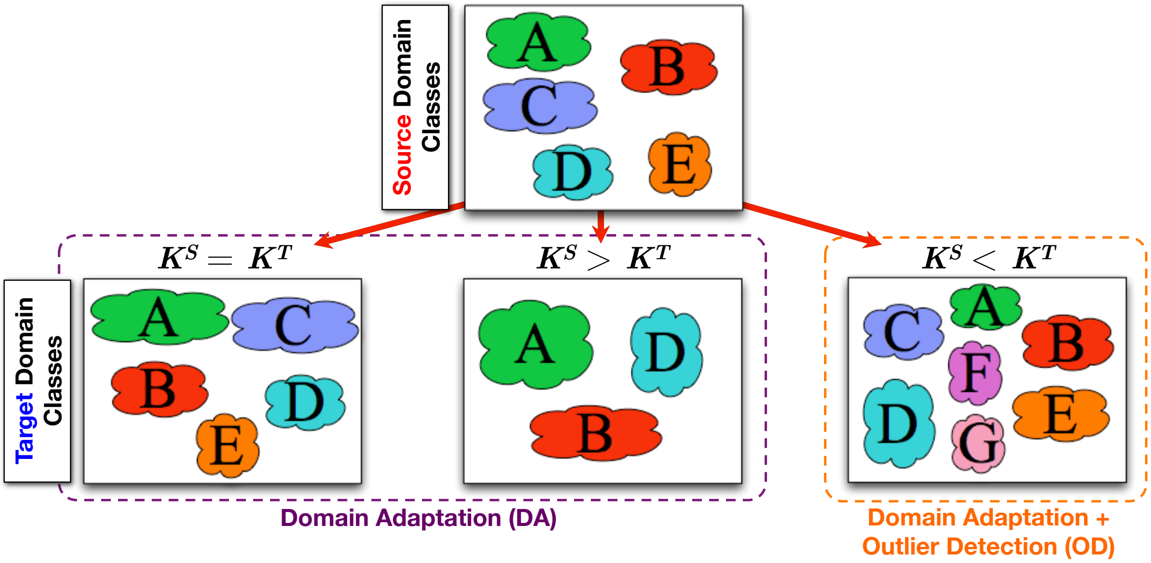


Figure 5.1 : Multiclass class knowledge transfer settings for K^S source classes (top box) vs. K^T target classes (bottom three boxes). In the domain adaptation setting (DA, purple dashed box), we can potentially predict accurate labels for all of the target classes if differences between the source and target feature spaces can be adequately reconciled. In the outlier detection setting (OD, orange dashed box), it becomes necessary to detect samples representing target classes not present in the source domain as *unknowns*.

Learning (SCL) algorithm of Blitzer et al. [2006], which creates a mapping between a set of labeled source domain samples and a set of unlabeled target domain samples using a set of discrete *pivot features* common to both domains. RelTrans extends SCL to feature spaces where the samples we classify are continuous-valued functions (e.g., hyperspectral signatures) by defining the mapping between domains based upon distances to a set of canonical *pivot samples* that represent classes present in both the source and target domains.

Figure 5.2 summarizes the main steps of the RelTrans framework. Formally, RelTrans maps samples $\mathbf{x}^D$, $D \in \{S, T\}$ (S=source, T=target) to a common, relational feature space (the *R-space*) according to distances $\mathrm{d}(\cdot, \cdot)$ to Q paired *pivot* samples

$(P^S, P^T, Y^P) = \left\{(\mathbf{p}_i^S, \mathbf{p}_i^T, y_i^P)\right\}_{i=1}^{Q}$ via the following function:

$$\mathrm{R}(\mathbf{x}^D, P^D) = \left(\frac{\mathrm{d}(\mathbf{x}^D, \mathbf{p}_1^D)}{\sum_{\ell=1}^{Q} \mathrm{d}(\mathbf{x}^D, \mathbf{p}_\ell^D)}, \ldots, \frac{\mathrm{d}(\mathbf{x}^D, \mathbf{p}_Q^D)}{\sum_{\ell=1}^{Q} \mathrm{d}(\mathbf{x}^D, \mathbf{p}_\ell^D)} \right), \tag{5.1}$$

Unless otherwise specified, $\mathrm{d}(\cdot, \cdot)$ is the Euclidean distance. The output of Equation (5.1) is a Q-dimensional vector whose i^{th} element estimates the likelihood of distinguishing sample $\mathbf{x}^D$ from pivot $\mathbf{p}_i^D$ with respect to the other samples in the pivot set P^D.

Figure 5.3 illustrates the effects of applying Equation (5.1) to spectra representing the same material class (i.e., "grass") in the source and target domains. When the relative distances between samples from different classes are approximately preserved across the domains, the R-space mapping captures the multiclass structure common to both domains. This allows us to effectively train a classifier using labeled source samples to classify target samples.

5.4 Related Work

Several recent works propose domain adaptation techniques to reconcile differences between spectra captured under different conditions. Some of these involve active learning techniques, which require user intervention during training to select target samples most relevant to the domain adaptation problem (e.g., [Kim et al., 2008; Persello and Bruzzone, 2011]). While active-learning approaches produce good results, requiring expert intervention during training limits the applicability of the technique for fully-autonomous applications, such as onboard spacecraft, and is often impractical for rapid exploitation of new data. Another approach is to automatically adapt a pre-trained classifier to classify similar imagery (e.g., [Bruzzone and Marconcini, 2010;

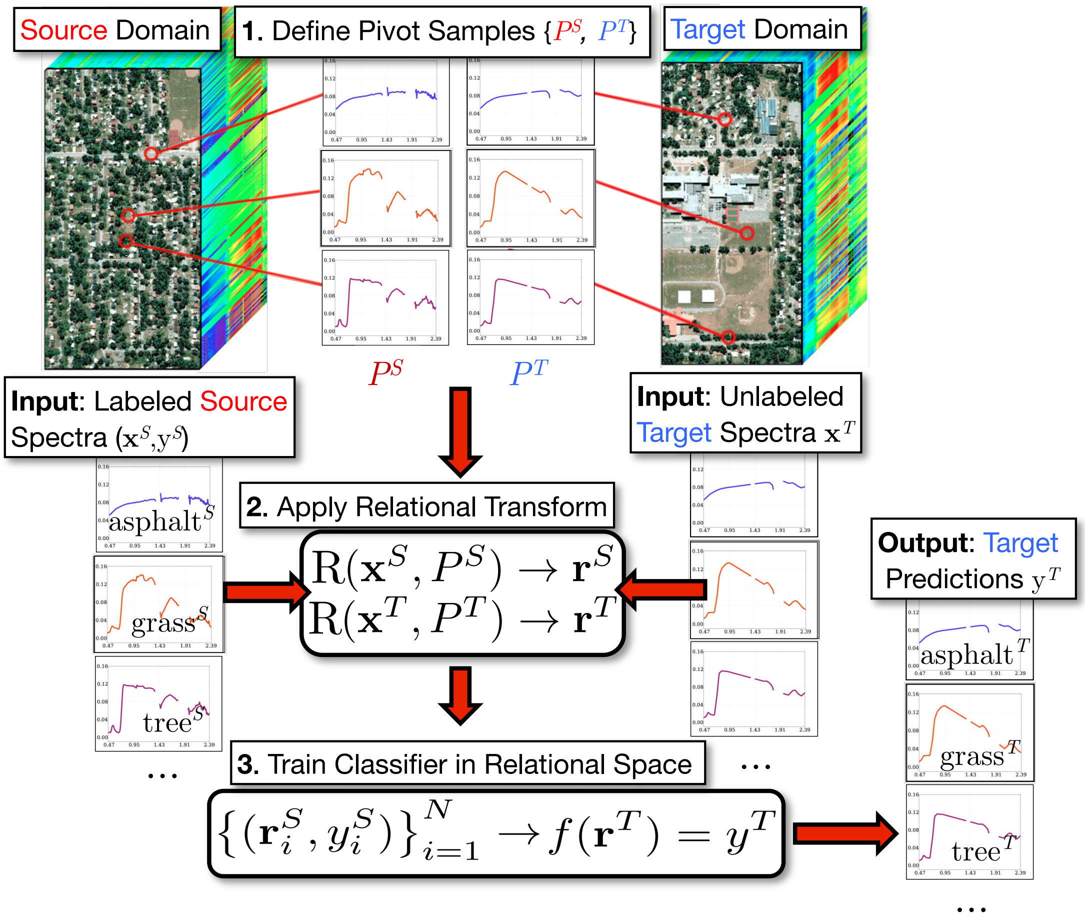

Source Domain
1. Define Pivot Samples $\{P^S, P^T\}$
Target Domain
P^S
P^T
Input: Labeled Source Spectra ($\mathbf{x}^S$,y^S)
Input: Unlabeled Target Spectra $\mathbf{x}^T$
2. Apply Relational Transform
$\mathrm{R}(\mathbf{x}^S, P^S) \rightarrow \mathbf{r}^S$
$\mathrm{R}(\mathbf{x}^T, P^T) \rightarrow \mathbf{r}^T$
asphalt^S
grass^S
tree^S
...
Output: Target Predictions y^T
asphalt^T
grass^T
tree^T
3. Train Classifier in Relational Space
$\{(\mathbf{r}_i^S, y_i^S)\}_{i=1}^N \rightarrow f(\mathbf{r}^T) = y^T$
...
...

Figure 5.2 : Overview of the Relational Class Knowledge Transfer (RelTrans) framework. The set of pivot samples define the mapping to the "R-space" (step 2) between a set of source domain spectra and a set of target domain spectra, typically sampled from two different images. The R-space mapping reconciles systematic differences between the source and target domains, allowing us to train a classifier on the source samples that we can subsequently apply to classify the target samples.

Kim and Crawford, 2010; Rajan et al., 2006]). However, such techniques assume a specific type of classifier has been trained that can subsequently be tuned to the new data.

Domain adaptation problems bear a close resemblance to *multi-task* learning problems [Caruana, 1997], also called *inductive* or *transfer* learning problems. In multi-task learning, labeled samples are available from one or more related source and target *tasks* (i.e., domains), and the goal is to model the underlying structure of the

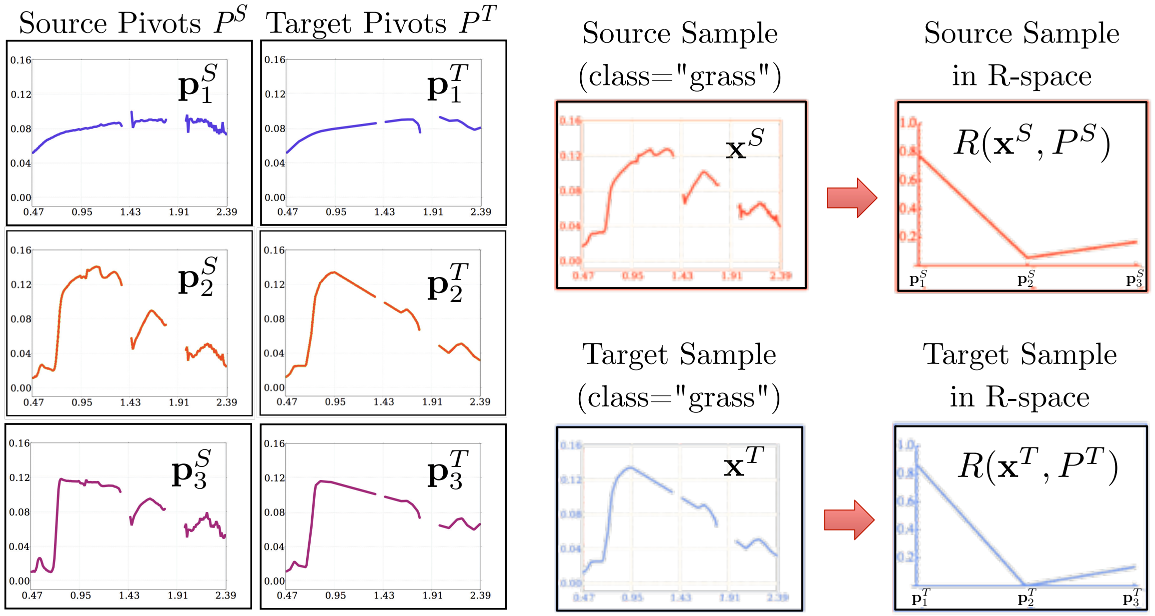


Figure 5.3 : Effect of R-transform with ($Q = 3$) source and target pivots (left six plots) on a source sample (red plots, top right) and a target sample (blue plots, bottom right) representing the same class (i.e., “grass”). Domain-specific differences are reconciled in the R-space than in the original feature space.

tasks to construct a classifier to classify samples from the target task. Supervised domain adaptation problems can potentially be viewed as an instance of the latter case, that is, by viewing the set of labeled target samples as one of the source tasks. We evaluate several widely-used multitask learning techniques in comparison to RelTrans in Section 5.9.2. However, as we show later, an issue with many existing domain adaptation/multitask learning techniques is that they are designed for problems involving two classes (e.g., [Chen et al., 2011; Daume, 2007; Zhen and Li, 2008]), and do not generalize well to multiclass domain adaptation problems.

An alternative to the aforementioned domain adaptation/multitask learning methods is to apply *manifold alignment* techniques to learn a transformation that maps the spectral features of the source and target spectra to a similar feature space. By learning such a mapping, we can apply a classification algorithm of our choosing in the transformed feature space. Yang et al. recently demonstrated that manifold alignment techniques are well-suited to learn such mappings for hyperspectral domain adaptation tasks [Yang and Crawford, 2011]. However, existing manifold alignment techniques learn a single *global* transformation between domains. While applying a global transformation can resolve systematic differences between domains, it may prove inadequate in resolving the *class-specific* differences caused by varying viewing geometries, illumination or atmospheric conditions that alter the radiances observed at the sensor of specific materials [Adams and Gillespie, 2006]. In Section 6.8, we describe an extension to our RelTrans framework aligns the manifolds of the source and target data on a per-class basis, and show our extension outperforms manifold alignment techniques that learn a single global transformation between the domains.

5.5 Class Knowledge Transfer using Labeled Source and Target Data

In this section, we consider inter-domain material identification problems in the supervised domain adaptation setting, where a small number of labeled target samples are available to define the mapping between the domains. We start with an overview describing how we adapt a MinDist classifier trained on data from a source image to classify a similar target spectra. We demonstrate that we significantly improve inter-domain classification accuracy by using RelTrans to map source and target spectra to the R-space, in comparison to classifying the target spectra in their original feature space. We then demonstrate RelTrans generalizes to arbitrary similarity-based classifiers. We provide several case studies demonstrating the effectiveness of RelTrans for domain adaptation and outlier detection tasks on both synthetic and real hyperspectral image data sets, and show that our techniques produce comparable or better performance than several recent multi-task learning algorithms.

5.5.1 RelSim: A RelTrans Proof of Concept

In this section, we describe how we apply the RelTrans framework to adapt a MinDist classifier (Equation (1.2)) trained on source domain data to classify data from a similar target domain data. Our algorithm, RelSim, is given in Algorithm 5.1. RelSim computes an $N^T \times K^S$ similarity matrix $\mathbf{R}$ between the unlabeled target samples X^T and the source classes in the R-space. Given the set of Q pivot samples $P = (P^S, P^T, Y^P) = \left\{(\mathbf{p}_i^S, \mathbf{p}_i^T, y_i^P)\right\}_{i=1}^{Q}$, the algorithm computes the class means for the labeled source samples $\boldsymbol{\mu}_j^S$ and for the source and target pivot samples $\boldsymbol{\mu}_j^{PS}$ and $\boldsymbol{\mu}_j^{PT}$, respectively (Step 1), which are subsequently mapped to the R-space (Step 3). We then map each target sample $\mathbf{x}_i^T$ to the R-space vector $\mathbf{r}_i^T$ using the R-space target pivot means (Step 5), and compute its similarity to the R-space source means $\mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^S)$, weighted by the similarity to the R-space pivot class means ($\mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^{PS})$, $\mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^{PT})$, Step 6). We measure the similarity between samples $\mathbf{r}_i$ and $\mathbf{r}_j$ in the R-space according to

$$\mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i, \mathbf{r}_j) = 1 - \frac{\sqrt{Q}}{2}\|\mathbf{r}_i - \mathbf{r}_j\|, \tag{5.2}$$

where $\|\cdot\|$ is the L^2 norm. The $\mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i, \mathbf{r}_j)$ function yields values in the $[0, 1]$ range that increase with the similarity of $\mathbf{r}_i$ and $\mathbf{r}_j$.[†]

We predict the class label y_i^T of target pixel $\mathbf{x}_i^T$ according to the following decision rule

$$y_i^T = \operatorname*{argmax}_j \mathbf{R}_{i,j}. \tag{5.3}$$

where the $(i, j)^{\text{th}}$ entry of the similarity matrix $\mathbf{R}$ gives the likelihood that target pixel $\mathbf{x}_i^T$ is a member of the j^{th} source class.

[†] In [Bue and Merényi, 2010], we scale $\|\mathbf{r}_i - \mathbf{r}_j\|$ by $^1/_2$, rather than $\sqrt{Q}/2$. However, scaling by $^1/_2$ collapses the range of the $\mathrm{R}_{\mathrm{sim}}$ function to $[1 - {}^1/\sqrt{Q}, 1]$, whereas scaling by $\sqrt{Q}/2$ yields values in the entire $[0, 1]$ range. We use the $\sqrt{Q}/2$ scaling in this work.

Algorithm 5.1 RelSim

```
Input: Labeled source samples $(X^S, Y^S)$, unlabeled target samples $X^T$, pivot set
    $P = (P^S, P^T, Y^P)$
Output: $N^T \times K^S$ similarity matrix between target samples vs. source classes $\mathbf{R}$
$M^S = \{\boldsymbol{\mu}_j^S\}_{j=1}^{K^S}$, $M^{PS} = \left\{\boldsymbol{\mu}_j^{PS}\right\}_{j=1}^{K^S}$, $M^{PT} = \left\{\boldsymbol{\mu}_j^{PT}\right\}_{j=1}^{K^S}$
for $j = 1$ to $K^S$ do
    $\mathbf{r}_j^S = \mathrm{R}(M_j^S, M^S)$, $\mathbf{r}_j^{PS} = \mathrm{R}(M_j^{PS}, M^{PS})$, $\mathbf{r}_j^{PT} = \mathrm{R}(M_j^{PT}, M^{PT})$
    for $i = 1$ to $N^T$ do
        $\mathbf{r}_i^T = \mathrm{R}(\mathbf{x}_i^T, M^{PT})$
        $\mathbf{R}_{i,j} = \mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^S) \cdot \mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^{PS}) \cdot \mathrm{R}_{\mathrm{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^{PT})$
    end for
end for
```

In settings where outlier detection is desirable, we can apply a user-specified confidence threshold $\tau \in [0, 1]$ to Equation (5.3) to detect samples representing target domain classes that are dissimilar from the source domain classes, using the updated decision rule

$$y_i^T = \begin{cases} \underset{j}{\mathrm{argmax}}\ \mathbf{R}_{i,j} & \text{if } \mathbf{R}_{i,j} \geq \tau \\ 0 & \text{otherwise,} \end{cases} \tag{5.4}$$

We flag sample $\mathbf{x}_i^T$ as a member of an unknown class by assigning label $y_i^T = 0$ when $\mathbf{R}_{i,j}$ is not sufficiently similar (i.e., $\mathbf{R}_{i,j} < \tau$) to any of the source classes in the R-space.

5.5.2 Adaptive Outlier Detection with RelThresh

We can potentially predict a good value of τ based upon the relationships between the source and target domain classes captured in the pivot set using our RelThresh algorithm (Algorithm 5.2). The algorithm takes as input the $\mathbf{R}$ matrix produced by the RelSim algorithm (Algorithm 5.1), along with the $Q \times K^S$ matrices $\mathbf{R}^{PS}$ and $\mathbf{R}^{PT}$ matrices that give the $\mathrm{R}_{\mathrm{sim}}$ similarities between each pivot $\mathbf{p}_i^D$ vs. their respective pivot class means M^{PD} according to

$$\mathbf{R}_{i,j}^{PS} = \mathrm{R}_{\text{sim}}(\mathrm{R}(\mathbf{p}_i^S, M^{PS}), \mathbf{r}_j^{PS}) \tag{5.5}$$

$$\mathbf{R}_{i,j}^{PT} = \mathrm{R}_{\text{sim}}(\mathrm{R}(\mathbf{p}_i^T, M^{PT}), \mathbf{r}_j^{PT}), \tag{5.6}$$

where $i \in \{1, \ldots, Q\}$, $j \in \left\{1, \ldots, K^S\right\}$ and M^{PS}, M^{PT}, $\mathbf{r}_j^{PS}$, and $\mathbf{r}_j^{PT}$ are calculated as described in Algorithm 5.1. RelThresh discretizes the range of RelSim similarities between the source pivots and target samples into n_{step} segments, and traverses the range (Steps 2-9) to select the threshold τ_{best} that ensures none of the source or target pivots are flagged as unknowns (Step 6) while correctly classifying the most target pivots (Step 8).

Algorithm 5.2 RelThresh

Input: RelSim similarity matrices $\mathbf{R}$, $\mathbf{R}^{PS}$, and $\mathbf{R}^{PT}$. Total τ steps n_{step}
Output: RelSim threshold τ_{best}

```
 1: $\tau_{\text{max}} = \max(\mathbf{R})$, $\tau_{\text{min}} = \min(\mathbf{R})$, $\tau_{\text{step}} = \frac{\tau_{\text{max}} - \tau_{\text{min}}}{n_{\text{step}}}$, $\tau_{\text{cur}} = \tau_{\text{max}}$, $n_{\text{correct}}^{\text{best}} = -\infty$
 2: while $\tau_{\text{cur}} > \tau_{\text{min}}$ do
 3:     $n_{\text{correct}} = 0$
 4:     for $i = 1$ to $Q$ do
 5:         $j = \underset{j}{\operatorname{argmax}}\, \mathbf{R}_{i,j}^{PT}$
 6:         if $\mathbf{R}_{i,j}^{PS} > \tau_{\text{cur}}$ and $\mathbf{R}_{i,j}^{PT} > \tau_{\text{cur}}$ then $n_{\text{correct}} = n_{\text{correct}} + \mathrm{I}(y_i^P = j)$
 7:     end for
 8:     if $n_{\text{correct}} > n_{\text{correct}}^{\text{best}}$ then $n_{\text{correct}}^{\text{best}} = n_{\text{correct}}$, $\tau_{\text{best}} = \tau_{\text{cur}}$
 9:     $\tau_{\text{cur}} = \tau_{\text{cur}} - \tau_{\text{step}}$
10: end while
```

5.6 Multisensor Material Identification

For our first experiment, we consider a class knowledge transfer problem using state-of-the-art synthetic imagery generated using RIT Digital Imaging and Remote-Sensing

Image Generation (DIRSIG, [Schott et al., 1999]) model. We study a subregion of the RIT "Megascene" [Salvaggio et al., 2005], with 400x400 pixels at 4m/pixel resolution. Spectral responses are modeled after the HYDICE [Basedow et al., 1995] instrument, with 210 bands over 0.4-2.5 μm. We perform atmospheric calibration via the empirical line method using the software package ENVI [RSI, 2008]. For this initial evaluation, we assume the spatial extents of the source and target images partially overlap, which provides a natural means to select pivot samples that correspond to the same material classes between the two images. We extract two spatially overlapping sub-images (*Source* and *Target* in Figure 5.4) from the RIT Megascene. The source image remains at HYDICE spectral resolution, while the target image is downsampled to MASTER [Hook et al., 2001] spectral resolution. Initial experiments using spectral responses modeled after the 128-band HyMap [Cocks et al., 1998] instrument proved trivially classifiable with a simple linear classifier. Thus, we opted for the lower spectral resolution of the MASTER instrument, with 23 bands in the 0.4-2.5 micron range, for our target image. Examples of the HYDICE spectra and their MASTER equivalents are shown in Figure 5.5. We considered the 159 overlapping wavelengths that remained after removing saturated water absorption bands in both images, and then upsampling the MASTER spectra back to the HYDICE wavelengths, using the appropriate FWHM parameters. The overlapping wavelengths are shown in Figure 5.6, and the removed water bands are indicated as gaps in the spectra shown in Figure 5.5. We scale each pixel by its Euclidean norm to account for linear illumination effects.

5.6.1 Evaluation Methodology

We consider the domain adaptation (DA, $K^S = K^T$ and $K^S > K^T$) and outlier detection (OD, $K^S < K^T$) settings described in Section 5.2. In each setting, we use

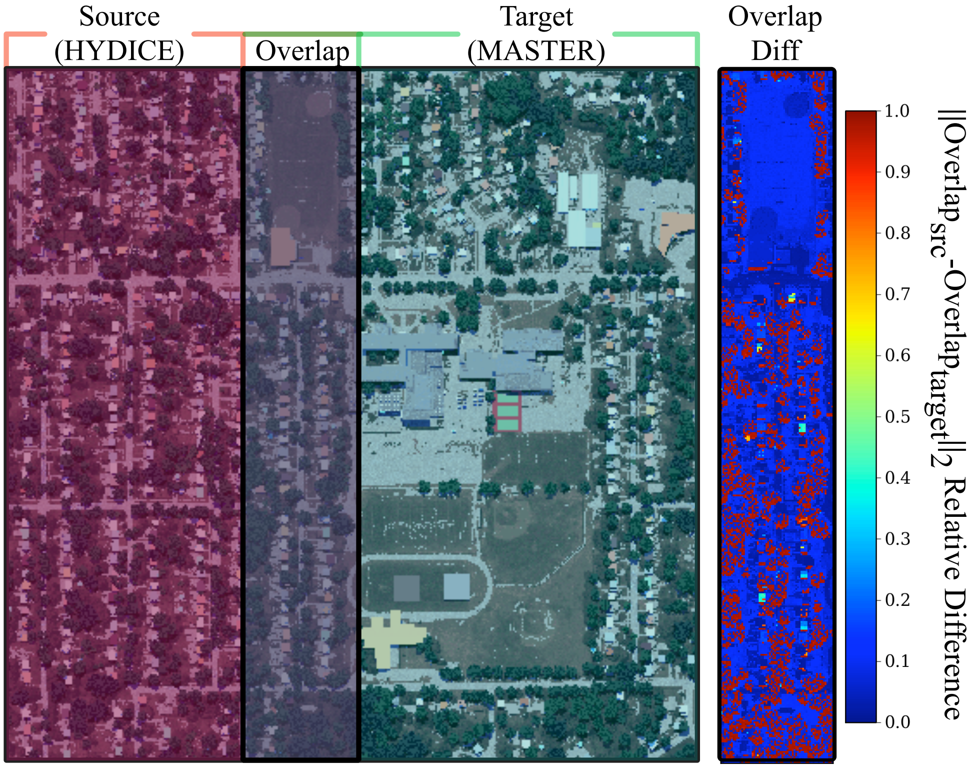

Source
(HYDICE)
Overlap
Target
(MASTER)
Overlap
Diff
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
$||\text{Overlap}_{\text{src}}\text{-Overlap}_{\text{target}}||_2$ Relative Difference

Figure 5.4 : Source (left, red tint), and Target (right, green tint) sub-images of the RIT DIRSIG synthetic image. The source image remains at HYDICE spectral resolution, and the target image is downsampled to MASTER spectral resolution. The target image is then upsampled back to HYDICE spectral resolution. Pivot samples are selected from the overlap region (center, blue tint). The relative difference in Euclidean distances between source and target pixels in the overlap region is also provided (right) and is largest for shadow pixels (Figure 5.5, class C).

the Self-Organizing Map-based clustering described in [Merényi et al., 2009] to guide the extraction of 1000 spectra from each of the source and target images. The mean signatures of the SOM clusters are provided in [Merényi et al., 2009], and the material class labels of the clusters we consider are provided in the second column of Table 5.2, below. An additional 300 labeled spectra are selected as pivot samples by using the labeled source data to pick target samples at identical spatial locations in the overlap region. We distribute the pivot samples evenly over the set of classes shared between the source and target domains, and assume that at least one pivot sample is available

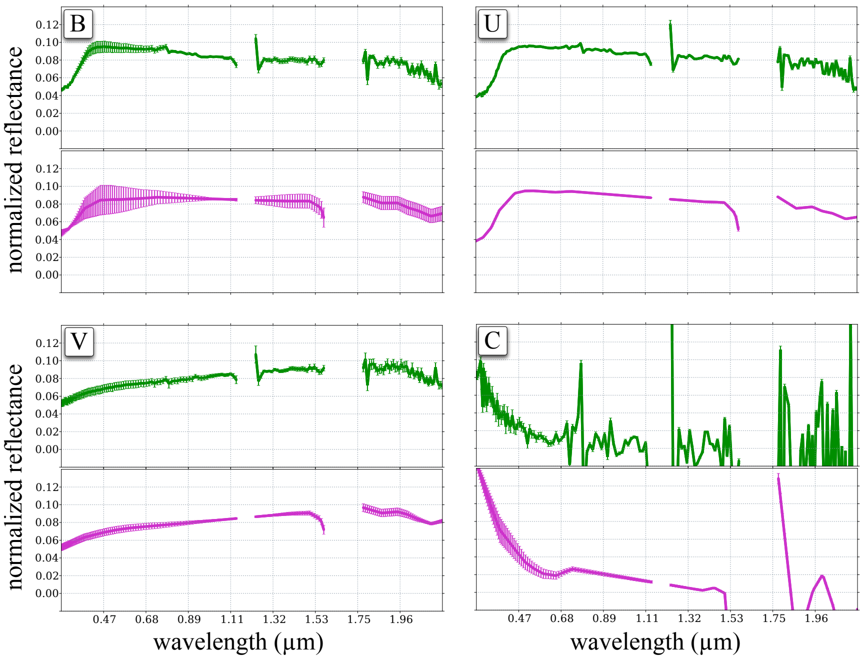


Figure 5.5 : Mean and standard deviation for classes B, U, V and C from source (HYDICE, green) and target (MASTER, magenta) images. Class B consists of a combination of tan asphalt shingle and gray gravel roof spectra, and is often confused with class U (brown asphalt shingles) and class V (black and gray asphalt materials). Class C is an example of a shadow class consisting of several heterogeneous materials. Predictions for such heterogeneous classes tend to be poor due to high intra-class variance.

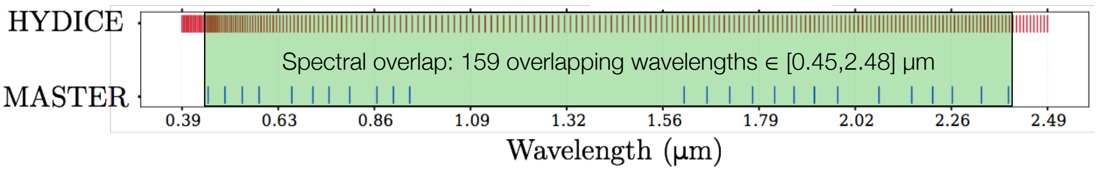


Figure 5.6 : Range of overlapping HYDICE and MASTER wavelengths.

for each class.

The set of classes shared between the source and target domains we consider in each of the DA and OD settings include the following SOM cluster labels: {A, C, J, K, Q, R, U, V, Y, a, c, d, j}. In the $K^S > K^T$ setting, the source data also contains samples from SOM clusters {F, L, h, i}, not present in the target data. In the OD setting (i.e., $K^S < K^T$), we exclude samples from clusters {B, E, M, P, S, k} from the source data. The absence of these samples in the source data forces each classifier to choose the best matching source class when the "true" target class is not present, and thus, the maximum attainable accuracy without applying outlier detection techniques is limited by the number of target samples that represent source classes. In this case, samples from the unknown target classes represent $\approx 30\%$ of the total target samples – and thus, the maximum attainable accuracy without outlier detection is $\approx 70\%$. We consider incorrectly flagged pixels (i.e., flagged target pixels that represent classes present in the source data) misclassifications – i.e., we report the classification accuracy as 100% if all target samples representing source classes are correctly classified, and all unknown samples are correctly flagged.

We compare results using the following classifiers: MinDist– Minimum Euclidean Distance to class means (Equation (1.2)); $\text{MinDist}_{\text{rel}}$ – MinDist applied to image pixels in the R-space using the source class means as pivot samples (i.e., $P^S = P^T = M^S$); $\text{RelSim}_{\text{src}}$ – RelSim algorithm without pivot weighting (i.e., $\mathbf{R}_{i,j} = \text{R}_{\text{sim}}(\mathbf{r}_i^T, \mathbf{r}_j^S)$); RelSim – the "full" RelSim algorithm as described in Algorithm 5.1, $\text{RelSim}_{\text{thresh}}$ – RelSim with ≈15% of the least-confident target predictions flagged as unknowns; and finally, $\text{RelSim}_{\text{RT}}$ – RelSim using the τ calculated by RelThresh. We classify samples in each setting using ten-fold random stratified sampling of the source and target classes, using half of the combined source and target data for training and the remaining half for testing. Target class labels used only for validation purposes, and are not used in training. We report classification accuracy on test predictions only.

5.6.2 Multisensor Material Identification Results

Table 5.1 summarizes our classification results for the DA and OD settings. We see a dramatic performance increase by classifying target spectra in the R-space, rather than in their original feature spaces. We stress that the decision rules for $\text{MinDist}_{\text{rel}}$ and $\text{RelSim}_{\text{src}}$ are functionally equivalent, as a result these classifiers produce equivalent accuracies. The improved accuracies in the R-space is not surprising, since the spectra of identical materials are much less detailed lower-resolution MASTER image spectra than their HYDICE equivalents, particularly at longer wavelengths where downsampling from HYDICE to MASTER spectral resolution causes aliasing (see Figure 5.5 for examples). This reduction in spectral fidelity results in misclassifications using MinDist (85.8% in the $K^S = K^T$ setting) that do not occur with RelSim (94%), as the inter-class relationships in each domain are not significantly altered by the difference in spectral resolution. Thus, by characterizing the multiclass structure within each domain, we are able to form a more robust descriptor for inter-domain comparisons than the pixels themselves (a similar phenomenon was also observed by Rajan et al. in their domain adaptation work [Rajan et al., 2006]). We also observe that the $\mathbf{R}$ matrix weighted using the R_{sim} similarities between the pivot samples (RelSim) yields improved accuracies over RelSim using only the source class means ($\text{RelSim}_{\text{src}}$). In the domain adaptation setting ($K^S = K^T$ and $K^S > K^T$), we acheive the same accuracy using $\text{RelSim}_{\text{thresh}}$ as RelSim, indicating that $\text{RelSim}_{\text{thresh}}$ does not incorrectly flag any target samples as unknowns. In the outlier detection ($K^S < K^T$) setting, we observe an 12% relative improvement in classification accuracy using $\text{RelSim}_{\text{thresh}}$ (74.4% vs. 66.4%), and a 47% relative improvement (97.4% vs. 66.4%) with the automatically-calculated threshold used by $\text{RelSim}_{\text{RT}}$.

We provide the per-class accuracies using RelSim vs. $\text{RelSim}_{\text{thresh}}$ in Table 5.2. Of

	DA		OD
	$K^S = K^T$	$K^S > K^T$	$K^S < K^T$
MinDist	0.858 (1.898e-3)	0.825 (4.922e-4)	0.579 (2.054e-4)
$\textbf{MinDist}_{\textbf{rel}}$	0.947 (3.355e-4)	0.877 (4.197e-4)	0.640 (1.038e-4)
$\textbf{RelSim}_{\textbf{src}}$	0.947 (3.355e-4)	0.877 (4.197e-4)	0.640 (1.038e-4)
RelSim	0.990 (3.454e-6)	0.933 (1.736e-5)	0.664 (2.036e-6)
$\textbf{RelSim}_{\textbf{thresh}}$	0.990 (8.343e-8)	0.933 (1.599e-6)	0.744 (1.210e-7)
$\textbf{RelSim}_{\textbf{RT}}$	0.991 (9.636e-8)	0.933 (1.671e-6)	0.974 (1.210e-7)

Table 5.1 : Mean and standard deviation of classification accuracies for HYDICE (source) vs. MASTER (target) data. The mean and standard deviation of the τ computed by RelThresh for the RelSim$_{\text{RT}}$ classifier are: 0.9689 and 0.0015, respectively. We observe substantial improvements in accuracy over MinDist using RelSim in both the DA and OD settings.

the 303 pixels RelSim$_{\text{thresh}}$ flags as unknowns, 227 are from classes not present in the source data. 113 of these pixels are from class P (red tennis court) and another 113 belong to class E (glass). Both of these classes are fairly dissimilar from the source classes, and are consequently flagged appropriately as unknowns by RelSim$_{\text{thresh}}$. Of the remaining flagged pixels, 34 from class K (green and brown grass) are flagged due to their close similarity to class K (Norway and silver maple trees). Class V has trace elements of gray gravel rooftop spectra (along with several asphalt-based materials), and is often confused with class k (containing only gray gravel rooftop spectra). The pairings of class M (also gray gravel rooftops) with class Q (red weathered stained wood), and class S (gray tarp) with class j (brown mixed brick) are unintuitive, given their respective material compositions. Nonetheless, their spectra are quite similar, even at full HYDICE resolution, and as a consequence are often confused.

More interesting are the results for classes B and C. The material composition of class B (a class not represented in the source data) is a combination of tan asphalt shingles (73.9%) and gray gravel roof (23.6%) spectra. Class U consists entirely of brown asphalt roof shingles, and class V is primarily composed of black (25.6%) and gray (73.9%) asphalt surfaces, with trace elements of gravel rooftop materials. Of

		RelSim						RelSim$_{thresh}$					
Class	**Primary Materials**	**?**	**n**	**PA(%)**	**EO(%)**	**EC(%)**	**CA(%)**	**?**	**n**	**PA(%)**	**EO(%)**	**EC(%)**	**CA(%)**
A	**Roof Shingle, Asphalt, Brown and Red Blend**	0	109	100.0	0.0	50.9	49.1	0	109	100.0	0.0	0.0	100.0
B*	**Roof Shingle, Asphalt, Tan (73.9%), Roof, Gravel, Gray (23.6%)**	0	113	0.0	100.0	0.0	100.0	🟩 1	112	0.0	100.0	0.0	100.0
C	**Shadow Materials**	0	30	100.0	0.0	0.0	100.0	🟧 30	0	100.0	0.0	0.0	100.0
E*	**Glass**	0	113	0.0	100.0	0.0	100.0	🟩 113	0	100.0	0.0	0.0	100.0
J	**Tree, Maple, Silver (46.7%), Norway (53.3%)**	0	113	100.0	0.0	9.6	90.4	0	113	100.0	0.0	2.6	97.4
K	**Grass, Green, Healthy (91.1%), Brown (8.9%)**	0	113	89.4	10.6	0.0	100.0	🟧 34	79	96.2	3.8	0.0	100.0
M*	**Roof, Gravel, Gray (98.9%)**	0	92	0.0	100.0	0.0	100.0	🟥 0	92	0.0	100.0	0.0	100.0
P*	**Tennis court, Playing Surface, Red**	0	113	0.0	100.0	0.0	100.0	🟩 113	0	100.0	0.0	0.0	100.0
Q	**Wood, Stained, Red, Old, Weathered**	0	113	100.0	0.0	45.4	54.6	0	113	100.0	0.0	44.9	55.1
R	**Roof Shingle, Asphalt, Brown, Black, New (86.5%), Roadway Surfaces, Asphalt, Old, Gray (8.7%)**	0	113	100.0	0.0	0.0	100.0	🟧 8	105	100.0	0.0	0.0	100.0
S*	**Gray Tarp**	0	112	0.0	100.0	0.0	100.0	🟥 0	112	0.0	100.0	0.0	100.0
U	**Roof Shingle, Asphalt, Mix Brown**	0	113	100.0	0.0	42.9	57.1	0	113	100.0	0.0	42.6	57.4
V	**Roadway Surfaces, Asphalt, Old, Gray (73.9%), Asphalt, Black, New (25.6%)**	0	113	97.3	2.7	56.0	44.0	🟧 4	109	100.0	0.0	56.2	43.8
Y	**Grass, Brown and Green w/Dirt**	0	113	100.0	0.0	0.0	100.0	0	113	100.0	0.0	0.0	100.0
a	**Roof Shingle, Asphalt, Black, Weathered**	0	110	100.0	0.0	0.9	99.1	0	110	100.0	0.0	0.0	100.0
c	**Sheet Metal, White, Fair (72.8%), Saturn Hood Paint, White (18.5%)**	0	79	100.0	0.0	0.0	100.0	0	79	100.0	0.0	0.0	100.0
d	**Roof Shingle, Asphalt, Black**	0	113	100.0	0.0	50.0	50.0	0	113	100.0	0.0	0.0	100.0
j	**Brick, Siding, Mix Brown, Fair (98.8%)**	0	113	100.0	0.0	49.8	50.2	0	113	100.0	0.0	49.8	50.2
k*	**Roof, Gravel, Gray**	0	112	0.0	100.0	0.0	100.0	🟥 0	112	0.0	100.0	0.0	100.0
Totals		0	2000	**OVR=66.4%, AVG =67.6%, κ=0.6445**				303	1697	**OVR=74.4%, AVG =78.5%, κ=0.7277**			

Table 5.2 : RelSim$_{\text{thresh}}$ class statistics for the outlier detection ($K^S < K^T$) setting before thresholding (left table, under RelSim heading) and after thresholding (right table, under RelSim$_{\text{thresh}}$ heading). ?=Unknown class counts, n=Labeled class counts, PA=producer's accuracy, CA=consumer's accuracy, EO=omission errors, EC=commission errors, OVR=#correct/#samples, AVG=mean producer accuracy, κ=kappa statistic. Classes with a (*) represent outlier/unknown classes, and are not included in the source (training) data. Green cells indicate unknown classes correctly flagged as unknowns, orange cells indicate incorrectly flagged source classes, and red cells indicate unknown classes that are not correctly flagged. Classes B, U, V, and C are shown in Figure 5.5.

the 113 target pixels in class B, the RelSim classifier assigns 84 (74.3%) to source class U, and 28 (24.7%) to source class V, thereby reproducing the true proportions of U and V, with high confidence (i.e., large **R** values). Class C, which is represented the source data, is small (64 pixels), and consists of several materials in shadows – specifically, gray and black asphalt roof shingles (53.1%, 1.6%), brown plank wood siding (18.8%), concrete cinder blocks (23.4%), and dark gray BMW Paint (1.6%). Due to this heterogeneity combined with the relatively large variance caused by the shadow pixels, these pixels receive low **R** scores, and are incorrectly flagged as unknowns as a result.

To demonstrate that thresholding target samples in the R-space yields better performance than in the original feature space, we calculate the number of samples flagged by RelSim$_{\text{thresh}}$ as unknowns, and then flag the same quantity of the least-confident MinDist predictions, referring to this thresholded version of MinDist as MinDist$_{\text{nthresh}}$. Intuitively, by thresholding the same number of samples using the predictions produced by each classifier, the more robust feature space will yield higher classification accuracy. Table 5.3 provides a comparison between MinDist$_{\text{nthresh}}$ and RelSim$_{\text{thresh}}$ using several values of τ in the OD setting. We observe that RelSim$_{\text{thresh}}$ outperforms MinDist$_{\text{nthresh}}$ for each threshold. Also, while the MinDist$_{\text{nthresh}}$ accuracy does improve with increasing τ values, the relative improvements occur more slowly than with RelSim$_{\text{thresh}}$. These results suggest that the source and target spectra are not only better reconciled in the R-space, but also that the target classes are better separated in the R-space than in their original feature space. Additionally, we note that the τ value computed by RelThresh for the RelSim$_{\text{RT}}$ classifier shown in Table 5.1 ($\tau = 0.9689$) yields competitive accuracy (97.4%) to the most accurate τ shown in Table 5.3.

	$\tau = 0.820$	$\tau = 0.892$	$\tau = 0.964$
MinDist$_{\text{nthresh}}$	60.8	73.1	86.3
RelSim$_{\text{thresh}}$	75.0	95.9	98.7

Table 5.3 : Comparison of RelSim$_{\text{thresh}}$ vs. MinDist$_{\text{nthresh}}$ for several values of τ. The RelSim classifier produces more accurate results than MinDist by operating in the R-space.

5.7 Hyperspectral Class Knowledge Transfer

We now consider several class knowledge transfer scenarios using synthetic hyperspectral images captured under different environmental conditions. Our target image for this experiment, which we denote D^1, is the 400x400 pixel, 210 band HYDICE image described in Section 5.6. We select source samples from two different versions of D^2, which is a “cleaner” version of D^1 with reduced atmospheric contamination and fewer shadow pixels. The first, $\mathrm{D}^2_{\mathrm{G}}$, was converted from image radiances to reflectances using the empirical line (ELM) method using the software package ENVI [RSI, 2008]. The second,$\mathrm{D}^2_{\mathrm{B}}$, is a distorted version of D^2 produced by an incorrect atmospheric calibration procedure. Specifically, we applied the EML calibration procedure to D^1 using a relatively absorption-free radiance spectrum paired to a field reflectance spectrum with prominent absorption features. As before, we remove noisy spectral bands in the extreme short and long wavelengths, and also remove the water vapor saturation bands, leaving 160 of the original 210 bands for analysis, and perform illumination normalization by dividing each spectrum by its L^2 norm.

5.7.1 Evaluation Methodology

We mimic the methodology described in Section 5.6 and sample 1000 pixels from each of the source and target images via random stratified sampling, and report the average test classification accuracies over five randomized 50%/50% training/testing splits. We manually select a maximum of 50 pivot samples for each source class that represent the same class in the target image, and match the target class pixels well in terms of spectral shape and absorption features. Figure 5.7 gives the mean signatures of the pivot samples for the six classes shared between the source and target domains that consider in the D^1, $\mathrm{D}^2_{\mathrm{G}}$ and $\mathrm{D}^2_{\mathrm{B}}$ images. We compare the classification accuracies produced by the $\mathrm{RelSim}_{\mathrm{RT}}$ and $\mathrm{MinDist}_{\mathrm{nthresh}}$ classifiers. The mean spectra of the five target classes excluded from the source data in the OD setting, consisting of 44% of the total target samples, are shown in Figure 5.8.

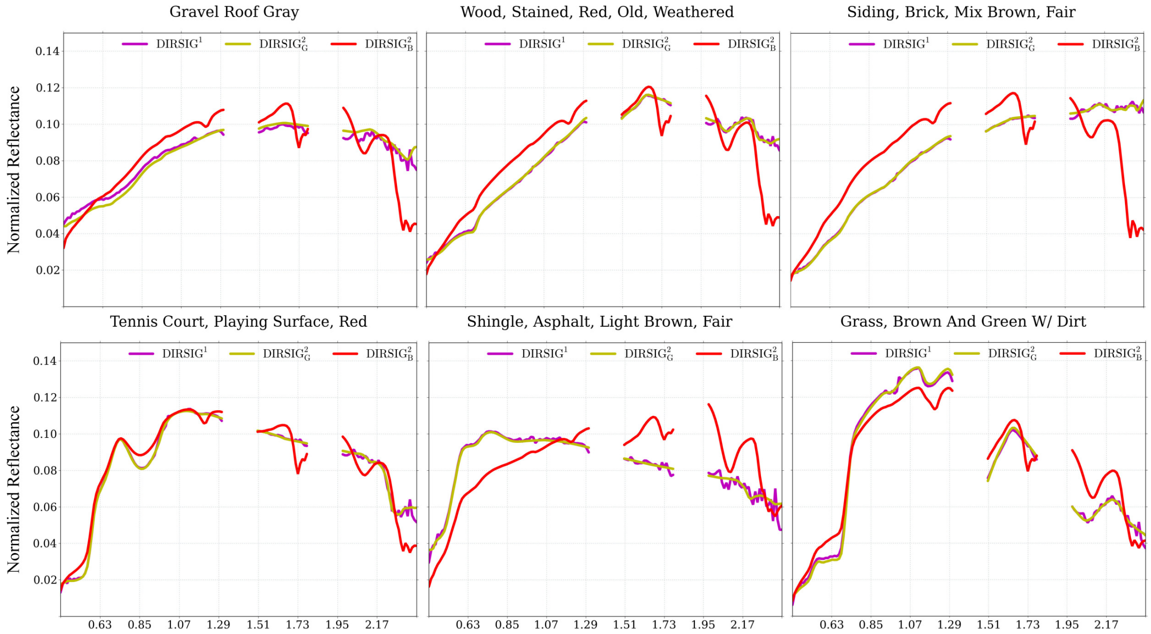


Figure 5.7 : Mean spectra for manually-selected pivot samples between D^1 (magenta), D^2_G (yellow) and D^2_B (red) images. Spectra from images D^1 and D^2_G are similar after ELM atmospheric calibration, while those from the poorly-calibrated D^2_B image are considerably distorted with respect to the D^2_G spectra.

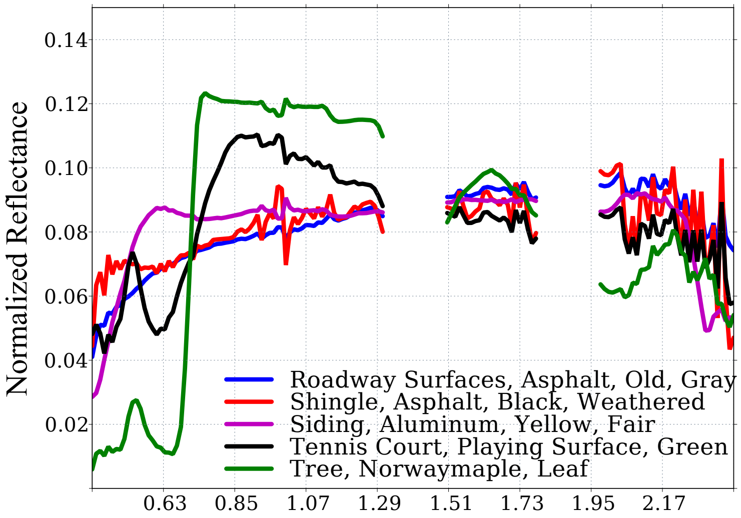


Figure 5.8 : Mean spectra of target classes in D^1 image excluded from the source data representing 44% of the total target samples in the OD setting.

5.7.2 Experimental Results

Table 5.4 summarizes the results in the DA and OD settings using the D^2_G or D^2_B images as source data to classify spectra from the D^1 image. The *intra-image* (i.e., training and testing samples are drawn from the same image) classification accuracies for the three images in each setting are given in the shaded columns. We observe equivalent MinDist and RelSim accuracies in the D^2_G vs. D^1 scenario, which is not surprising, given that the D^2_G and D^1 spectra are nearly identical. However, we observe more substantial improvements in accuracy in the D^2_B vs. D^1 scenario (73% MinDist vs. 100% RelSim). MinDist tends to misclassify samples from the "Siding, Brick, Mix Brown, Fair" class as "Wood, Stained, Red, Old, Weathered" (Figure 5.7, top right and bottom right, respectively). Visually, these classes are similar in image D^2_B, but less so in D^1, and our results suggest that these spectrally-similar materials are difficult to discriminate using the source data without first reconciling their domain-specific differences. We also point out that the RelThresh procedure produces no incorrectly-flagged samples in the DA setting for both of the D^2_G vs. D^1 and D^2_B vs. D^1 cases, as shown by the percentages of correctly-flagged samples in square brackets.

In the OD setting, MinDist performs about 50% worse than RelTrans in the D^2_B vs. D^1 scenario, due to considerable differences between the D^1 and the distorted D^2_B spectra, combined with the presense of the unknown target classes. RelTrans is unaffected by these differences, since the systematic distortions in D^2_B do not significantly alter the relative intra-class distances in the source and target images. As a result, the RelSim classification accuracy without thresholding is optimal (56%) regardless of whether D^2_G or D^2_B is used as source data. With outlier detection, we observe a substantial MinDist improvement in classification accuracy using both MinDist (24 → 50) and RelSim (56 → 93). However, RelSim correctly flags 100% of

	DA ($K^S = K^T$)					**OD** ($K^S < K^T$)				
	D^1	D^2_G	D^2_B	D^2_G vs. D^1	D^2_B vs. D^1	D^1	D^2_G	D^2_B	D^2_G vs. D^1	D^2_B vs. D^1
MinDist	99	99	84	99	73	99	96	83	56	24
MinDist$_{\mathbf{nthresh}}$	99	99	84	99 [100]	73 [100]	99	96	83	93 [92]	50 [17]
RelSim	99	99	84	100	100	99	96	83	56	56
RelSim$_{\mathbf{RT}}$	99	99	84	100 [100]	100 [100]	99	96	83	99 [100]	99 [100]

Table 5.4 : Domain adaptation ($K^S = K^T$) and outlier detection ($K^S < K^T$) results for source images D^2_G and D^2_B vs. target image D^1. For each of the DA and OD settings, the intra-image classification accuracies for each of the D^1, D^2_G and D^2_B images are given in the shaded columns. Values in square brackets give the percentage of correctly flagged unknown samples.

the samples representing unknown classes, whereas MinDist only flags 92% and 17% of the unknown samples correctly in the D^2_G vs. D^1 and D^2_B vs. D^1 cases, respectively.

5.8 Synthetic to Real Class Knowledge Transfer

We now assess the feasibility of predicting the materials of spectra from a real low-altitude AVIRIS target image using a classifier trained using samples from the synthetic D^2_G image. We consider the AVIRISLA image of Ocean City, MD, initially described in Section 2.4.3, denoted AVIRIS$_\mathrm{OC}$. We select a set of six clusters from the AVIRIS$_\mathrm{OC}$ cluster map described in [Merényi et al., 2007] that correspond well to in terms of spectral characteristics and expert field-knowledge to materials present in the D^2_G image. Specifically, we match cluster C to "Tennis Court, Playing Surface, Green," G to "Roadway Surfaces, Asphalt, Old, Gray," U to "Roof Shingle, Asphalt, Brown and Red Blend, Fair," L to "Grass, Brown and Green w/ Dirt," T to "Roof, Gravel, Gray," and f to "Wood, Stained, Red, Old, Weathered." Note that we select these matches based on expert knowledge and spectral characteristics of materials, and *not the objects to which the materials belong.* For instance, we know from field knowledge that segment G is made of rooftop materials with spectral features indicating the presense of asphalt, and we pair it with the “Roadway Surfaces, Asphalt Old Gray” synthetic image class, as they share the same material composition.

5.8.1 Evaluation Methodology

We apply the same methodology as in Section 5.7.1, sampling 1000 pixels from each of the D^2_G and D^2_B images. We manually select 50 pivot samples for each source class. The mean spectra of the pivot samples are shown in Figure 5.9. The mean spectra of the five unknown target classes in the OD setting excluded from the source data, consisting of 46% of the total target data, are shown in Figure 5.10.

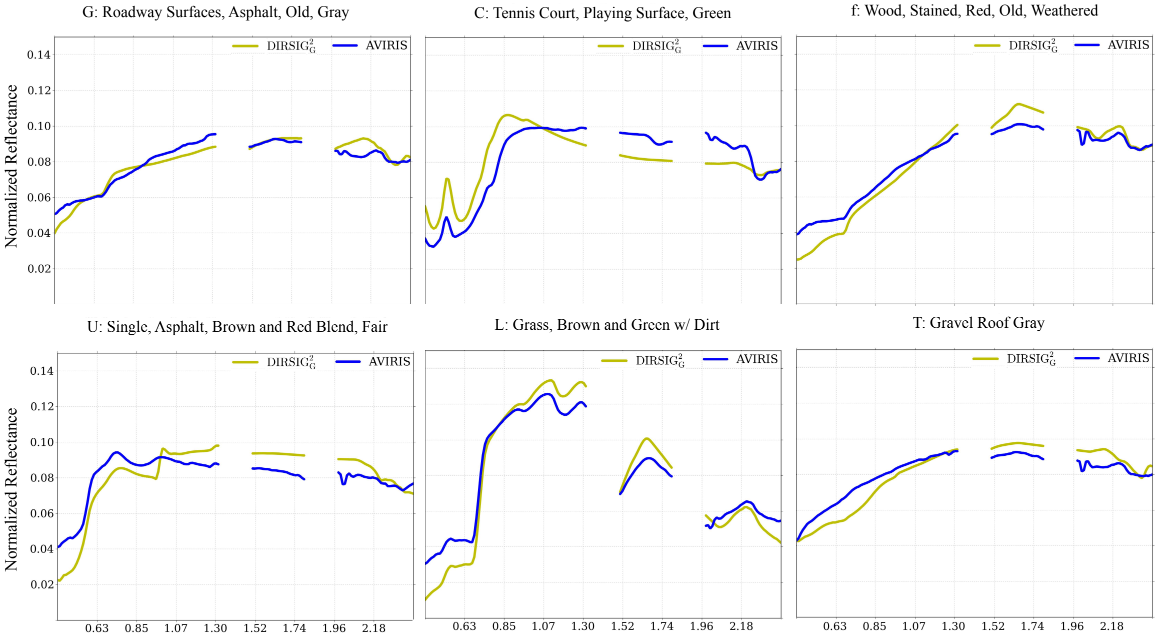


Figure 5.9 : Mean spectra for pivot samples between the D^2_G (yellow), and $\text{AVIRIS}_{\text{OC}}$ (blue) images.

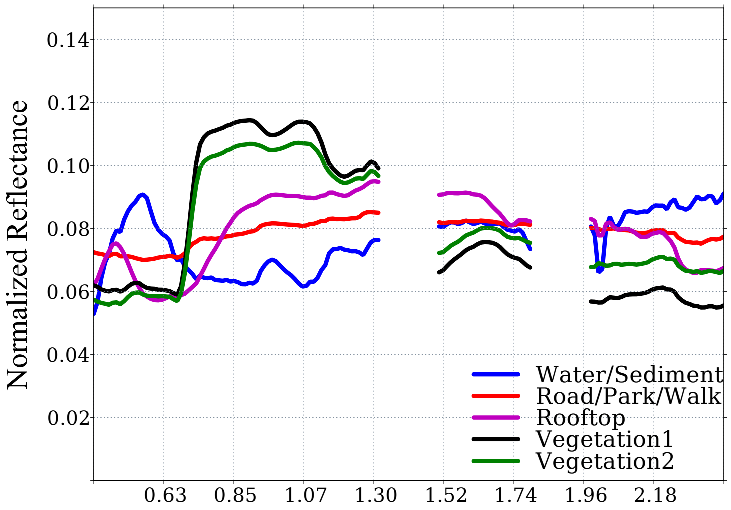


Figure 5.10 : AVIRIS$_{\mathrm{OC}}$ class mean spectra excluded from source classes in the outlier detection setting. These spectra represent 46% of the total target samples in the OD setting.

5.8.2 Experimental Results

Table 5.5 provides the overall accuracies for the DA and OD settings, respectively. We see similar trends in classification accuracy as we observed on the synthetic data with both MinDist and RelSim. In particular, we observe considerable improvements with RelSim over MinDist in the DA (72%s vs. 45%) and OD (43% vs. 26%) settings. We observe a 138.7% relative improvement with RelSim$_{\mathrm{RT}}$ over MinDist$_{\mathrm{nthresh}}$. Notably, RelSim$_{\mathrm{RT}}$ achieves comparable classification accuracy in the outlier detection setting to the domain adaptation setting (74% vs. 72%, respectively) by correctly flagging 100% of the unknown samples.

The per-class accuracies for MinDist and RelSim in the DA setting are shown in Table 5.6. We observe that the RelSim misclassifications are generally more intuitive than MinDist. For instance, MinDist misclassifies all "T: Roof, Gravel, Gray" samples as either "f: Wood, Stained, Red, Old, Weathered", "G: Roadway Surfaces, Asphalt, Old, Gray", or "C: Tennis Court, Playing Surface, Green," whereas RelTrans correctly

	DA ($K^S = K^T$)			**OD** ($K^S < K^T$)		
	$\mathrm{D}^2_{\mathrm{G}}$	$\mathrm{AVIRIS}_{\mathrm{OC}}$	$\mathrm{D}^2_{\mathrm{G}}$ vs. $\mathrm{AVIRIS}_{\mathrm{OC}}$	$\mathrm{D}^2_{\mathrm{G}}$	$\mathrm{AVIRIS}_{\mathrm{OC}}$	$\mathrm{D}^2_{\mathrm{G}}$ vs. $\mathrm{AVIRIS}_{\mathrm{OC}}$
MinDist	98	83	45	92	82	26
MinDist$_{\mathbf{nthresh}}$	98	83	45 [100]	92	82	31 [77]
RelSim	98	83	72	92	82	43
RelSim$_{\mathbf{RT}}$	98	83	72 [100]	92	82	74 [100]

Table 5.5 : Domain adaptation ($K^S = K^T$) and outlier detection ($K^S < K^T$) results for source image $\mathrm{D}^2_{\mathrm{G}}$ vs. target image $\mathrm{AVIRIS}_{\mathrm{OC}}$ before/after thresholding. For each of the two settings, the intra-domain classification accuracies for each of the $\mathrm{AVIRIS}_{\mathrm{OC}}$ and $\mathrm{D}^2_{\mathrm{G}}$ images are given in the shaded columns. Values in square brackets give the percentage of correctly flagged unknown samples. No target samples belong to unknown classes in the domain adaptation ($K^S = K^T$) setting (and thus, no samples should be flagged), while 461 of the 1000 target samples are unknowns in the outlier detection setting ($K^S < K^T$).

classifies 35% of those same pixels, with the remaining misclassifications falling into classes G and f, but not the (spectrally dissimilar) class C.

	Accuracy (%)	
Cluster Label: Material Class	**MinDist**	**RelSim**
C: Tennis Court, Playing Surface, Green	**7**	**100**
G: Roadway Surfaces, Asphalt, Old, Gray	**55**	**26**
L: Grass, Brown and Green w/ Dirt	**63**	**93**
T: Gravel Roof Gray	**0**	**100**
U: Shingle, Asphalt, Brown and Red Blend, Fair	**57**	**100**
f: Wood, Stained, Red, Old, Weathered	**28**	**83**

Table 5.6 : Per-class accuracy (%) for $\mathrm{D}^2_{\mathrm{G}}$ source classes and $\mathrm{AVIRIS}_{\mathrm{OC}}$ target classes in the domain adaptation setting. MinDist produces poor performance due to substantial differences between the source and target feature spaces.

Table 5.7 gives the per-class percentages of unknown samples that each classifier correctly flags as unknowns. We see relatively good outlier detection performance with both MinDist and RelSim, as the unknown target classes are reasonably dissimilar from the source classes. However, MinDist regularly misclassifies the $\mathrm{AVIRIS}_{\mathrm{OC}}$ class “Vegetation2” as “Tennis Court, Playing Surface, Green” due to their similar absorption features. MinDist also often misclassifies “Road/Park/Walkway” and

"Rooftop" pixels as the "Roadway Surfaces, Asphalt, Old, Gray" D_G^2 material class, which are misclassifications that RelSim correctly resolves, despite their similar material compositions.

	Outlier class in AVIRIS$_{\mathbf{OC}}$ image				
	Water/Sediment	Road/Park/Walk	Rooftop	Vegetation1	Vegetation2
MinDist$_{\mathbf{nthresh}}$	100	90	78	100	44
RelSim$_{\mathbf{RT}}$	100	100	100	100	100

Table 5.7 : Percentage of correctly flagged unknowns for the D_G^2 vs. AVIRIS$_{OC}$ OD setting. Both classifiers yield good performance, but RelSim flags a higher percentage of unknown pixels correctly.

5.9 Radiance vs. Reflectance Classification

The majority of hyperspectral image classification techniques consider atmospherically-corrected reflectance spectra. This is largely motivated by the fact that reflectance signatures provide a dimensionless frame-of-reference that can be directly compared to lab-measured spectral reflectance signatures. To convert a given target image from at-sensor radiance measurements to reflectance units, it is necessary to apply atmospheric calibration techniques requiring ground-measured reflectance signatures from the scene under investigation, or computationally intensive radiative transfer modeling techniques. However, if we have access to labeled reflectance spectra for the materials we wish to classify in our target image, we can potentially avoid the process of calibrating the target image by mapping the radiance and reflectance spectra to a similar feature space using RelTrans. This can be of great advantage when such ground-measured reflectance spectra from the target scene are unavailable, and/or when applying radiative-transfer modeling techniques is not computationally feasible. We can also apply the same methodology to classify atmospherically-calibrated target spectra using labeled radiance spectra, potentially enabling faster exploitation of newly-captured radiance imagery.

To demonstrate this capability, we use RelTrans to classify spectral signatures in radiance units using a classifier trained using reflectance signatures of identical materials. We refer to this scenario as RAD2RFL. In our second scenario, RFL2RAD, our goal is to classify target spectra in radiance units using atmospherically-calibrated source spectra in reflectance units. Our data consists of spectral signatures from a set of ten distinct materials that reflect the diversity in a typical urban material identification problem, sampled from the DIRSIG synthetic HYDICE image D^2 (described in Section 5.7). We denote the atmospherically-calibrated version of the D^2 image in reflectance units as $\mathrm{D}^2_{\mathrm{RFL}}$, and its uncalibrated counterpart in radiance units as $\mathrm{D}^2_{\mathrm{RAD}}$. As described in Section 5.7, we use the ELM atmospheric calibration technique to calibrate the $\mathrm{D}^2_{\mathrm{RFL}}$ image. We use the ground-truth labels to sample a set of 400 "pure" pixels (i.e., unmixed pixels consisting of a single material) for each class at identical spatial locations in the $\mathrm{D}^2_{\mathrm{RFL}}$ and $\mathrm{D}^2_{\mathrm{RAD}}$, selecting the pixels nearest to their respective class means in the $\mathrm{D}^2_{\mathrm{RFL}}$ image. This filtering step is necessary to exclude pixels representing multiple materials, or those that are excessively noisy or in shadows. The resulting class means for the $\mathrm{D}^2_{\mathrm{RAD}}$ and $\mathrm{D}^2_{\mathrm{RFL}}$ data are shown in Figure 5.11. In both scenarios, we assume a small number of labeled target spectra Q_k are available for each class to help reconcile differences between the source and target domains. For simplicity, we select the top Q_k pixels nearest their class means in each of the source and target domains to form the pivot sets P^S and P^T.

5.9.1 RelTrans with Different Classifiers

So far, we have focused on domain adaptation results using the RelSim classifier, which can be viewed as a thresholded version of MinDist applied in the R-space.

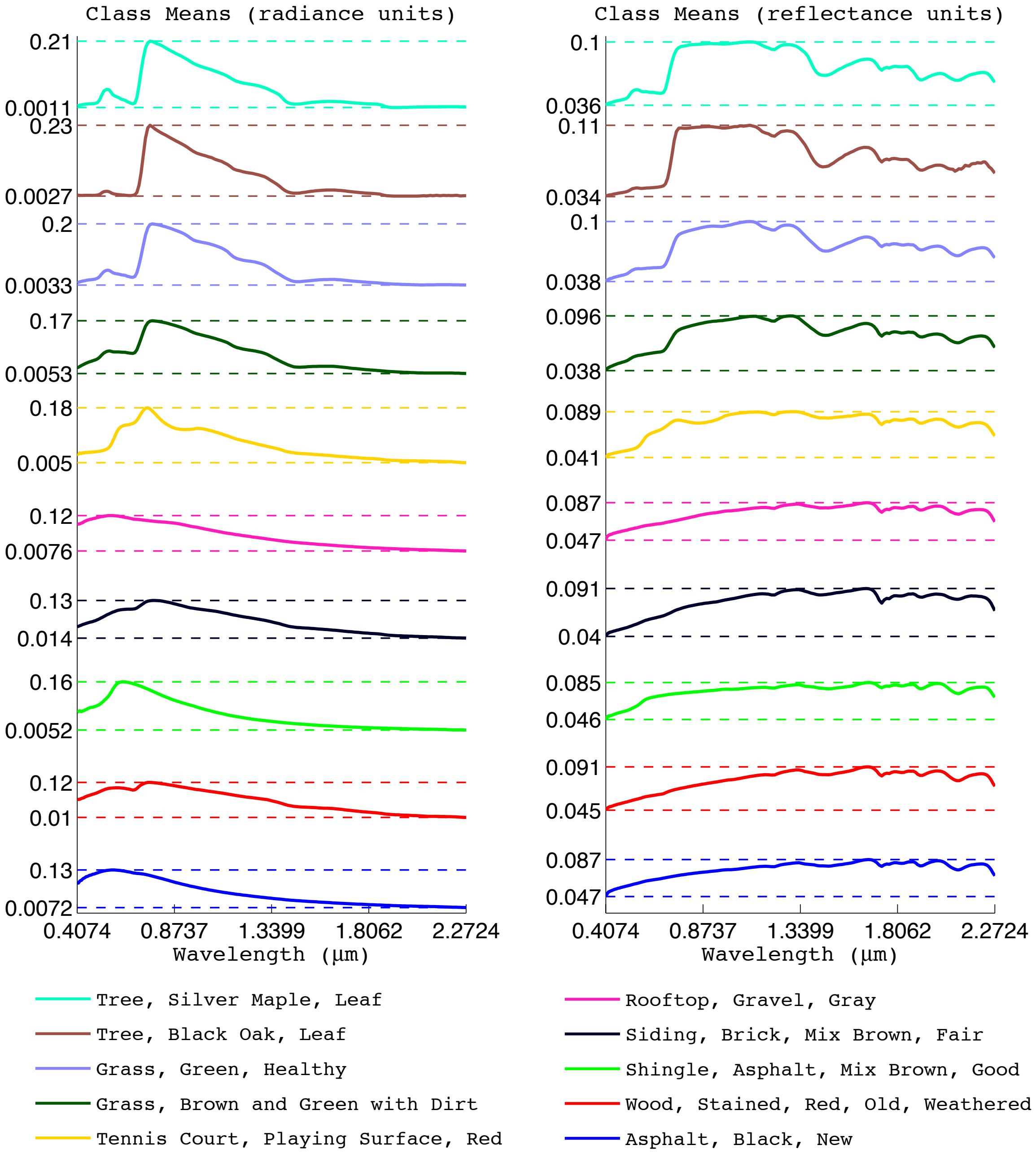


Figure 5.11 : L^2-normalized class means from image D^2 in radiance (left) vs. reflectance (right) units. Y-axis tick marks give the minimum and maximum value for each spectrum in each class.

In this section we demonstrate that the R-space mapping is robust to the choice of classification algorithm by using several different multiclass classification algorithms to generate predictions for the target spectra in the R-space. Specifically, we map each sample $\mathbf{x}^D$ to the R-space using its corresponding pivot set P^D, for $D \in \{S, T\}$, according to Equation (5.1). We then train a multiclass classifier using the source samples in the R-space, and use the classifier to predict labels for the target samples in the R-space.

As before, our baseline multiclass classifier is MinDist. We also consider the multiclass (one-vs.-one) Support Vector Machine provided in the LIBSVM package [Chang and Lin, 2011] with linear (SVM-lin) and radial basis function (SVM-rbf) kernels, along with the GLVQ and GRLVQ algorithms described in Section 1.4, using the implementation provided by Strickert[‡] [Strickert, 2011]. We select the SVM slack parameter C, and the SVM-rbf kernel width parameter γ from the range $\{10^{-4}, \ldots, 10^{4}\}$; the number of GLVQ/GRLVQ prototypes per class n_{proto} from $\{1, 3, 5, 10\}$; and the steepness parameter for the GLVQ/GRLVQ logistic function σ from $\{1, 25, 50, 100, 250\}$, that produce the highest accuracy on the training data. To balance the amount of computation time while also characterizing generalization performance, we report the average test accuracy over five cross-validation folds. In addition to evaluating classification accuracy in the R-space for $Q_k \in \{10, 25, 50, 75, 100\}$, we provide the baseline intra-domain (denoted RAD and RFL, respectively) and inter-domain (denoted *Base*) classification accuracies acheived by applying each classifier in the original RAD and RFL feature spaces.

Table 5.8 gives the mean and standard deviation of classification accuracies in the RAD2RFL and RFL2RAD scenarios using each classifier. The *Overall* column gives the mean of RAD2RFL and RFL2RAD accuracies for each value of Q_k. The mean of the R-space accuracies over the range of Q_k for each classifier are also provided. We

[‡] Available at: http://mloss.org/software/view/323/

observe extremely low baseline classification accuracies (shaded cells) in both scenarios due to the considerable difference between the source and target feature spaces. This difference between feature spaces typically forces the classifier to predict almost all of the target samples belong to two source classes representing $\approx 20\%$ of the total target samples (e.g., SVM-lin/SVM-rbf in the RAD2RFL scenario), or a single source class representing $\approx 11\%$ of the target samples (e.g., MinDist/GLVQ/GRLVQ in both scenarios, in the RFL2RAD scenario).

We see that classifying the spectra in the R-space yields considerable improvements over the baseline accuracies, often acheiving accuracies comparable to or better than the respective intra-domain accuracies. MinDist, in particular, performs substantially better in the R-space than in the original feature space, achieving accuracies $\approx$4-8% better than in the original feature space, due to the additional structure that the classifier can exploit in the R-space. Overall, the SVM-lin and SVM-rbf classifiers produce the highest average accuracies in the R-space, owing to their good performance $(85-92\%)$ for the smallest $Q_k = 10$ value, where the other algorithms occasionally yield relatively low $(76-83\%)$ accuracies. However, classification accuracy remains reasonably high $(> 85\%)$ and stable using all of the algorithms with a sufficiently large $Q_k(\approx 25)$, as evidenced by the typically small $(< 1\%)$ standard deviation of the classification accuracies, and by the relative differences between the accuracies of adjacent Q_k values. The typically lower accuracies in the RFL2RAD scenario in comparison to the RAD2RFL scenario, combined with the lower intra-domain RFL accuracies, suggest that the RFL2RAD scenario is the more challenging of the two domain adaptation problems. However, the accuracies in the domain adaptation scenarios are reasonably close (within 5%) to one another for each Q_k value. This suggests that the quality of the pivot set has a similar effect on the classification accuracy independent of whether the RAD or the RFL is used as the training data.

		RAD2RFL		**RFL2RAD**		**Overall**	
	Q_k	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
MinDist	Base	0.1123	0.0087	0.1123	0.0008	0.1123	0.0048
RAD=0.8254	10	0.8880	0.0036	0.8476	0.0187	0.8678	0.0112
RFL=0.8195	25	0.8908	0.0028	0.8838	0.0254	0.8873	0.0141
	50	0.8908	0.0027	0.8653	0.0111	0.8781	0.0069
	75	0.8919	0.0076	0.8608	0.0120	0.8764	0.0098
	100	0.8903	0.0123	0.8597	0.0024	0.8750	0.0074
	Mean	*0.8904*	0.0058	0.8634	0.0139	0.8769	0.0099
GLVQ	Base	0.1123	0.0000	0.1123	0.0000	0.1123	0.0000
RAD=0.9102	10	0.8701	0.0075	0.8262	0.1148	0.8482	0.0612
RFL=0.9068	25	0.8962	0.0039	0.8737	0.0150	0.8850	0.0095
	50	0.8757	0.0377	0.8681	0.0056	0.8719	0.0217
	75	0.8869	0.0108	0.8585	0.0112	0.8727	0.0110
	100	0.8701	0.0273	0.8611	0.0012	0.8656	0.0143
	Mean	0.8798	0.0174	0.8575	0.0296	0.8687	0.0235
GRLVQ	Base	0.1123	0.0000	0.1123	0.0000	0.1123	0.0000
RAD=0.9107	10	0.8285	0.0241	0.7569	0.0271	0.7927	0.0256
RFL=0.8978	25	0.9001	0.0080	0.8793	0.0063	0.8897	0.0072
	50	0.8900	0.0199	0.8684	0.0138	0.8792	0.0169
	75	0.8779	0.0211	0.8706	0.0004	0.8743	0.0108
	100	0.8976	0.0075	0.8585	0.0182	0.8781	0.0129
	Mean	0.8788	0.0161	0.8467	0.0132	0.8628	0.0146
SVM-lin	Base	0.2024	0.0076	0.1123	0.0000	0.1574	0.0038
RAD=1.0000	10	0.8658	0.0007	0.8869	0.0035	0.8764	0.0021
RFL=0.9941	25	0.8863	0.0084	0.8844	0.0008	0.8854	0.0046
	50	0.8729	0.0012	0.8796	0.0076	0.8763	0.0044
	75	0.8720	0.0063	0.8768	0.0019	0.8744	0.0041
	100	0.8740	0.0036	0.8754	0.0095	0.8747	0.0066
	Mean	0.8742	0.0040	*0.8806*	0.0047	*0.8774*	0.0044
SVM-rbf	Base	0.2063	0.0011	0.1123	0.0000	0.1409	0.0266
RAD=1.0000	10	0.8571	0.0234	0.9183	0.0067	0.8877	0.0151
RFL=0.9962	25	0.8908	0.0170	0.8437	0.0670	0.8673	0.0420
	50	0.8978	0.0047	0.8939	0.0096	0.8959	0.0072
	75	0.8931	0.0020	0.9032	0.0027	0.8982	0.0024
	100	0.8726	0.0024	0.8877	0.0112	0.8802	0.0068
	Mean	*0.8823*	0.0099	*0.8894*	0.0194	*0.8858*	0.0147

Table 5.8 : Mean and standard deviation of classification accuracy using different multiclass classifiers to classify RAD2RFL and RFL2RAD data in the original source and target feature spaces (Base, shaded cells) vs. in the R-space with $Q_k \in \{10, 25, 50, 75, 100\}$ labeled pivots/class. The mean accuracy over the range of Q_k values and the intra-domain accuracies on the RAD and RFL data are provided for each classifier. The best and second best performing classifiers are given in red and blue italics for each scenario. All of the R-space classifiers produce considerable improvements over their respective baseline accuracies, and show competitive performance to one another, with overall accuracies differing by $< 2\%$.

5.9.2 Comparisons to Multitask Learning Techniques

We also provide a comparison between classifying spectra using RelTrans vs. using MultiTask Learning (MTL) techniques. Given the set of N^{t} *tasks* (i.e., domains) $\{(X_i, Y_i)\}_{i=1}^{N^{\mathrm{t}}}$, each consisting of N_i^{t} samples $\{(X_{i,j}, Y_{i,j})\}_{j=1}^{N_i^{\mathrm{t}}}$ of dimensionality n with corresponding *binary* labels $Y_{i,j} \in \{-1, 1\}$, the MTL techniques we consider minimize the regularized logistic loss

$$\min_{\mathbf{W},\mathbf{c}} \sum_{i=1}^{N^{\mathrm{t}}} \sum_{j=1}^{N_i^{\mathrm{t}}} \log\left(1 + \exp\left(-Y_{i,j}\left(\langle \mathbf{W}_{i,\cdot}, X_{i,j}\rangle + \mathbf{c}_i\right)\right)\right) + \Omega(\mathbf{W}, \boldsymbol{\gamma}), \tag{5.7}$$

where $\mathbf{W}$ is the $N^{\mathrm{t}} \times n$ matrix of weight vectors for each of the tasks, $\mathbf{c}$ is the vector of scalar offsets for each task, and $\Omega(\mathbf{W}, \boldsymbol{\gamma})$ is an algorithm-specific regularization function that encodes the relatedness between the tasks, according to parameter vector $\boldsymbol{\gamma}$. The binary prediction $y \in \{-1, 1\}$ for sample vector $\mathbf{x}$ from task i is computed according to $y = \mathrm{sign}(\langle \mathbf{W}_{i,\cdot}, \mathbf{x}\rangle + \mathbf{c}_i)$.

We stress that the above formulation is designed for binary classification problems. At this time, however, we are not aware of existing techniques that can handle MTL problems consisting of more than two classes. Consequently, to compare our multiclass classification results to those produced using the MTL techniques, we decompose our multiclass domain adaptation problem into $\frac{K(K-1)}{2}$ binary subproblems, and learn a MTL model using one of the algorithms described below for each pair of classes. We predict the label for each unlabeled sample via majority vote over all of the binary models. We note that this scheme (i.e., one-vs.-one decomposition + majority-voting) is the same method used by the LIBSVM classifier for multiclass classification [Duan and Keerthi, 2005]. The training data for each of the binary subproblems consists of the labeled source data and the set of labeled target pivots for each pair of classes.

The MTL techniques we consider are Joint Feature Selection (JFS,Obozinski et al. [2009]), Multi-task Feature Learning (MTFL, Argyriou et al. [2007]) and Trace norm minimization (Trace, Ji and Ye [2009]). Each technique computes a low-dimensional feature representation that is shared among the tasks by applying different regularization functions to $\mathbf{W}$. The form of the regularization function $\Omega(\mathbf{W}, \boldsymbol{\gamma})$ for each algorithm are summarized in Table 5.9. The JFS technique enforces sparsity across the tasks by penalizing the L^1 norm of the matrix $\mathbf{W}$, and limits the complexity of each model by penalizing the L^2 (Frobenious) norm of $\mathbf{W}$. MTFL also penalizes the L^2 norm of $\mathbf{W}$, and also promotes similar sparsity patterns among the tasks by penalizing the sum of the L^2 norms of the tasks $\|\mathbf{W}\|_{1,2} = \sum_{j=1}^{n} \|\mathbf{W}_j\|_2$. Finally, the Trace method gives preference to low-rank models by penalizing the trace norm (i.e., the sum of singular values) $\|\mathbf{W}\|_*$. We use the implementation of each of the aforementioned algorithms provided in the Multi-tAsk Learning via StructurAl Regularization (MALSAR) toolbox [Zhou et al., 2011].

Method	$\Omega(\mathbf{W}, \boldsymbol{\gamma})$
JFS	$\gamma_1 \|\mathbf{W}\|_1 + \gamma_2 \|\mathbf{W}\|_2^2$
MTFL	$\gamma_1 \|\mathbf{W}\|_{1,2} + \gamma_2 \|\mathbf{W}\|_2^2$
Trace	$\gamma_1 \|\mathbf{W}\|_*$

Table 5.9 : Regularization functions $\Omega(\mathbf{W}, \boldsymbol{\gamma})$ for MTL Algorithms.

We use the labeled source data and the set of Q_k labeled target pivots from each class as training data for the binary multitask subproblems, and report the average test accuracy on the target task over five cross-validation folds. In each fold, we estimate the regularization parameters $\boldsymbol{\gamma}$ for each multitask model by splitting the training data for each binary task evenly into a train$_{CV}$ and a test$_{CV}$ set twice, and selecting the values of $\boldsymbol{\gamma}$ yielding the highest accuracy on the test$_{CV}$ set when trained on the train$_{CV}$ set. We found experimentally that selecting the $\boldsymbol{\gamma}$ values that maximize the average accuracy on both the source and target domains produced more accurate and stable results than the $\boldsymbol{\gamma}$ values that maximize accuracy on the target domain only, most likely due to the limited quantity of available labeled target samples to train the MTL algorithm.

Table 5.10 gives the classification accuracies using the multitask classifiers for the RAD2RFL and RFL2RAD scenarios. We observe comparable performance in the RAD2RFL scenario to the R-space classifiers shown in Table 5.8 using the MTFL and Trace algorithms, with MTFL slightly outperforming SVM-rbf (89.21% vs. 88.25%). Interestingly, we observe notably better accuracies for *small* values of Q_k in comparison to large Q_k using all three MTL algorithms, suggesting that a small set of representative labeled target samples is preferable to a larger set of potentially redundant labeled target samples, in some cases. However, we see significantly worse performance by all of the MTL algorithms in the more challenging RFL2RAD scenario in comparison to the R-space classifiers, with the most accurate MTL algorithm (MTFL, 80.15) yielding an average accuracy 4% worse than the least accurate R-space classifier (GRLVQ, 84.67%). Consequently, the overall accuracies in the two scenarios using the MTL algorithms are at least 2% worse than the R-space classifiers. We suspect that the reduced performance by the MTL algorithms is primarily due to the fact that each binary MTL task is learned independently of the others, and thus, the learned parameters only reflect the characteristics of each pair of classes across the tasks. With RelTrans, even if the problem is decomposed into a set of binary subproblems (as is the case with the SVM), the multiclass structure of the problem is reflected in the embedding in the R-space (we discuss this in greater detail later in Section 6.6). Consequently, the R-space classifiers can more accurately account for relationships between all of the classes across domains, instead of only pairwise relationships.

MTL		**RAD2RFL**		**RFL2RAD**		**Overall**	
	$\mathbf{Q_k}$	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
JFS	10	0.9248	0.0008	0.7980	0.0072	0.8614	0.0040
	25	0.8712	0.0099	0.7862	0.0127	0.8287	0.0113
	50	0.8681	0.0175	0.8143	0.0024	0.8412	0.0100
	75	0.7048	0.2214	0.7926	0.0155	0.7487	0.1185
	100	0.8530	0.0111	0.7854	0.0139	0.8192	0.0125
	Mean	0.8444	0.0521	0.7953	0.0103	0.8198	0.0312
MTFL	10	0.9273	0.0004	0.7943	0.0012	0.8608	0.0008
	25	0.8861	0.0032	0.8123	0.0139	0.8492	0.0086
	50	0.8852	0.0020	0.8143	0.0071	0.8498	0.0046
	75	0.8260	0.0436	0.7957	0.0048	0.8109	0.0242
	100	0.8678	0.0139	0.7910	0.0020	0.8294	0.0080
	Mean	*0.8785*	0.0126	*0.8015*	0.0058	*0.8400*	0.0092
Trace	10	0.9276	0.0000	0.8019	0.0119	0.8648	0.0060
	25	0.8861	0.0056	0.8002	0.0151	0.8432	0.0104
	50	0.8838	0.0008	0.8134	0.0083	0.8486	0.0046
	75	0.8833	0.0008	0.7943	0.0139	0.8388	0.0074
	100	0.8796	0.0004	0.7924	0.0048	0.8360	0.0026
	Mean	*0.8921*	0.0015	*0.8004*	0.0108	*0.8463*	0.0062

Table 5.10 : Mean and standard deviation of classification accuracy in the RAD2RFL and RFL2RAD scenarios using different MTL algorithms for $Q_k \in \{10, 25, 50, 75, 100\}$ labeled target samples / class. The mean accuracy over the range of Q_k values is provided for each algorithm. The best and second best performing algorithms for the average of the Q_k values are given in red and blue italics for each scenario, and overall. While the MTFL and Trace algorithms show competitive accuracies to RelTrans in the RAD2RFL scenario, they perform substantially worse in the RFL2RAD scenario.

5.9.3 Domain Adaptation, Learning Bounds and Model Selection

So far, we have shown that training/applying a multiclass classifier in the R-space often signficantly improves class knowledge transfer for a wide range of Q_k values. A question remains on how we can measure the quality of a set of pivots with respect to the domain adaptation task. Here, we describe an algorithm which leverages recent work by Ben-David et al. [2010a], who provide learning bounds on target domain error for binary domain adaptation problems.

The learning bounds provided by Ben-David et al. [2010a] each take the form

$$\epsilon_T(h) \leq \epsilon_S(h) + \mathrm{div}(\mathbf{D}^S, \mathbf{D}^T) + V \tag{5.8}$$

where $h(\mathbf{x}) \to \{-1, 1\}$ is a binary classifier trained using the source domain data, $\epsilon_D(h)$ is the error in the domain $D \in \{S, T\}$ using h, $\mathrm{div}(\mathbf{D}^S, \mathbf{D}^T)$ is a measure of divergence between the source and target distributions (described below), and V characterizes the complexity of the learning problem in each domain, along with the adaptability of the problem across the domains according to the true labeling functions $f^D(\mathbf{x})$ for domain $D \in \{S, T\}$. However, because the true labeling functions are unknown, we cannot estimate V in practical domain adaptation settings. Although it is possible to bound V based on the Vapnik-Chervonenkis (VC) dimension of the problem [Vapnik and Chervonenkis, 1971], determining the VC dimension is itself a nontrivial task, and the resulting bounds are typically too conservative to be of practical utility. Despite these issues, we show later that a classifier h which minimizes both the source domain error $\epsilon_S(h)$ and a measure of divergence between the domains $\mathrm{div}(\mathbf{D}^S, \mathbf{D}^T)$ often produces lower target domain error $\epsilon_T(h)$ than classifiers that do not minimize these criteria.

Measuring the difference between the source and target domains is a challenging problem, and a number of approaches have been proposed to acheive this goal (e.g., [Ben-David et al., 2010a; Gretton et al., 2009; Kifer et al., 2004]). The approach we take is inspired by the empirical H-divergence proposed by Ben-David et al. [2010a], which measures the difference between two distributions by learning a binary classifier to separate samples drawn from either. Here, we assume $\mathbf{D}^S$ and $\mathbf{D}^T$ are similar, and estimate the generalization performance of predictor f by measuring the divergence $\hat{d}_{h_j}$ between the set of source samples belonging to class j, X_j^S, and the set of target samples that f *predicts* belong to class j, $X_j^{T'}$, according to

$$\hat{d}_{h_j}(X_j^S, X_j^{T'}) = \min_{h_j} \left[\frac{1}{N_j^S} \sum_{\mathbf{x}:h_j(\mathbf{x})=1} \mathrm{I}\left(\mathbf{x} \in X_j^S\right) + \frac{1}{N_j^T} \sum_{\mathbf{x}:h_j(\mathbf{x})=-1} \mathrm{I}\left(\mathbf{x} \in X_j^{T'}\right) \right] \tag{5.9}$$

where $\mathrm{I}(\cdot)$ is the indicator function. $\hat{d}_{h_j}$ scores near 0.5 indicate we cannot distinguish samples drawn from either domain. Thus, we seek the classifier f that minimizes the average $\hat{d}_{h_j}$ over all K classes

$$\bar{d}_h = \frac{1}{K} \sum_{j=1}^{K} \hat{d}_{h_j}(X_j^S, X_j^{T'}) \tag{5.10}$$

Intuitively, if the classifier f generates accurate predictions, the set of samples $X_j^{T'}$ will contain many of the true target samples representing the j^{th} class. Moreover, because we assume the source and target domains are similar, it should be difficult to distinguish samples in X_j^S from samples $X_j^{T'}$. Quantitatively, we measure the difference X_j^S and $X_j^{T'}$ by training a binary classifier $h_j(\mathbf{x}) \rightarrow \{-1, 1\}$ that outputs the label 1 if sample $\mathbf{x} \in \left\{X_j^S, X_j^{T'}\right\}$ is a member of the source domain, and outputs the label -1 if $\mathbf{x}$ is a member of the target domain.

Algorithm 5.3 summarizes our algorithm, Prediction Divergence (PredDiv), which estimates $\bar{d}_h$ according to the predictions generated by classifier $f(\mathbf{x}) \rightarrow \{1, \ldots, K\}$. The algorithm proceeds by collecting the labeled source samples for each class j, X_j^S, and the set of samples that classifier f predicts belong to class j, $X_j^{T'}$ (Step 2). When f predicts no target samples belong to class j, then we assume we can easily distinguish between source and target samples from class j, and thus $\hat{d}_{h_j} = 1$. Otherwise, we calculate $\hat{d}_{h_j}$ by training a binary classifier h_j to separate 50% of the (randomly-selected) samples from (X_j^S, X_j^T), and applying h_j on the remaining 50%, averaging over L random splits (Step 8). $\bar{d}_h$ is calculated from the average sum of the $\hat{d}_{h_j}$ values. Below, $\mathbf{1}^N$ is an N-dimensional vector of ones, and $|X|$ gives the cardinality of set X.

Algorithm 5.3 PredDiv

```
Input: $N^S$ source samples $X^S$, source labels $y_i^S \in \{1, \ldots, K\}$, $N^T$ target samples
    $X^T$, multiclass predictor $f(\mathbf{x}) \rightarrow \{1, \ldots, K\}$, number of splits $L$
Output: Average per-class divergence score $\bar{d}_h$.
 1: for $j = 1$ to $K$ do
 2:     $X_j^S = \{\mathbf{x}^S\}_{\mathbf{x}^S : y_i^S = j}$              # Labeled source data for class $j$
 3:     $X_j^{T'} = \{\mathbf{x}^T\}_{\mathbf{x}^T : f(\mathbf{x}^T) = j}$          # Target predictions for class $j$
 4:     if $|X_j^{T'}| = 0$ then
 5:         $\hat{d}_{h_j} = 1$                          # No target predictions with label $j$
 6:     else
 7:         $X_j = \{X_j^S, X_j^T\}$, $Y_j \leftarrow \left[-\mathbf{1}^{|X_j^S|}, \mathbf{1}^{|X_j^{T'}|}\right]$
 8:         $\hat{d}_{h_j} = \frac{1}{L}\sum_{\ell=1}^{L} \mathrm{TwoFoldCV}(X_j, Y_j)$
 9:     end if
10: end for
11: return $\bar{d}_h = \frac{1}{K}\sum_{j=1}^{K} \hat{d}_{h_j}$
```

In the experiments below, we apply Algorithm 5.3 to measure the quality of the R-space models parameterized by the number of pivots/class $Q_k \in \{10, 15, 20, 25, 30, 35, 40\}$ in the RAD2RFL and RFL2RAD scenarios. We use the linear SVM classifier described in Section 5.9.1 as our multiclass predictor f, and train a separate binary linear SVM for each h_j. We select the SVM slack parameter C for f via cross-validation as described in Section 5.9.1. For each h_j, we fix C to one of $\{10^{-50}$, 10^{-25}, 10^{-10}, 10^{-5}, $10^{-2}\ \}$. As we show later, fixing C for each h_j classifier is necessary in order to compare the $\bar{d}_h$ values for different models. We set the number of splits L in the PredDiv algorithm to 5.

Figure 5.12 gives the R-S and R-ST accuracies for the RAD2RFL and RFL2RAD scenarios for the R-space models with $Q_k \in \{10, 15, 20, 25, 30, 35, 40\}$ pivots/class. Also provided are the corresponding $\bar{d}_h$ scores for each model, using the C values listed above. Several trends are apparent. First, the $\bar{d}_h$ scores follow similar trends for the values of C, although their ranges are dependant on the value of C. Consequently, we must select a fixed value of C for each h_j classifier in order to compare $\bar{d}_h$ scores of different models. We also observe that, for each value of C, the models which maximize the R-S and R-ST accuraces differ in both scenarios. In fact, the most accurate R-S model in the RAD2RFL scenario yields the 2nd lowest R-ST accuracies (0.8647). The models that minimize $\bar{d}_h$ yield comparable or higher R-ST accuracies in comparison to those produced by maximizing the R-S accuracies. However, while we observe an inverse relationship between the R-ST accuracy and $\bar{d}_h$ in the RAD2RFL scenario, a similar trend does not occur in the RFL2RAD scenario. The low R-S accuracies and relatively high $\bar{d}_h$ scores for the most accurate R-ST models $Q_k \in \{20, 25\}$ in the RFL2RAD scenario indicates that an inaccurate source model may increase the difference between the domains, but can potentially be more accurate for domain adaptation than models that are measureably more similar. However, we note that all of the models in the RFL2RAD scenario produce nearly equivalent R-ST accuracies, with the most accurate and least accurate models differring by only $\approx 1.5\%$. Additionally, the low $\bar{d}_h$ scores for the models with the lowest dimensionality ($Q_k \in \{10, 15\}$) suggest that $\bar{d}_h$ may show a preference for low-dimensional models when provided several models that yield similar target predictions.

5.9.4 On the Analysis of Synthetic Hyperspectral Data

An issue with synthetic hyperspectral data is that it often does not adequately capture all of the complex phenomena that occur in real imagery. For instance, in the studies we just described, the intra-domain accuracies for the source data and the target data are fairly high. While it is certainly possible to achieve such high accuracy when

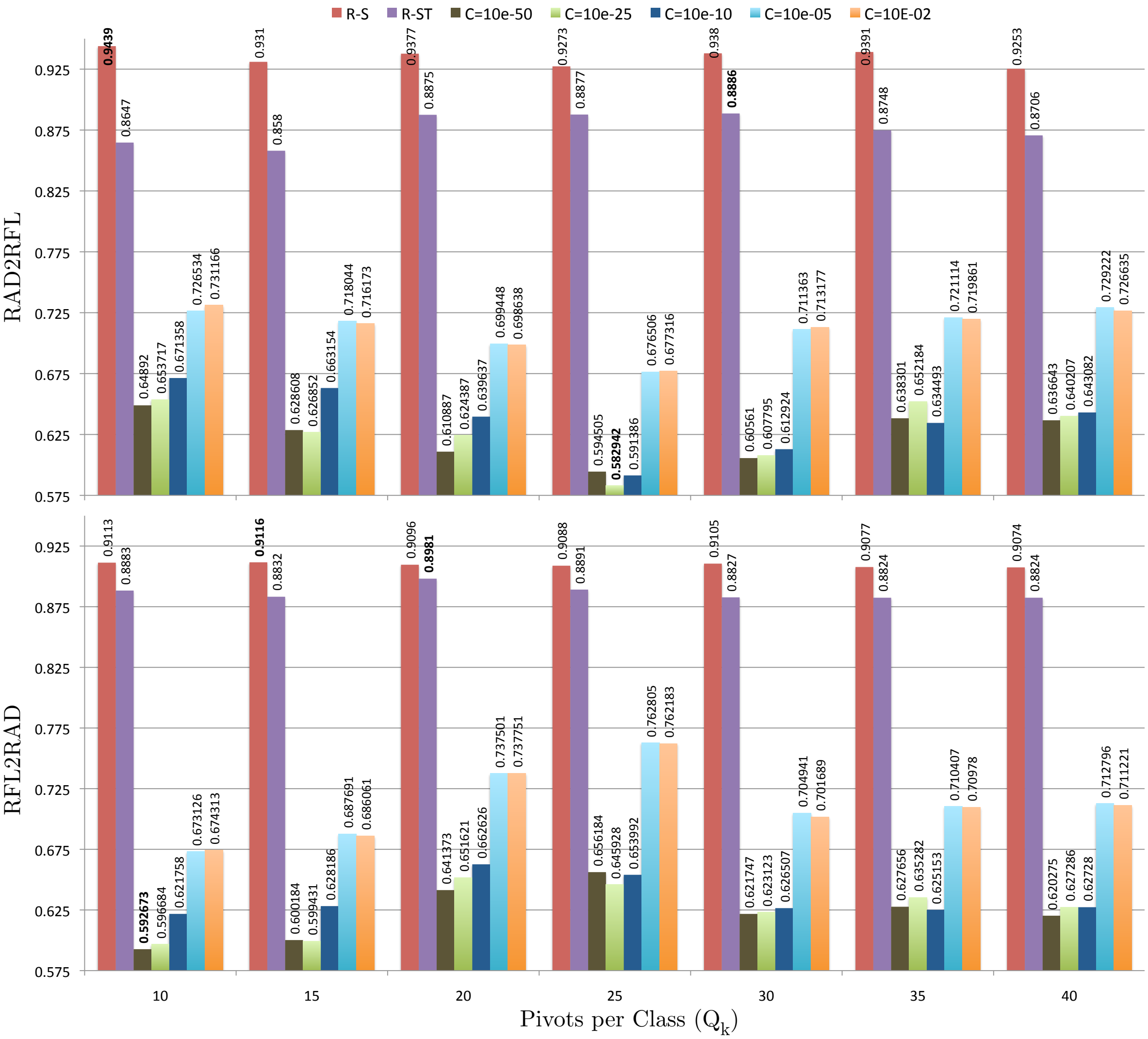


Figure 5.12 : RAD2RFL (top) and RFL2RAD (bottom) R-S and R-ST classification accuracies for R-space models with $Q_k \in \{10, 15, 20, 25, 30, 35, 40\}$ pivots/class, along with corresponding $\bar{d}_h$ scores using $C \in \{10^{-50}, 10^{-25}, 10^{-10}, 10^{-5}, 10^{-2}\}$. The numerical value of each vertical bar is given in rotated text. The most accurate R-S and R-ST results, and the minimum $\bar{d}_h$ scores are indicated in bold font. The Q_k at minimum $\bar{d}_h$ values produce higher R-ST accuracies than those with the highest R-S accuracies.

classifying real data, real image data is often less pristine, and classification accuracies using such data may be optimistic. However, recent work suggests [Mendenhall and Merényi, 2009] that the DIRSIG model is viable for the development of complex exploitation algorithms, based upon a comparative analysis between DIRSIG generated imagery and two previous analyses on real hyperspectral (AVIRIS) images. Yet, further validation on real hyperspectral imagery is crucial to ensure the robustness of the techniques we proposed. In the next chapter, we describe one such study using real hyperspectral imagery, and demonstrate that our methods yield good performance in both supervised and unsupervised domain adaptation problems.

Chapter 6

Unsupervised Domain Adaptation

Portions of this chapter are based upon the following publications:

- BD Bue and DR Thompson. "Multiclass Continuous Correspondence Learning". *NIPS Domain Adaptation Workshop* [Dec. 2011].
- BD Bue and C Jermaine. "Multiclass Domain Adaptation with Iterative Manifold Alignment". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS) (to appear)* [2013].

6.1 Class Knowledge Transfer without Labeled Target Data

In this chapter* we extend the RelTrans framework to the unsupervised domain adaptation setting by providing a methodology to automatically select pivot samples that represent similar classes in the source and target domains. We evaluate our technique on a multisensor, multitemporal class knowledge transfer task using real hyperspectral imagery in comparison to several baseline approaches and recently-proposed domain adaptation techniques. We show empirically that when between-class distances are preserved across domains, our automated pivot selection technique performs competitively to the supervised domain adaptation setting. We also discuss the theoretical ramifications of classifying samples in the R-space vs. the original source and target feature spaces. Based on these investigations, we provide extensions to RelTrans that (1) exploit functional characteristics of hyperspectral data to improve

*The work in this chapter was performed in collaboration with David Thompson and Chris Jermaine, with assistance from Erzsébet Merényi, Devika Subramanian and Kiri Wagstaff.

pivot selection, and (2) apply manifold alignment techniques to better reconcile differences between the source and target domains in terms of the spectral features most relevant to each class.

6.2 The Multiclass Continuous Correspondence Learning Algorithm

We now present the **M**ulticlass **C**ontinuous **C**orrespondence **L**earning (MCCL) algorithm for unsupervised domain adaptation (Algorithm 6.1). We consider the $K^S = K^T$ setting where source and target distributions share the same set of classes with labels $\{Y^S, Y^T\} \in \{1, \dots, K\}$. Given a discrete set of Q_k values Q_{range}, we begin by forming the source pivot set P^S from the Q_k samples nearest to each class mean for each class (Step 4). We then choose target pivots P^T for each class that best preserve the relative distance relationships between source pivots (Step 7). After collecting Q_k samples for each class, we evaluate the *Pivot Divergence* (Pdiv, Algorithm 6.2, described below) between the resulting pivot sets P^S and P^T (Step 9). Finally, we train a multiclass predictor using the transformed source samples (Step 13) to classify the transformed target samples (Step 14).

The Pdiv algorithm uses a technique inspired by the H-divergence [Ben-David et al., 2010b] (described in Section 5.9.3), which measures the difference between two distributions by finding a classifier which separates samples drawn from either. Low H-divergence scores indicate we cannot distinguish between samples drawn from either domain. Thus, we seek the pivot set size Q_{best} yielding the smallest average per-class H-divergence H_{best}.[†]

[†] We note that the Pdiv algorithm is a predecessor of the PredDiv algorithm discussed in Section 5.9.3, which uses a similar model selection strategy. Further analysis is underway to evaluate

Algorithm 6.1 Multiclass Continuous Correspondence Learning (MCCL)

```
Input: source training data $(X^S, Y^S)$, target data $X^T$, set of $Q_k$ values $Q_{\mathrm{range}}$.
Output: predicted target labels $Y^T$
$H_{\mathrm{best}} = 0$, $Q_{\mathrm{best}} = \min(Qrange)$
for $Q_k$ in $Q_{\mathrm{range}}$ do
    for $j$ in $\{1, \ldots, K\}$ do
        Select $Q_k$ source pivots $P_j^S$ with class labels $y_i^P = j$.
    end for
    for $j$ in $\{1, \ldots, K\}$ do
        Build target pivot set $P_j^T$ from $X^T$ by selecting best matching target pivot,
        $\mathbf{p}_i^T = \mathbf{x}_\ell^T$, for each source pivot $\mathbf{p}_i^S \in P^S$ with class label $y_i^P = j$ according to
            $$\ell = \underset{j}{\operatorname{argmin}} \|\mathrm{R}(\mathbf{p}_i^S, P^S) - \mathrm{R}(\mathbf{x}_j^T, P^S)\|, \; j \in \{1, \ldots, N^T\} \qquad (6.1)$$
    end for
    $H = \mathrm{Pdiv}(P^S, P^T)$
    if $H < H_{\mathrm{best}}$ then $H_{\mathrm{best}} = H$, $Q_{\mathrm{best}} = Q_k$
end for
Translate source and target samples to common feature space using $Q_{\mathrm{best}}$ paired
source, target pivots: $R^S = \{\mathrm{R}(\mathbf{x}_i^S, P^S)\}_{i=1}^{N^S}$, $R^T = \mathrm{R}(\mathbf{x}_i^T, P^T)_{i=1}^{N^T}$.
Train a multiclass predictor $h : \mathrm{R}(\mathbf{x}, P) \to Y$ using $R^S$.
return Prediction vector $Y^T = h(\mathbf{r}_i^T)_{i=1}^{N^T}$, $\mathbf{r}_i^T \in R^T$.
```

Algorithm 6.2 Pivot Divergence (Pdiv)

```
Input: pivot sets $(P^S, P^T)$, each of length $Q = \sum_{k=1}^{K} Q_k$
Output: pivot divergence score $H$.
for $k = 1$ to $K$ do
    Define label vector $y = ((-1)_{i=1}^{Q_k}, (1)_{i=1}^{Q_k})$ for pivot samples belonging to class $k$.
    Train binary predictor $h : R(\mathbf{p}, P) \to \{-1, 1\}$.
    Calculate divergence between class $k$ source and target pivots
    $H_k = \frac{1}{2Q_k}\left(\sum_{i=1}^{Q_k} \mathcal{I}(h(\mathbf{p}_i^S, P^S) = y_i) + \sum_{i=Q_k+1}^{2Q_k} \mathcal{I}(h(\mathbf{p}_i^T, P^T) = y_i)\right)$
return $H = \frac{1}{K}\sum_{i=1}^{K} H_k$
```

the relative capabilities of each technique.

6.3 Evaluation Methodology

We evaluate the performance of the MCCL algorithm in comparison to several other contexts. First, we calculate the baseline *intra-domain* source (S) and target (T) classification accuracies. The maximum of these gives a rough upper bound on the best achievable domain adaptation accuracy. In the baseline class knowledge transfer context (ST), we train a classifier on the source data to classify the target data in the original source and target feature spaces, which gives a lower bound we expect to improve. Next, we calculate the domain adaptation accuracy in the *supervised* context, where we sample the Q_k pivots from *labeled* source and target data (R-S, R-T, and R-ST, respectively). Lastly, we calculate the accuracy in the *unsupervised* domain adaptation context, where we choose the target pivots using the MCCL algorithm (R*-ST). We classify samples using the multiclass (one-vs-one) Support Vector Machine implemented in the LIBSVM package [Chang and Lin, 2011] with the linear kernel, and report test accuracy averaged over five cross-validation folds. We select the SVM slack parameter C from $\{10^{-4}, \ldots, 10^4\}$ that yields the highest accuracy on the training set.

6.4 Synthetic Example: Transformed Gaussians

We first provide an illustrative example of our methodology on a synthetic data set, shown in Figure 6.1 (left two plots). Each class consists of 500 samples drawn from one of four 2D Gaussians, each with unit covariance. The mean of each target Gaussian (bottom plot) is a randomly perturbed version of its corresponding source mean (top plot). Diamond markers indicate the $Q_k = 50$ source/target pivots selected using MCCL. The source and target accuracies in the original feature space (S, T and

ST) vs. the R-space (R-S, R-T, R-ST), along with the accuracy using MCCL for unsupervised pivot selection (R-ST*) are shown at the top of the left plot. In the right plot we show the class means μ_i^D mapped to the R-space $R(\mu_i^D, P^D)$ using pivots P^D for $D \in \{S, T\}$.

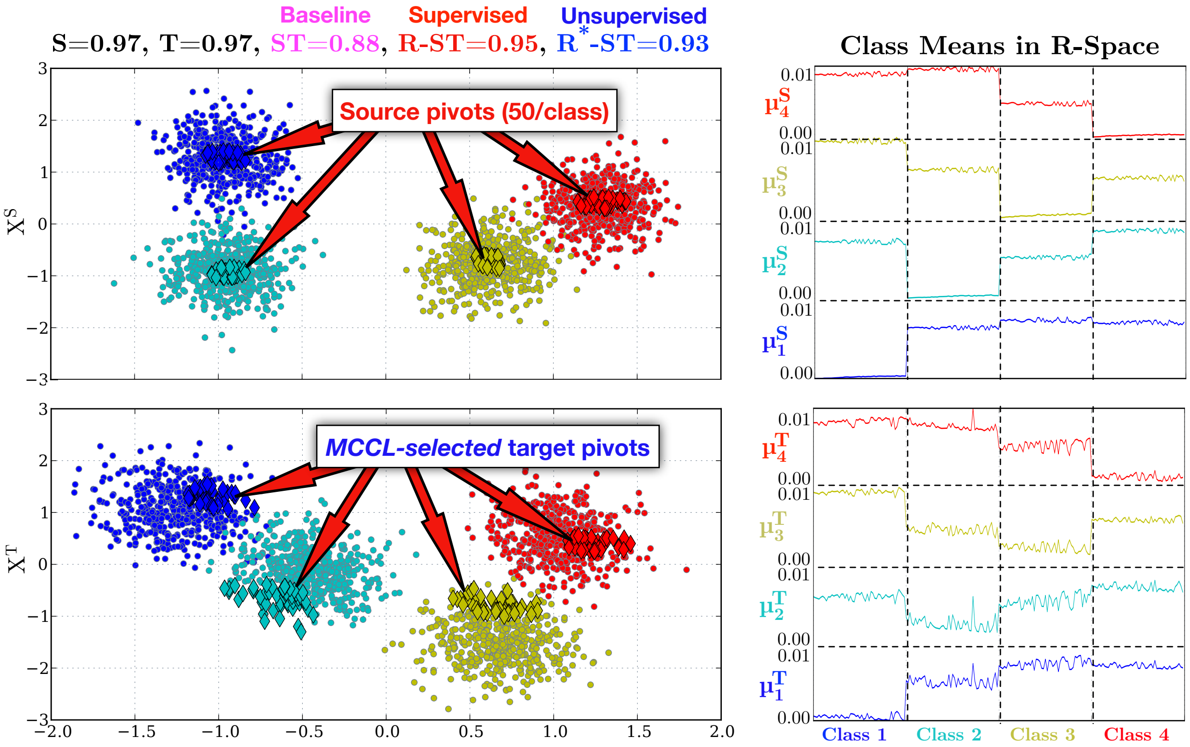


Figure 6.1 : Left: 4 class synthetic source (top) and target (bottom) data. Diamonds indicate the automatically-selected pivot samples selected by the MCCL algorithm. Right: source class means (top) and target class means (bottom) in the R-space R(μ_i^D, P^D) using source pivots P^S selected near the source class means, and target pivots P^T selected using MCCL.

Visually, the R-space class means appear better reconciled than in the original feature space, though not perfectly so due to the non-linear relationships between classes across the two domains, particularly between classes 2 (cyan) and 3 (yellow)). Despite this, we observe a notable improvement in accuracy over the baseline context (ST=0.88) after mapping the samples to the R-space in both the supervised (R-

ST=0.95) and unsupervised (R*-ST=0.93) context. We also observe that, although the pivots selected by MCCL in the target domain differ by a reasonable amount in position than the target class means, we still observe robust domain adaptation with the pivots, with the unsupervised (R*-ST) accuracy only 2% less than the supervised (R-ST) accuracy.

6.5 Case Study: Hyperspectral Imagery of Cuprite, NV

We now address the task of classifying a set of mineralogical spectra from one image using training data from another image captured under different conditions. This task represents a challenging multi-sensor, multi-temporal domain adaptation problem and is highly relevant to global hyperspectral mapping and analysis tasks. Our data consists of five mineralogical classes manually labeled by an expert geologist from two images of the Cuprite mining district in Cuprite, NV. Image *Av97* was captured in June 19, 1997 by the AVIRIS instrument, consists of 512×614 pixels, and was studied in detail in [Kruse et al., 2003]. Image *Hyp11* was acquired more recently on Feb. 06, 2011 by the Hyperion instrument onboard the EO-1 satellite, and contains 1798×779 pixels. Each pixel is a 29-dimensional vector of image radiance values measured at wavelengths in the range 2.1029-2.3249μm. We preprocess the images by applying the EML atmospheric calibration (i.e., conversion from spectral radiance to surface reflectance) procedure, and perform illumination normalization by scaling each pixel by its L^2 norm. False color composites of each image, along with training sample locations and class means are given in Figure 6.2.

We consider the following two domain adaptation scenarios. In the first scenario, we train a classifier using the Av97 image as the source data and test the classifier

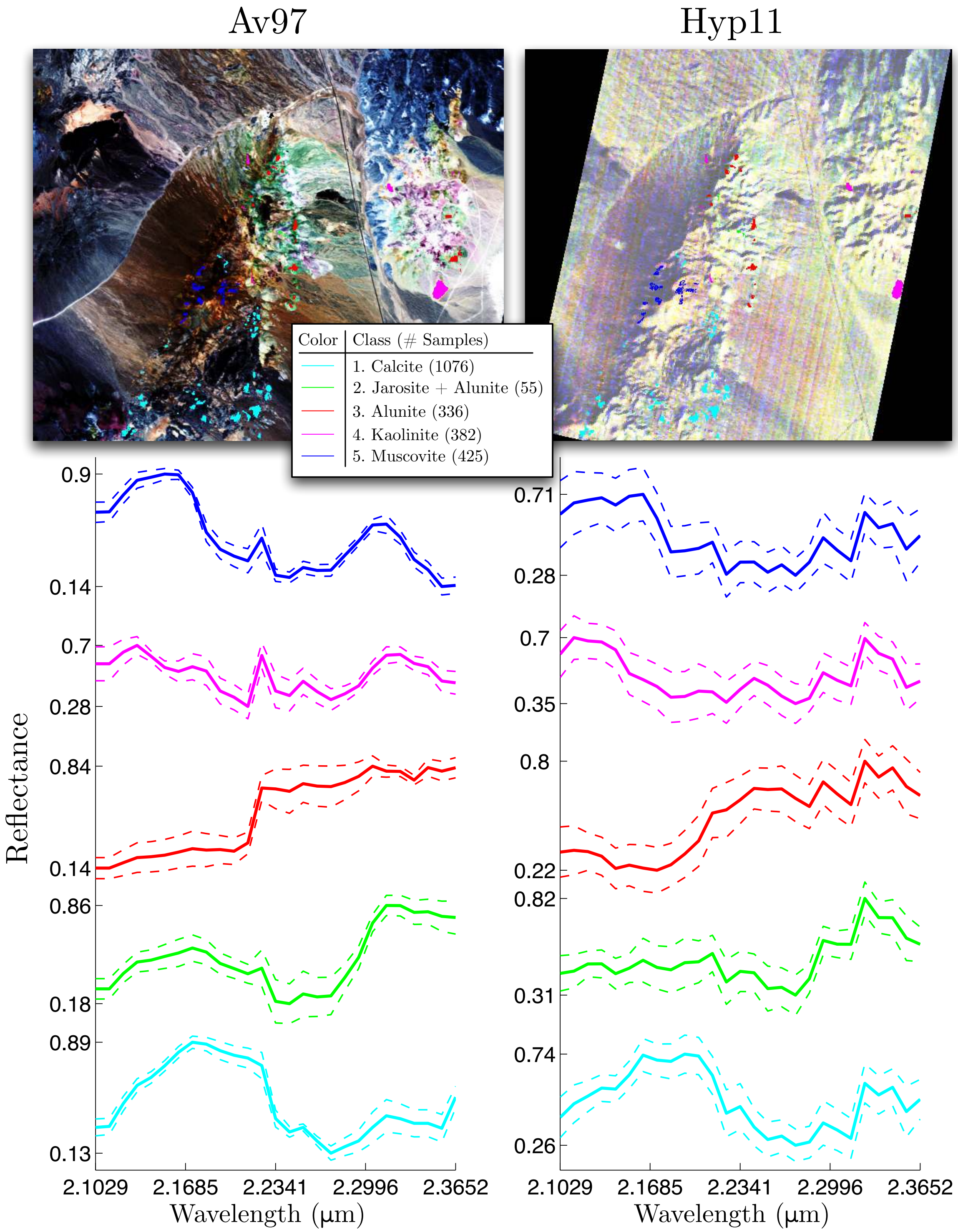


Figure 6.2 : Top: false color composites with sample locations for Av97 (left) and Hyp11 (right) images. The number of available training samples for each class are provided in parenthesis. Bottom: mean and standard deviation of each class.

on target data from the Hyp11 image. We refer to this scenario as *Av97⇒Hyp11*. In the second scenario we use the Hyp11 data as the source image, and the Av97 as the target image. We refer to this scenario as *Hyp11⇒Av97*. Because the smallest image contains over 300,000 pixels, we reduce the number of target pixels considered by the MCCL procedure by selecting the target pivots P^T from the means of the segments produced using the technique described in [Thompson et al., 2010].

6.5.1 Evaluation on Whitened Cuprite Spectra

As we can see from Figure 6.2, the means of identical classes appear differently in each image due to the differences in sensor type, environmental conditions, capture dates, and different atmospheric calibration techniques. In this section, we evaluate the domain adaptation performance after whitening each spectrum $\mathbf{x}^I$ for $\mathbf{I} \in$\{Av97, Hyp11\} as follows

$$\mathbf{x}^{\mathbf{I}}_{\text{white}} = (\mathbf{x}^{\mathbf{I}} - \boldsymbol{\mu}^{\mathbf{I}})(\mathbf{V}^{\mathbf{I}})(\mathbf{D}^{\mathbf{I}})^{-1/2}(\mathbf{V}^{\mathbf{I}})^T \qquad (6.2)$$

where $\boldsymbol{\mu}^{\mathbf{I}} = \mathrm{E}[\mathbf{I}]$ and $\mathrm{cov}(\mathbf{I}) = (\mathbf{V}^{\mathbf{I}})(\mathbf{D}^{\mathbf{I}})(\mathbf{V}^{\mathbf{I}})^T$ are the mean vector and global scatter matrix of *all* L^2-normalized samples in image **I**. The whitened class means are shown in Figure 6.3. We stress that, while the whitened spectral signatures are visually more similar than their unwhitened counterparts, and therefore potentially allow for improved class knowledge transfer over the unwhitened spectra, this may largely be a consequence of the fact that the Av97 and Hyp11 images both represent the same geographic region. Therefore, the images are quite similar in terms of their global covariance matrices, as the pixels from the same spatial locations represent identical materials. Even so, as we show in subsequent sections, we can often greatly improve class knowledge transfer between the two domains using MCCL without such preprocessing.

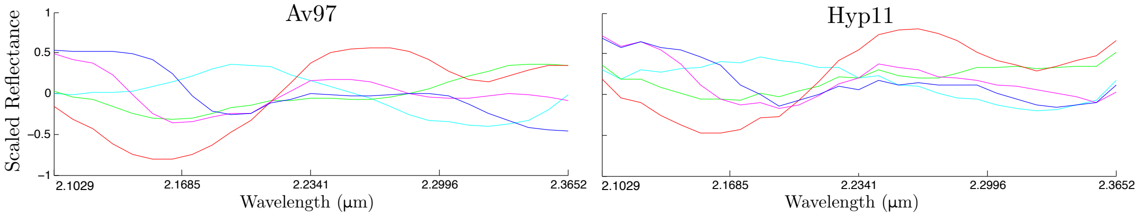


Figure 6.3 : Class means for Av97 (left) and Hyp11 (right) images after applying whitening filters.

Figure 6.4 gives classification accuracies (left two plots) and Pdiv scores (right two plots) on the whitened Av97 and Hyp11 data with respect to the number of pivots per class Q_k. Using RelTrans in the *supervised* domain adaptation context (R-ST), we select the top Q_k pivots for each class nearest to their corresponding class mean, as in Chapter 5. In the *unsupervised* context (R*-ST), we select the target pivots using Algorithm 6.1. We observe that the intra-image classification accuracies (S, T) are close to their corresponding R-space accuracies (R-S, R-T) when Q_k is sufficiently large ($Q_k \geq 10$), indicating that the R-space is as robust as the original feature space for intra-domain classification. We also observe that we achieve relatively high accuracy even for small Q_k in the supervised scenario (R-ST) when target labels are available. More importantly, both the R-ST and R*-ST results produce significantly higher accuracies than the baseline (ST) accuracies ($> 10\%$ in the Av97⇒Hyp11 scenario, and $\approx 3\%$ in the Hyp11⇒Av97 scenario). Additionally, the supervised and unsupervised results are comparable in both scenarios, differing by at most $\approx 2\%$. However, in the Av97⇒Hyp11 scenario, we observe lower domain adaptation accuracies than in the Hyp11⇒Av97 scenario, along with a larger gap between the R-ST and R*-ST results. Recall that the mapping between domains is defined by the source pivots, so if the classes are better separated in the target domain then in the source (e.g. the Hyp11⇒Av97 scenario), the mapping performs well. However, if the target classes are less separable than the source classes (e.g., the Av97⇒Hyp11 scenario), then the R-space induced by the source pivots may not discriminate the most ambiguous target classes, as the lower Av97⇒Hyp11 accuracies suggest.

We can see from the Pdiv scores in Figure 6.4 that the value of Q_k that minimizes the Pdiv also yields good classification performance. Specifically, we achieve the maximum R*-ST classification accuracy at the minimum Pdiv value in the Av97⇒Hyp11 scenario at $Q_k = 10$. Also, Pdiv increases with Q_k while the R*-ST accuracy remains relatively constant, indicating that additional pivots determined by the Av97 source data do not improve domain adaptation. In the Hyp11⇒Av97 scenario, while we see a gradual decrease in Pdiv for increasing Q_k – with slight improvements in accuracy, the Av97 classes are well separated for mid-range Q_k values $\in \{10, \ldots, 50\}$. For small Q_k, we observed low accuracy in all of R-S, R-T and R*-ST contexts, indicating the pivot set is inadequate to describe the classification task. We can filter such degenerate cases by ensuring that the R-space accuracy on the source data (R-S) is approximately the same as in the original feature space (S) (an approach also described in [Ben-David, 2006]). This potentially allows us to define a lower limit on the number of pivots necessary to define a feature space expressive enough for domain adaptation.

6.5.2 Comparison to Baseline and Related Techniques

In Section 6.5.1, we demonstrated improved domain adaptation performance by using RelTrans with automatically-selected pivot samples to reconcile differences between whitened source and target domain spectra representing identical classes from the same geographic region. However, as mentioned in Section 6.5, applying whitening filters can potentially reduce generalization performance for source and target data sets that differ in covariance structure (e.g., spectra from different regions). To

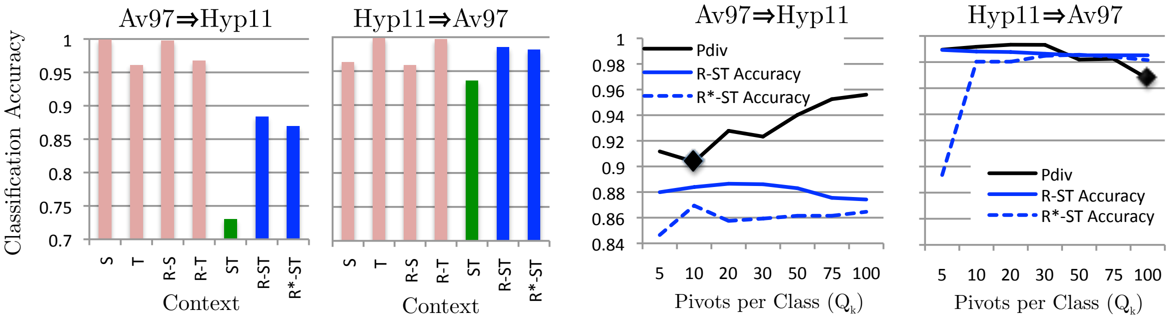


Figure 6.4 : Classification accuracies for Av97⇒Hyp11 and Hyp11⇒Av97 scenarios (left two plots) along with corresponding Pdiv scores vs. pivots/class Q_k (right two plots) for Av97⇒Hyp11 and Hyp11⇒Av97 scenarios. Black diamonds indicate the best Pdiv score for the R*-ST context yielding the classification accuracy in the left two plots.

further demonstrate the effectiveness of our methodology, here we compare our results to several baseline techniques applied to the *unwhitened* Av97 and Hyp11 spectra, to illustrate that we achieve similar accuracies without applying such preprocessing techniques. We provide the baseline (S, T, and ST) classification accuracies, along with the intra-domain accuracies in the R-space in both the supervised (R-T, R-ST) and unsupervised (R-T*, R-ST*) contexts, using the methodology described in Section 6.3. Additionally, we compute the target prediction accuracy using a classifier trained using only the Q target pivot samples (P^T, Y^P) selected using MCCL in the unsupervised (PivST*) context, and from the Q_k labeled samples nearest to each class mean in the supervised (PivST) contexts. We also compare these results to those produced using a classifier trained using the source data augmented with the target pivots $(\{X^S \cup P^T\}, \{Y^S \cup Y^P\})$ in the unsupervised (AugST*) and supervised (AugST) contexts, using the same pivots as in the PivST* and PivST contexts, respectively.

Table 6.1 shows the mean and standard deviation of classification accuracy using each the methods described above for $Q_k \in \{10, 25, 50, 75, 100\}$. We observe that the RelTrans results with unsupervised pivot selection (R*-ST) are nearly 8% better than the PivST* and AugST* results in the Av97⇒Hyp11 scenario, and produce comparable (within 0.5%) results in the Hyp11⇒Av97 scenario. In the supervised context, RelTrans (R-ST) performs comparably, but slightly worse (1-2%) than PivST and AugST in the Av97⇒Hyp11 scenario, whereas RelTrans outperforms PivST and AugST a similar (1-2%) margin in the Hyp11⇒Av97 scenario. The supervised results provide further insight into the difference in accuracies between the two scenarios observed in Section 6.5. Specifically, while we observe more significant gains in accuracy using RelTrans in comparison to the baseline in the Av97⇒Hyp11 scenario due to the source data being better separated than the target data, the PivT and AugT accuracies suggest that we may acheive higher accuracies by training/applying a classifier using the target pivots in the original feature space, instead of applying a classifier in the R-space. However, because the pivots in the R-ST context are selected by choosing the top Q_k samples nearest their respective class means in both of the source and target feature spaces, the respective R-space mappings are consequently skewed according to the inter-class distances in each feature space. Thus, the mapping to the R-space imposes a source-domain specific bias based upon these inter-class relationships. When the source and target feature spaces are already fairly similar (as with the Av97 and Hyp11 data), this bias slightly degrades prediction accuracy when the target classes are less separable than the source classes (as in the Av97⇒Hyp11 scenario), but does not degrade the classification accuracy in the other direction (as in the Hyp11⇒Av97 scenario). As we show later in Section 6.7, we can potentially improve these results by selecting source and target pivots that better preserve inter-class relationships across domains, rather than using the pivots nearest to their respective class means in the supervised setting.

We also compared our results to several related domain adaptation techniques. Each of the following techniques computes transformation functions $T^D(\mathbf{x}^D) : \mathbb{R}^n \rightarrow \mathbb{R}^m$,

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**
	10	0.9905	0.9223	0.9251	0.8062	0.8206	0.7466	0.8568	0.7466	0.8227
		0.0059	0.0141	0.0095	0.0126	0.0207	0.0271	0.0233	0.0040	0.0200
	25	0.9914	0.9231	0.9235	0.8206	0.8597	0.7408	0.8696	0.7462	0.8725
		0.0042	0.0083	0.0159	0.0074	0.0101	0.0173	0.0109	0.0140	0.0194
	50	0.9926	0.9157	0.9194	0.8285	0.8692	0.7347	0.8675	0.7557	0.8704
		0.0023	0.0089	0.0114	0.0155	0.0111	0.0229	0.0092	0.0168	0.0070
	75	0.9926	0.9169	0.9186	0.8350	0.8383	0.7577	0.8700	0.7400	0.8737
		0.0043	0.0137	0.0105	0.0086	0.0110	0.0088	0.0064	0.0109	0.0176
	100	0.9918	0.9165	0.9190	0.8388	0.8610	0.7594	0.8947	0.7462	0.8799
		0.0041	0.0143	0.0117	0.0113	0.0138	0.0147	0.0104	0.0198	0.0135
	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**
	10	0.9239	0.9926	0.9914	0.9741	0.9918	0.9515	0.9860	0.9424	0.9535
		0.0117	0.0037	0.0037	0.0082	0.0025	0.0231	0.0045	0.0155	0.0078
	25	0.9218	0.9930	0.9922	0.9576	0.9901	0.9729	0.9856	0.9379	0.9679
		0.0090	0.0047	0.0027	0.0054	0.0067	0.0086	0.0050	0.0090	0.0110
	50	0.9149	0.9934	0.9926	0.9864	0.9909	0.9749	0.9720	0.9646	0.9720
		0.0191	0.0037	0.0047	0.0034	0.0034	0.0106	0.0047	0.0067	0.0024
	75	0.9206	0.9922	0.9926	0.9552	0.9905	0.9424	0.9766	0.9757	0.9679
		0.0154	0.0031	0.0045	0.0098	0.0034	0.0149	0.0087	0.0072	0.0078
	100	0.9198	0.9914	0.9914	0.9605	0.9905	0.9909	0.9889	0.9848	0.9650
		0.0098	0.0061	0.0027	0.0099	0.0023	0.0031	0.0031	0.0072	0.0033
	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

Table 6.1 : Mean and standard deviation of classification accuracy in the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios using different baseline techniques using the $Q_k \in \{10, 25, 50, 75, 100\}$ pivot samples. The mean accuracy over the range of Q_k values is provided for each technique. In the unsupervised context, R-ST* matches or outperforms both PivST* and AugST* techniques. In the supervised context R-ST outperforms PivST/AugST in the Hyp11⇒Av97 scenario, but performs slightly worse than PivST/AugST in the Av97⇒Hyp11 scenario. This discrepancy is likely caused by selecting the source and target pivots near the means of each class, which slightly misaligns samples in the R-space.

for $m > 0$, $D \in \{S, T\}$, that reconcile the differences between the source and target feature spaces. However, each algorithm described below assumes a set of labeled target samples are available to guide the reconciliation process between the domains. To provide a balanced comparison to our results, we provide each algorithm with the set of target pivots selected by MCCL as the labeled target domain data.

Manifold Alignment with Procrustes Analysis:: Procrustes manifold alignment is a technique proposed by Wang and Mahadevan [2008] that computes a transformation that minimizes the Frobenious norm $\|P^S - P^T\|_F$ between the paired source pivot samples P^S and P^T. The resulting transformation can be subsequently applied to samples in the source domain to map them to a similar feature space as the target samples using the following function

$$T^S(\mathbf{x}^S) = s_f \mathbf{x}^S \mathbf{U}\mathbf{V} \tag{6.3}$$

where $\mathbf{UDV} = \text{SVD}(\text{COV}(P^S, P^T))$ is the singular value decomposition of the co-variance matrix between the paired source and target pivot matrices, and $s_f = \text{tr}(\mathbf{D})/\text{tr}(\text{COV}(P^S))$ is a source-domain dependant scaling factor.

Feature-level Manifold Alignment:: Wang and Mahadevan [2009] also proposed an alternative manifold alignment approach that computes the transformation function from the source feature space to the target domain feature space by framing the alignment problem as a graph embedding problem. Given the N^S labeled source samples $(\mathbf{X}^S, Y^S)$ and N^T labeled target domain samples X^T, their algorithm computes transformation matrices $\mathcal{F}^S$ and $\mathcal{F}^T$ by solving the generalized eigenvalue problem

$$\mathbf{Z}\mathbf{L}\mathbf{Z}^T\boldsymbol{\psi} = \boldsymbol{\rho}\mathbf{Z}\mathbf{D}\mathbf{Z}^T\boldsymbol{\psi} \tag{6.4}$$

where

$$\mathbf{Z} = \begin{pmatrix} \mathbf{X}^S & 0 \\ 0 & \mathbf{X}^T \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} \mathbf{D}^S & 0 \\ 0 & \mathbf{D}^T \end{pmatrix}, \quad \mathbf{L} = \begin{pmatrix} \mathbf{L}^S + \mu\Omega^S & -\mu\mathbf{W}^{S,T} \\ -\mu\mathbf{W}^{T,S} & \mathbf{L}^T + \mu\Omega_T \end{pmatrix} \tag{6.5}$$

where $\mathbf{L}^D = \mathbf{W}^D - \mathbf{D}^D$ is the graph Laplacian of the samples in $D \in S, T$ defined by adjacency matrix $\mathbf{W}^D$ and diagonal matrix $\mathbf{D}^D_{i,i} = \sum_j \mathbf{W}^k_{i,j}$; $\mathbf{W}^{S,T}$ is a $N^S \times N^T$ matrix with $\mathbf{W}^{S,T}_{i,j} = 1$ when $\mathbf{x}^S_i$ and $\mathbf{x}^T_j$ are in correspondence, 0 otherwise; $\mathbf{W}^{T,S}$ is the transpose of $\mathbf{W}^{S,T}$; and Ω^D is an $N^D \times N^D$ diagonal matrix with $\Omega^D_{i,i} = \sum_j \mathbf{W}^{S,T}_{i,j}$. By forming the transformation matrices $\mathcal{F}^S = \left(\boldsymbol{\psi}_{1,.}, \ldots, \boldsymbol{\psi}_{d,.}\right)$ and $\mathcal{F}^T = \left(\boldsymbol{\psi}_{N^S+1,.}, \ldots, \boldsymbol{\psi}_{N^S+d,.}\right)$, the algorithm ensures that the local geometries of each manifold are captured (as characterized by their respective graph Laplacians), while penalizing the differences between the manifolds (as characterized by the correspondence mapping) according to scalar weight parameter μ. We compute the adjacency matrices $\mathbf{W}^D$ using the k-Nearest-Neighbor (kNN) graph for each domain, where $\mathbf{W}^D_{i,j} = 1$ if $\mathbf{x}^D_j \in \text{kNN}(\mathbf{x}^D_i, k)$, 0 otherwise. We apply transformation $T^D(\mathbf{x}^D) = (\mathbf{x}^D)^T\mathcal{F}^D$ to reconcile the differences between the source and target samples, respectively, and classify the transformed samples using the linear SVM classifiver, as above. We select the k for the kNN graph from the set {1, 3, 5, 10, 15, 25, 50, 100} and the weight parameter $\mu \in \{10^{-2}, \ldots, 10^{2}\}$ that yield the highest accuracy on the training set.

EasyAdapt:: EasyAdapt is a kernel-based feature augmentation approach proposed by Daume [2007] that maps source and target examples from $\mathbb{R}^n$ to $\mathbb{R}^{3n}$ according to the transformation functions

$$\mathrm{T}^S(\mathbf{x}^S) = \left[\mathbf{x}^S, \mathbf{x}^S, \mathbf{0}^n\right] \tag{6.6}$$

$$\mathrm{T}^T(\mathbf{x}^T) = \left[\mathbf{x}^T, \mathbf{0}^n, \mathbf{x}^T\right] \tag{6.7}$$

where $\mathbf{0}^n$ is an n-dimensional zero vector. Under this mapping, the kernel product $\mathrm{k}(\cdot, \cdot)$ between samples in this new space becomes

$$\mathrm{k}(\mathrm{T}^S(\mathbf{x}^S), \mathrm{T}^S(\mathbf{x}^S)) = 2\mathrm{k}(\mathbf{x}^S, \mathbf{x}^S) \tag{6.8}$$

$$\mathrm{k}(\mathrm{T}^T(\mathbf{x}^T), \mathrm{T}^T(\mathbf{x}^T)) = 2\mathrm{k}(\mathbf{x}^T, \mathbf{x}^T) \tag{6.9}$$

$$\mathrm{k}(\mathrm{T}^S(\mathbf{x}^S), \mathrm{T}^T(\mathbf{x}^T)) = \mathrm{k}(\mathbf{x}^S, \mathbf{x}^T) \tag{6.10}$$

In other words, during both learning and prediction, the EasyAdapt tranformation maps samples to a feature space where samples in the same domain are given twice as much weight as samples in different domains. We apply the EasyAdapt transformations to each of the source and target samples, and train a classifier using the labeled source samples (X^S, Y^S), along with the MCCL-selected target pivot samples (P^T, Y^P) to predict labels for the unlabeled target samples X^T.

Table 6.2 compares the classification accuracies produced by RelTrans to the domain adaptation techniques described above on the (unwhitened) Av97 and Hyp11 data described in the previous section. We also provide results using the Trace norm regularization multitask learning technique described in Section 5.9.2. RelTrans produces the highest average accuracies in both of the Av97⇒Hyp11 (82.8%)and Hyp11⇒Av97 (96.9%) scenarios. Of the remaining algorithms, only EasyAdapt and

MTL-Trace produce overall accuracies comparable to the baseline (84.3%), although both perform 3-6% worse than RelTrans. Interestingly, the relatively simple EasyAdapt tranformation yields the second-best accuracies overall (86.5%), which indicates that its domain-specific weighting approach provides useful information a classifier can exploit in both training and prediction. However, like MTL-Trace, we believe that the EasyAdapt transformation is better suited for binary classifiation problems, as it does not encode information on the multiclass structure of the problem (as RelTrans does) that a classifier can leverage in training/prediction.

We also observe that both manifold alignment techniques generate poor prediction accuracies for small values of Q_k in both scenarios, and yield comparable overall accuracies, but only the feature-level alignment technique shows performance competitive to RelTrans for large Q_k, and requires a reasonable number of correspondences (i.e., $Q_k \geq 50$) to produce accuracies better than the baseline. This is likely a result of the fact that Procrustes alignment computes a single affine transformation between the source and target feature spaces, whereas the feature-level alignment technique is capable of computing non-affine transformations. However, both algorithms are also limited by the fact that they do not distinguish between correspondences in the same class vs. correspondences in different classes when computing their respective transformations between the domains. We will explore this relationship in more detail later in Section 6.8.

6.6 Model Selection and Unsupervised Domain Adaptation

Evaluating generalization performance of trained models on unseen test data is crucial for classification tasks. Such evaluation is particularly challenging in domain adaptation

		Av97⇒Hyp11		**Hyp11⇒Av97**		**Overall**	
	$\mathbf{Q_k}$	**Mean**	**Std**	**Mean**	**Std**	**Mean**	**Std**
Baseline	N/A	0.7429	0.0098	0.9428	0.0057	0.8429	0.0078
Procrustes Alignment	10	0.6985	0.0156	0.7639	0.0159	0.7312	0.0158
	25	0.7289	0.0174	0.8128	0.0117	0.7709	0.0146
	50	0.8054	0.0134	0.8437	0.0143	0.8246	0.0139
	75	0.7993	0.0089	0.8671	0.0195	0.8332	0.0142
	100	0.8104	0.0115	0.9025	0.0263	0.8565	0.0189
	Mean	*0.7685*	0.0134	0.8380	0.0175	0.8033	0.0155
Feature-level Alignment	10	0.4932	0.0136	0.6442	0.0344	0.5687	0.0240
	25	0.6623	0.0168	0.9576	0.0075	0.8100	0.0122
	50	0.7750	0.0143	0.9400	0.0245	0.8575	0.0194
	75	0.8268	0.0163	0.9704	0.0058	0.8986	0.0111
	100	0.7532	0.0152	0.9441	0.0242	0.8487	0.0197
	Mean	0.7021	0.0152	0.8913	0.0193	0.7967	0.0173
EasyAdapt	10	0.7520	0.0259	0.9317	0.0240	0.8419	0.0250
	25	0.7478	0.0258	0.9638	0.0206	0.8558	0.0232
	50	0.7808	0.0214	0.9696	0.0088	0.8752	0.0151
	75	0.7861	0.0185	0.9877	0.0087	0.8869	0.0136
	100	0.7450	0.0175	0.9885	0.0043	0.8668	0.0109
	Mean	0.7623	0.0218	*0.9683*	0.0133	*0.8653*	0.0176
MTL-Trace	10	0.7422	0.0076	0.9659	0.0076	0.8541	0.0076
	25	0.7537	0.0076	0.9042	0.0331	0.8290	0.0204
	50	0.7434	0.0151	0.9141	0.0145	0.8288	0.0148
	75	0.7484	0.0047	0.9009	0.0227	0.8247	0.0137
	100	0.7368	0.0267	0.8956	0.0081	0.8162	0.0174
	Mean	0.7449	0.0123	0.9161	0.0172	0.8305	0.0148
RelTrans	10	0.8062	0.0126	0.9741	0.0082	0.8902	0.0104
	25	0.8235	0.0099	0.9622	0.0096	0.8929	0.0098
	50	0.8318	0.0188	0.9663	0.0028	0.8991	0.0108
	75	0.8375	0.0048	0.9840	0.0047	0.9108	0.0048
	100	0.8396	0.0139	0.9568	0.0116	0.8982	0.0128
	Mean	*0.8277*	0.0120	*0.9687*	0.0074	*0.8982*	0.0097

Table 6.2 : Mean and standard deviation of classification accuracy in the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios using different domain adaptation algorithms using the $Q_k \in \{10, 25, 50, 75, 100\}$ paired pivot samples / class selected by MCCL as labeled target data. The mean accuracy over the range of Q_k values is provided for each algorithm. The best and second best performing algorithms for the average of the Q_k values are given in red and blue italics for each scenario, and overall.

settings, as the distributions of the training (source) and testing (target) data differ. In such circumstances, it is necessary to measure generalization performance on target data, but labeled target data is often scarce or unavailable. When labeled target data is limited or unavailable, model selection methods using widely-used techniques such as cross-validation may overfit to the source distribution, and consequently fail to accurately measure generalization performance to target data. For example, in traditional, intra-domain classification settings, we select the SVM model with slack parameter C via cross-validation on a hold-out set of labeled samples. In unsupervised domain-adaptation settings, only labeled source samples are available, and thus, the C value selected using the labeled source data may be suboptimal for the target samples. However, as we show below, by mapping the source and target domains to a common feature space using RelTrans, the model (e.g., the SVM parameterized by C) selected using only the labeled source data will typically generalize well to the target data.

Consider Figure 6.5. Here, we show the accuracies with respect to SVM slack parameter C in the original Av97 and Hyp11 feature spaces, using the unweightened data from the previous section (top), in comparison to the average accuracies for $Q_k \in \{10, 25, 50, 75, 100\}$ in the R-space (bottom). In both of the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios, we compute the "true" classification accuracy for each value of C using labeled target data. Our model selection objective in unsupervised domain adaptation is to select the C value using only the source data that maximizes the accuracy on the target data. In the original Av97 and Hyp11 feature spaces, we can see that the most accurate C values in the source domain do not correspond to the most accurate C value for domain adaptation. Specifically, in the Av97 source domain (cyan bars), $C = 1000$ is optimal, but $C = 0.01$ is optimal in the Av97⇒Hyp11 scenario. $C = 1000$ is also optimal in the Hyp11 source domain (red bars), but $C = 1$ is optimal in the Hyp11⇒Av97 (maroon bars) scenario. In the R-space, we observe

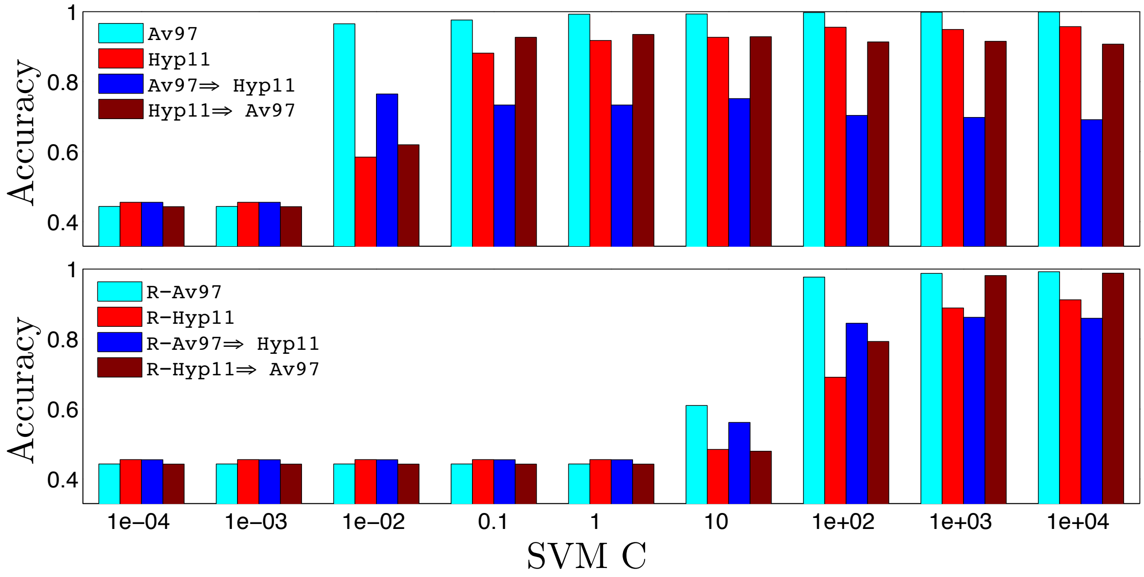


Figure 6.5 : Classification accuracy vs. SVM slack parameter C in the original Av97 and Hyp11 feature spaces (top) and in the R-space (bottom). Accuracies in the Av97⇒Hyp11 and Hyp11⇒Av97 and corresponding R-space scenarios computed based on the true target labels. R-space accuracies averaged over $Q_k \in \{10, 25, 50, 75, 100\}$. In the R-space, the C values maximizing the accuracy in the source domain typically maximize the domain adaptation accuracy as well, which is not the case in the original feature space.

that $C = 1000$ is optimal in both the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios, and also in the Av97 and Hyp11 source domains. Additionally, we observe that the accuracies in the source domains and in the domain adaptation scenarios follow similar trends in the R-space.

The results shown in Figure 6.5 suggest that we can perform more accurate model selection in the R-space than in the original feature space. However, we stress that while such techniques allow us to discriminate between acceptable vs. poor models, for fine-grained model selection tasks – for instance, selecting the best Q_k for a set of R-space models – the source domain model parameters will often not be optimal for the target domain. For instance, we can see from the accuracies given in Table 6.1 that selecting the value of Q_k that maximizes the source domain (R-S) accuracy does not necessarily yield the best performance in domain-adaptation (R-ST*, R-ST). Specifically, in the Av97⇒Hyp11 scenario, $Q_k \in \{50, 75\}$ give equal R-S accuracies, but $Q_k = 100$ yields the best R-ST* and R-ST performance. Similarly, in the Hyp11⇒Av97 scenario, $Q_k = 10$ is optimal in the R-S and R-ST contexts, but gives the 2nd highest accuracy (97.4%) in the R-ST* context. While, admittedly, the difference in accuracies between the best Q_k value selected from the R-S vs. R-ST*/R-ST contexts is often not particularly large (e.g., 83.5% vs. 83.9% in the R-ST* context and 83.8% vs. 86.1% in the R-ST context in the Av97⇒Hyp11 scenario), these observations suggest that good classification accuracy in the R-S context is a necessary, and not a sufficient, condition for optimal R-ST*/R-ST accuracy. In such cases, we can apply our PredDiv (Algorithm 5.3) or Pdiv (Algorithm 6.2) algorithms, or apply other recently-proposed model selection techniques for domain adaptation (e.g.,[Bruzzone and Marconcini, 2010; Gretton et al., 2009; Zhong et al., 2010]), to select a good model for the target domain.

6.7 Pivot Selection with Functional Measures

Until now, the distance measure we have used to map our source and target data to the R-space via Equation (5.1) has been the Euclidean distance. However, our results in Part II show that we can improve prediction accuracy using similarity measures that exploit characteristics specific to spectral data. Here, we focus on applying the Sobolev metric (Equation (3.12)) to the task of target pivot selection in the MCCL algorithm.

6.7.1 Methodology

We map a sample $\mathbf{x}^D$, $D \in S, T$, to a new feature space defined by distances between the spectral derivates of $\mathbf{x}^D$ and pivots $\mathbf{p}_i^D \in P^D$ with respect to their wavelengths (we hereafter refer to this feature space as the "R$^\kappa$-space") via the following transformation

$$\mathrm{R}^\kappa(\mathbf{x}^D, P^D) = \frac{1}{\kappa+1}\sum_{l=0}^{\kappa}\left(\frac{\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_1^D)}{\sum_{i=1}^{Q}\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_i^D)}, \ldots, \frac{\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_Q^D)}{\sum_{i=1}^{Q}\mathrm{d}^{(l)}(\mathbf{x}^D, \mathbf{p}_i^D)}\right), \qquad (6.11)$$

where $\mathrm{d}^{(l)}(\mathbf{x}_i, \mathbf{x}_j)$ is the Euclidean distance between the l^{th} derivatives of $\mathbf{x}_i$ and $\mathbf{x}_j$ (Equation (3.13)). When $\kappa = 0$, Equation (6.11) is equivalent to Equation (5.1). The i^{th} entry in the resulting Q-dimensional vector produced by the R$^\kappa$ function gives the likelihood of distinguishing sample $\mathbf{x}^D$ from pivot $\mathbf{p}_l^D$ with respect to the pivot set P^D, averaged over derivates $\{0, \ldots, \kappa\}$. We can now update the MCCL target pivot selection rule (Algorithm 6.1, Step 7) as follows

$$\ell = \underset{j}{\operatorname{argmin}} \; \|\mathrm{R}^\kappa(\mathbf{p}_i^S, P^S) - \mathrm{R}^\kappa(\mathbf{x}_j^T, P^S)\|, \; j \in \{1, \ldots, N^T\} \qquad (6.12)$$

to select target pivots $\mathbf{p}_i^T = \mathbf{x}_\ell^T$ that best preserve the *functional* relationships between the target pivots and source pivots $\mathbf{p}_i^S \in P^S$. For reasons that will be made clear later, we stress that we only use Equation (6.12) during pivot selection, and *not* for translating the source and target samples to the R-space during training (i.e., Algorithm 6.1, Step 12).

6.7.2 Evaluation on Cuprite Imagery

We evaluate the unsupervised domain adaptation performance using Equation (6.12) in the MCCL algorithm to select target pivots for the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios described above. We note that we do *not* apply whitening filters to the source and target spectra as described in Section 6.5, and thus we compute the mapping between domains in the unwhitened, L^2-normalized source and target feature spaces. The first three derivatives of the Av97 and Hyp11 class means are shown in Figure 6.6. We use the methodology described in Section 6.3 to compute the baseline intra-image (S) and domain adaptation (ST) classification accuracies for the first five derivatives of the Av97 and Hyp11 spectra, and provide those results in Table 6.3. As κ increases, the derivatives become more smooth and consequently, the intra-image accuracies tend to decrease in both images, though more rapidly for the noisier Hyp11 image classes than the better-separated Av97 image classes. We can also clearly see that, while the differences between the class means appear visually more similar to one another as κ increases, we also observe an inverse relationship between κ and the domain adaptation classification accuracies. However, we observe one critical exception to this trend between $\kappa = 0$ and $\kappa = 1$, where we observe a slight ($\approx 1.5\%$) increase in accuracy on the Hyp11 data and in the Av97⇒Hyp11 scenario, and a small decrease – relative to the larger κ – of $\approx 3\%$ in accuracy in the Hyp11⇒Av97 scenario. The intra-image accuracies indicate that the first derivative features are equally or more robust than the original $\kappa = 0$ features for classification, and the increase in ST accuracy in the Av97⇒Hyp11 scenario can be attributed to the improved separability of the first-derivative of the Hyp11 image classes, combined with the fact that the source and target feature spaces are better reconciled (as we can see visually in Figure 6.6), in the $\kappa = 1$ feature space. The reduction in accuracy in the Hyp11⇒Av97 scenario is a consequence of using a classifier trained on the well-separated Av97 spectra to classify the noisier Hyp11 spectra, as observed in previous sections.

Figure 6.7 gives our results using functional pivot selection in the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios. We denote the classification accuracies using functional

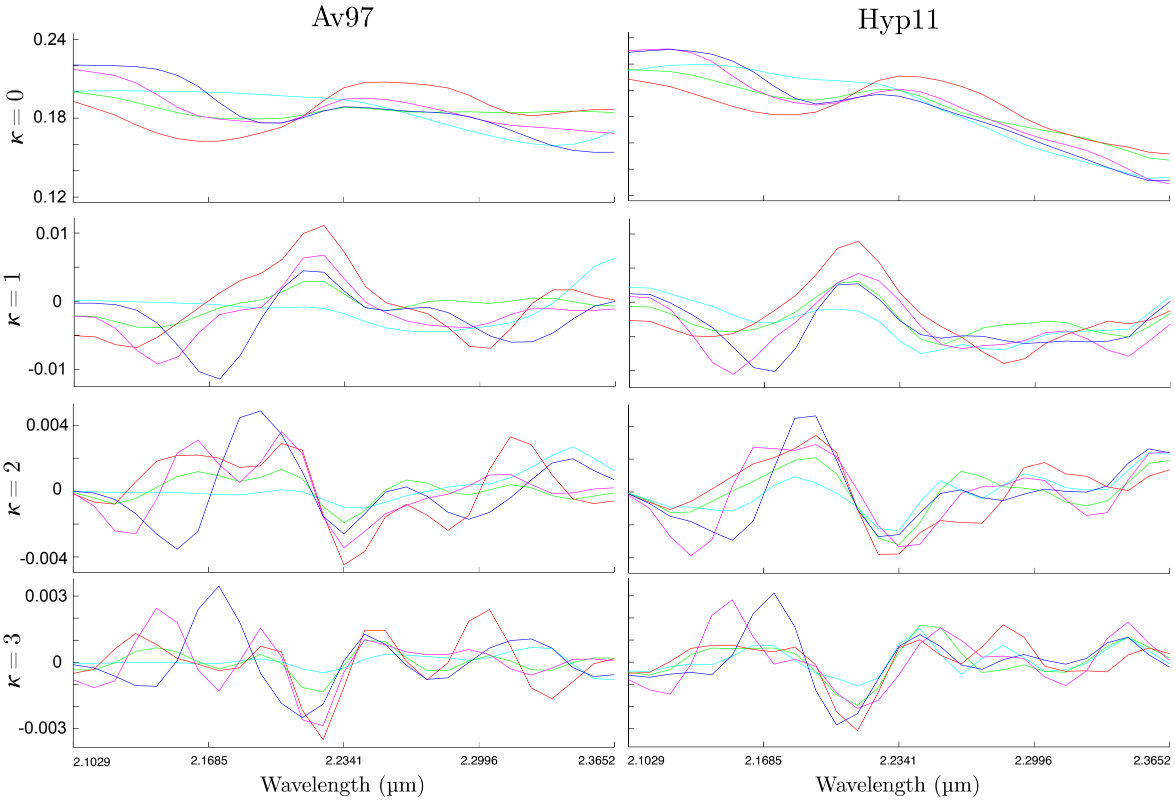


Figure 6.6 : Av97 and Hyp11 class means for derivates $\kappa \in \{0, \ldots, 3\}$.

	S		**ST**	
κ	**Av97**	**Hyp11**	**Av97⇒Hyp11**	**Hyp11⇒Av97**
0	1.0000	0.9626	0.7441	0.9362
1	1.0000	0.9724	0.7635	0.9087
2	0.9997	0.9412	0.5443	0.6610
3	0.9961	0.9120	0.4770	0.4426
4	0.9992	0.8663	0.4340	0.4426
5	0.9971	0.8499	0.4332	0.4426

Table 6.3 : Baseline (S, ST) classification accuracies for derivates $\kappa \in \{0, \ldots, 5\}$ for the Av97 and Hyp11 images. Accuracy typically decreases with κ, except in the Av97⇒Hyp11 scenario between $\kappa = 0$ and $\kappa = 1$, where the first derivative features are better reconciled than the original (i.e., $\kappa = 0$) features.

pivot selection in the unsupervised context as R$^{\kappa}$∗-ST, and use R-ST to denote the accuracies in the supervised context. We acheive the highest R$^{\kappa}$∗-ST accuracy using functional pivot selection in both the Av97⇒Hyp11 (0.8622 with $Q_k = 10, \kappa = 4$)

and Hyp11⇒Av97 (0.9868 with $Q_k = 50, \kappa = 3$) scenarios, acheiving comparable accuracies to the supervised (R-ST) context (0.8684 with $Q_k = 10$ and 0.9914 with $Q_k = 75$ in the Av97⇒Hyp11 and Hyp11⇒Av97 scenarios, respectively). Several other trends are also apparent. First, as the classification accuracy in the Hyp11⇒Av97 scenario is already reasonably high, we do not see as significant an improvement as in the Av97⇒Hyp11 scenario. However, the classification accuracy in both scenarios typically increases with κ, except in a few cases in the Hyp11⇒Av97 scenario where small κ produce high classification accuracies (i.e., $Q_k \in \{ 10, 75 \}$). We also see that the relative increase in accuracy between the R$^\kappa$∗-ST models

The results shown in Figure 6.7 are particularly interesting when we take into account the correlation between the intra-image d$^{(l)}$ distances in each domain (Table 6.4). As we discussed in Section 3.3.2, the accuracy of the adaptive Sobolev metric tends to decrease when the distances between the derivates become highly-correlated, because each derivate $f^{(j)}$ captures the same information as the preceeding $0 < k < j < \kappa$ derivatives. However, as Figure 6.7 shows, we see increased accuracy for increasing κ. This redundancy across the derivates improves accuracy in the pivot selection, allowing us to select a more representative pivot set than we select with our original pivot selection rule (i.e., Algorithm 6.1, Step 7).

	Av97						**Hyp11**					
d$^{(l)}$	**d0**	**d1**	**d2**	**d3**	**d4**	**d5**	**d0**	**d1**	**d2**	**d3**	**d4**	**d5**
d0	1.0000	0.8485	0.8338	0.7901	0.7820	0.7641	1.0000	0.7330	0.4652	0.3986	0.3802	0.3774
d1	0.8485	1.0000	0.9579	0.8698	0.8428	0.8150	0.7330	1.0000	0.8604	0.7507	0.7030	0.6729
d2	0.8338	0.9579	1.0000	0.9596	0.9442	0.9258	0.4652	0.8604	1.0000	0.9166	0.8597	0.8203
d3	0.7901	0.8698	0.9596	1.0000	0.9965	0.9895	0.3986	0.7507	0.9166	1.0000	0.9828	0.9646
d4	0.7820	0.8428	0.9442	0.9965	1.0000	0.9978	0.3802	0.7030	0.8597	0.9828	1.0000	0.9927
d5	0.7641	0.8150	0.9258	0.9895	0.9978	1.0000	0.3774	0.6729	0.8203	0.9646	0.9927	1.0000

Table 6.4 : Correlation coefficients for d$^{(l)}$ distances between labeled samples and their respective class means for derivates $l \in \{0, \ldots, 5\}$ in each of the Av97 and Hyp11 images.

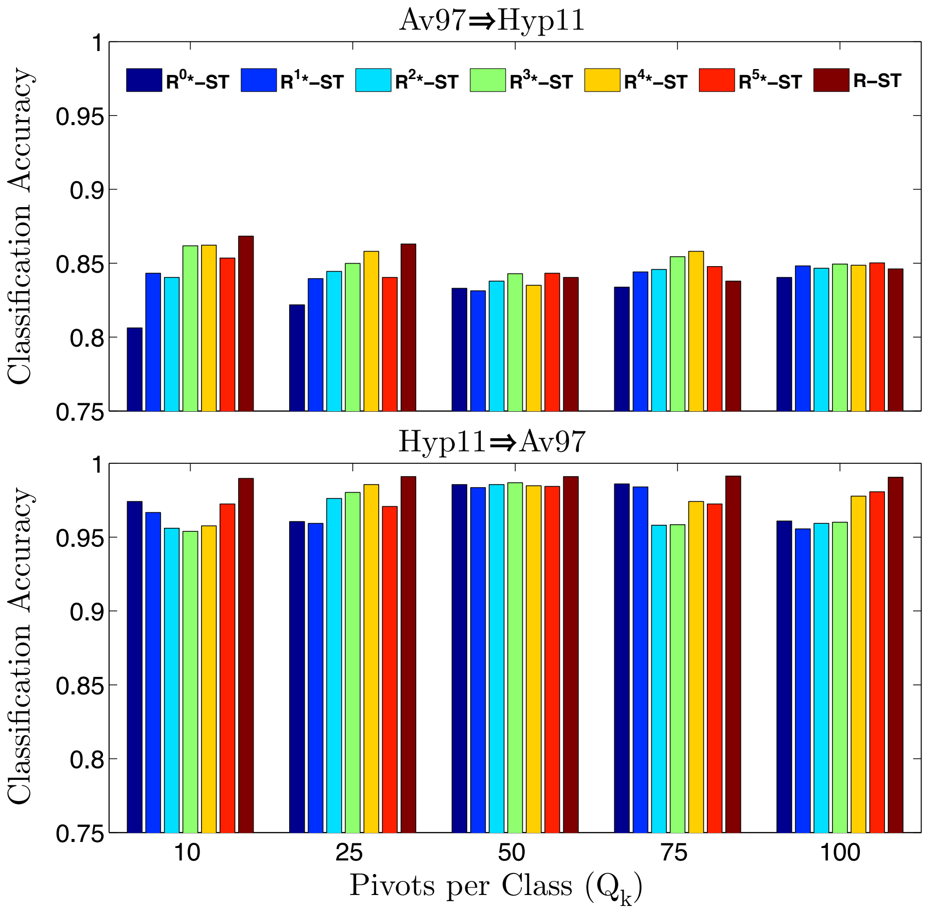


Figure 6.7 : Unsupervised domain adaptation accuracies using functional pivot selection ($R^{\kappa*}$-ST) with $\kappa \in \{0, \ldots, 5\}$ vs. the supervised domain adaptation accuracy (R-ST) in the Av97⇒Hyp11 (top) and Hyp11⇒Av97 (bottom) scenarios. Functional pivot selection typically improves classification accuracy when the baseline (R^0-ST) accuracy is low (e.g., in the Av97⇒Hyp11 scenario), and produces comparable accuracies when the baseline accuracy is already high (e.g., in the Hyp11⇒Av97 scenario).

Figure 6.8 demonstrates that classifying spectra in the R^{κ}-space typically produces suboptimal results. The difference between the results shown in Figure 6.7 vs. those shown in Figure 6.8 are the result of the fact that as κ increases, samples that represent identical classes in the source and target domains become closer to one another, but the inter-class distances also decrease as the derivatives become increasingly smooth. The consequence of this is that, in the R^{κ}-space, we can be more confident that pivots that match well in both domains represent the same classes, at the cost of reduced discrimination capabilities between spectra near the class decision boundaries. Thus, when we classify spectra in the R^{κ}-space, we observe a similar phenomenon as in Section 3.3.2, where accuracy decreases due to the combining redundant and increasingly ambiguous derivates. While the classification accuracy remains stable for small κ, accuracy rapidly decreases in a similar manner as seen in Table 6.3 for the larger κ values. The stability for $\kappa \in \{0, 1, 2\}$ is also explained by the relative robustness of each of their respective per-derivate feature spaces (99-100% accuracy in the Av97 image, and 94-96% accuracy in the Hyp11 image, as shown in Table 6.3), combined with the relatively low correlation between their per-derivate $d^{(l)}$ distances – ranging from 0.849-0.958 in the Av97 image, and from 0.465-0.860 in the Hyp11 image (Table 6.4), whereas the higher order derivates show correllation coefficients of over 0.959 and 0.916 in the Av97 and Hyp11 images, respectively. Consequently, we classify spectra in the R^0-space, and use the R^{κ} function only for pivot selection, i.e., we do not translate the source and target spectra to the R^{κ}-space ($\kappa > 0$) for classification.

6.8 Multiclass Manifold Alignment for Domain Adaptation

We now present an extension to the RelTrans multiclass domain adaptation procedure that computes band-weighted transformations from the source domain spectra to the target domain. Our algorithm, dubbed MARTIAL (MAnifold Reconciliation Through Iterative ALignment), incorporates an iterative manifold alignment approach inspired by the TRIAL protein structure alignment algorithm of Venkateswaran et al. [2011]. By learning rigid transformations between the source and target feature spaces based upon the pivot samples, MARTIAL can reconcile class-specific differences more

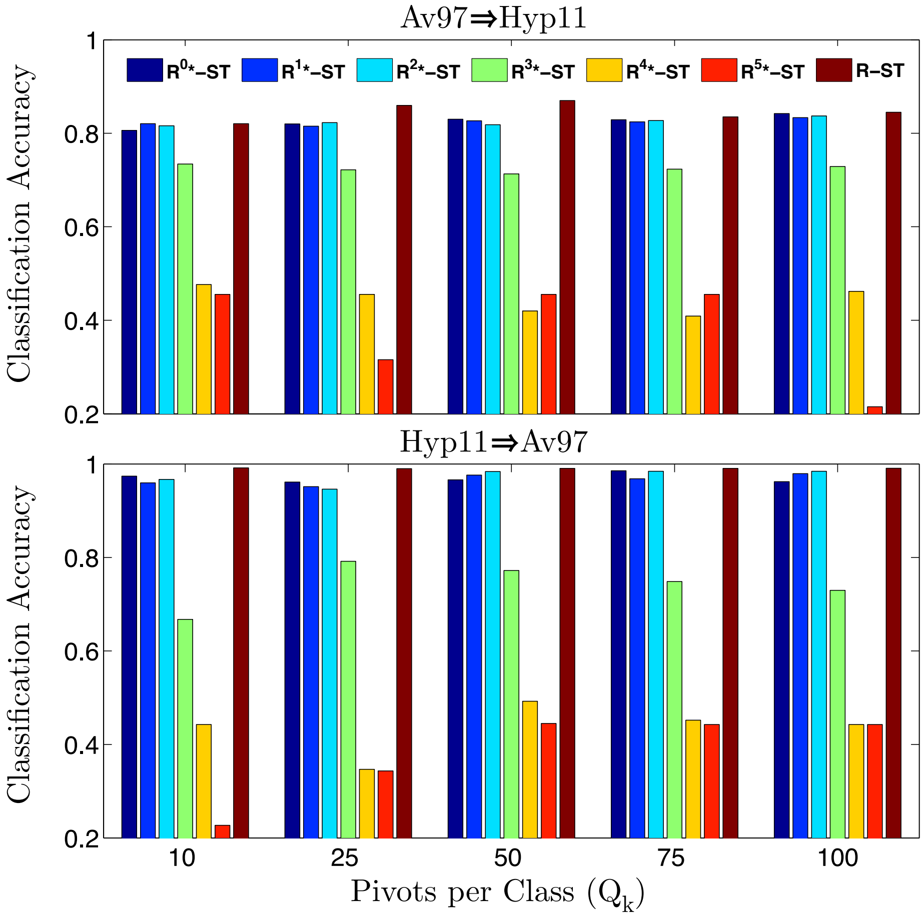


Figure 6.8 : Unsupervised domain-adaptation accuracies in the R^κ spaces ($R^{\kappa*}$-ST) with functional pivot selection for $\kappa \in \{0, \ldots, 5\}$ vs. the supervised domain adaptation accuracy (R-ST) in the Av97⇒Hyp11 (top) and Hyp11⇒Av97 (bottom) scenarios. Classification accuracy decreases with κ in both scenarios due to decreased inter-class distances in the R^κ-space. Thus, a better methodology is to select pivots in the R^κ-space, and classify in the R^0-space.

accurately than techniques that learn a single global transformation between domains. Additionally, because our technique is not tied to a specific classifier, we can apply any classifier in the transformed feature space to classify target data. We evaluate our results on real-world hyperspectral images of Cuprite, NV, and provide a MATLAB implementation[‡] online.

[‡] Available at: http://www.ece.rice.edu/~bdb1/#code

6.8.1 Methodology

As before, we assume we are given N^S source domain samples $(\mathbf{X}^S, Y^S) = \left\{(\mathbf{x}_i^S, y_i^S)\right\}_{i=1}^{N^S}$, $\mathbf{x}_i^S \in \mathbb{R}^n$, $y_i^S \in \{1, \ldots, K\}$, drawn from source distribution $\mathrm{p}^{\mathrm{S}}(\mathcal{X}, \mathcal{Y})$. We also assume $M^S >> N^S$ unlabeled samples are available in the source domain $\mathbf{X}^{Su} = \left\{\mathbf{x}_i^{Su}\right\}_{i=1}^{M^S}$. Our goal is to find a transformation $T : \mathbb{R}^n \to \mathbb{R}^n$ that maps samples drawn from $\mathrm{p}^S(\mathcal{X}, \mathcal{Y})$ to the feature space of N^T target samples $\mathbf{X}^T = \left\{(\mathbf{x}_i^T)\right\}_{i=1}^{N^T}$, $\mathbf{x}_i^T \in \mathbb{R}^n$, drawn from a similar distribution $\mathrm{p}^T(\mathcal{X}, \mathcal{Y})$. We can subsequently train a predictor $h : X \to Y$ to predict the class labels Y^T using the transformed source samples $\mathbf{X}^{ST}$ as training data. The MARTIAL algorithm uses several components of the TRIAL algorithm to learn a transformation for each source class to the target domain. Before we describe the MARTIAL algorithm, we provide a brief synopsis of the TRIAL algorithm below.

The TRIAL Algorithm : Given a pair of proteins $\mathbf{A} = \{\mathbf{a}_i\}_{i=1}^{N^A}$ and $\mathbf{B} = \{\mathbf{b}_j\}_{j=1}^{N^B}$, each consisting of C_α atoms $\mathbf{a}_i, \mathbf{b}_j \in \mathbb{R}^3$, The TRIAL algorithm aligns the manifolds defined by $\mathbf{A}$ and $\mathbf{B}$ such that their *alignment length* – the number of paired C_α atoms $(\mathbf{a}_i, \mathbf{T}\mathbf{b}_j)$ nearby one another after applying a (3×3) transformation matrix $\mathbf{T}$ to each $\mathbf{b}_j \in \mathbf{B}$ – is maximized. This allows TRIAL to identify structural commonalities between $\mathbf{A}$ and $\mathbf{B}$, which may be arbitrarily rotated with respect to one another. The algorithm consists of three main steps: (1) triplet (seed) alignment (denoted *Seed*) (2) initial alignment (*Align*), and (3) iterative improvement (*Improve*). In step (1), TRIAL searches for pairs of *triplet* (or *seed*) C_α atoms $\mathbf{P}^A = \left\{\mathbf{p}_1^A, \ldots, \mathbf{p}_3^A\right\} \subset \mathbf{A}$, $\mathbf{P}^B = \left\{\mathbf{p}_1^B, \ldots, \mathbf{p}_3^B\right\} \subset \mathbf{B}$ that are structurally similar to one another in terms of the Euclidean distances between their constituent atoms. TRIAL uses each of these pairs to find a preliminary, minimum root mean square deviation (RMSD) alignment between $\mathbf{A}$ and $\mathbf{B}$ using the Kabsch algorithm [Kabsch, 1978]. After removing any seed pairs producing degenerate (i.e., high RMSD) alignments, TRIAL reduces the distance between the two proteins while increasing the number of atoms in alignment (step (2)). It achieves this by iteratively recomputing $\mathbf{T}$ after adding any pairs $(\mathbf{a}_i, \mathbf{T}\mathbf{b}_j)$ whose Euclidean distances are less than a user-defined distance threshold ϵ to the $\mathbf{P}^A$, $\mathbf{P}^B$ sets, repeating the process until no more such pairs within the distance threshold can be added. During the final, iterative improvement step (3), TRIAL ensures that the maximum number of C_α atoms in $\mathbf{A}$ and $\mathbf{B}$ are aligned without increasing the RMSD of the aligned solution. Similarly to step (2), this involves iteratively adding any $(\mathbf{a}_i, \mathbf{b}_j)$ with distance less than an upper bound $\epsilon_{\max}$, based upon the current $(\mathbf{P}^A, \mathbf{P}^B, \mathbf{T})$ solution. After processing all of the candidate triplet pairs, TRIAL returns the $(\mathbf{P}^A, \mathbf{P}^B, \mathbf{T})$ solution maximizing the alignment length between $\mathbf{A}$ and $\mathbf{B}$.

Domain Adaptation with the MARTIAL Algorithm : The TRIAL algorithm has several attractive properties that lend themselves favorably to domain adaptation problems. Whereas several existing manifold alignment techniques assume a substantial quantity of (labeled) pairwise correspondences between domains are available at initialization (e.g. [Wang and Mahadevan, 2008; Yang and Crawford, 2011]), TRIAL is capable of adapting to the properties of the source and target manifolds with a relatively small number of labeled correspondences between domains ($\approx 10 - 100$ per-class) by iteratively incorporating informative *unlabeled* samples to refine the mapping between the domains. Additionally, the rigid transformations computed by TRIAL preserve functional relationships between adjacent spectral bands, which are crucial for accurate classification of hyperspectral signatures [Villmann et al., 2003].

However, several issues arise which prevent us from applying TRIAL directly in domain adaptation scenarios. Specifically, in domain adaptation, our objective is to minimize misclassifications, rather than maximizing the number of aligned samples between the source and target domains. Additionally, while we can assume that the C_α atoms in $\mathbf{A}$ and $\mathbf{B}$ each lie on single submanifold of $\mathbb{R}^3$, samples representing different classes in the source and target data can be viewed as lying on their own submanifolds of $\mathbb{R}^n$, and the submanifold of a particular class in the target domain may be arbitrarily transformed with respect to the submanifold of the same class in the source domain. Finally, in domain adaptation, we must consider problems involving hundreds to thousands of samples of high dimensionality, which involves significantly greater computational costs than those involved in protein alignment problems.

We account for the challenges involved in multiclass domain adaptation by making the following modifications the TRIAL algorithm: (1) we perform an initial filtering step where we select a pool of candidate pivot samples that are structurally similar to the source domain classes in both the source and target domains; (2) rather than learning a single global transformation between the domains, we learn a transformation for each source class using the pivot samples. This allows us to resolve domain-specific differences relative to each class, while also constraining the number of samples necessary to consider during alignment; and (3) we automatically compute the RMSD threshold ϵ for each class by randomly selecting a set of initial *seed* pairs of fixed size from the set of candidate pivots. While this is not guaranteed to produce an optimal RMSD transformation, we found that selecting the lowest RMSD transformation over 25-50 randomly selected seed pairs works well in practice to filter out degenerate solutions.

Algorithm 6.3 MARTIAL

```
Input: $N^S$ labeled source samples $(\mathbf{X}^S, Y^S)$, $M^S$ unlabeled source samples $\mathbf{X}^{Su}$, $N^T$ unlabeled target samples $\mathbf{X}^T$, number of candidate pivots $N_i^P$ per class, number of seed samples per class $Q_k$, number of random inits $N_{\text{rand}}$.
Output: Target-transformed source samples $\mathbf{X}^{ST}$
Use MCCL to select $N^P$ candidate pivots $\mathbf{P} = (\mathbf{P}^S, \mathbf{P}^T, Y^P)$, $\mathbf{P}^S \subset (\mathbf{X}^S \cup \mathbf{X}^{Su})$, $\mathbf{P}^T \subset \mathbf{X}^T$, consisting of $N_i^P$ samples per-class.
$\mathbf{X}^{ST} = \emptyset$
for $i = 1$ to $K$ do
    $\mathbf{X}_i^S = \left\{\mathbf{x}_j^S \in \mathbf{X}^S | y_j^S = i\right\}$
    $\mathbf{P}_i = \left\{(\mathbf{p}_j^S, \mathbf{p}_j^T, y_j^P) \in \mathbf{P} | y_j^P = i\right\}$
    $(\mathbf{P}_0, \mathbf{T}_0, \epsilon_0) = \text{RANDINIT}(\mathbf{P}_i, Q_k, N_{\text{rand}})$
    $\mathbf{T}_i^S = \text{TRIAL}(\mathbf{P}_i, \mathbf{P}_0, \mathbf{T}_0, \epsilon_0)$
    $\mathbf{X}^{ST} = \left\{\mathbf{X}^{ST} \cup \mathbf{T}_i^S \mathbf{X}_i^S\right\}$
end for
```

Algorithm 6.3 describes the MARTIAL algorithm, which maps a set of labeled source samples $(\mathbf{X}^S, Y^S)$ to the target domain feature space. The algorithm begins by using the Multiclass Continuous Correspondence Learning (MCCL) algorithm Algorithm 6.1 to select a pool $\mathbf{P} = (\mathbf{P}^S, \mathbf{P}^T, Y^P)$ of N^P candidate *pivot samples*, consisting of N_i^P paired samples representing each of the K source classes. We denote the set of N_i^P pivots representing the i^{th} class as $\mathbf{P}_i = \left\{(\mathbf{p}_j^S, \mathbf{p}_j^T, y_j^P)\right\}_{j=1}^{N_i^P}$, where $y_j^P = i$. The set of N^P source pivots $\mathbf{p}_j^S \in \mathbf{P}^S$ consist of the top N_i^P samples in $(\mathbf{X}^S \cup \mathbf{X}^{Su})$ nearest to the mean of each source class. For each source pivot $\mathbf{p}_j^S \in \mathbf{P}^S$, MCCL selects the target pivot $\mathbf{p}_j^T = \mathbf{x}_\ell^T \in \mathbf{X}^T$ that is most likely to belong to the same class as $\mathbf{p}_j^S$ according to

$$\ell = \underset{i}{\text{argmin}} \, \|\text{R}(\mathbf{p}_j^S, \mathbf{P}^S) - \text{R}(\mathbf{x}_i^T, \mathbf{P}^S)\|, \; i \in \{1, \ldots, N^T\}, \tag{6.13}$$

By selecting the candidate pivots in this "R-space," MCCL finds target samples that approximately preserve the relative distances between the source classes, as characterized by the source pivots. When the source and target feature spaces are similar, these target pivots typically represent the same classes as their corresponding source pivots.

After the candidate pivots are selected, the MARTIAL algorithm uses the pivots from the i^{th} class, $\mathbf{P}_i$, to compute the Seed alignment transformation $\mathbf{T}_0$ for samples in that class . This is achieved by sampling N_{rand} seed pairs from $\mathbf{P}_i$, each consisting of $Q_k < N_i^P$ samples of the form $\mathbf{P}_0 = (\mathbf{P}_0^S, \mathbf{P}_0^T) = \left\{(\mathbf{p}_j^S, \mathbf{p}_j^T)\right\}_{j=1}^{Q_k}$, applying the Kabsch algorithm to each seed pair, and returning the $(\mathbf{T}_0, \mathbf{P}_0)$ producing the smallest value of $\epsilon_0 = \text{RMSD}(\mathbf{T}_0\mathbf{P}_0^S, \mathbf{P}_0^T)$ (Step 6).

We then pass this filtered set of pivots to the TRIAL function for refinement (Step 7). The TRIAL function performs the initial alignment and iterative improvement steps of the TRIAL algorithm as described in ([Venkateswaran et al., 2011], Figures 2 and 4), returning the $n \times n$ transformation matrix $\mathbf{T}_i^S$ that maps samples from class i to the target feature space. We apply $\mathbf{T}_i^S$ to the source samples $\mathbf{X}_i^S$, and add them to the set of transformed source samples $\mathbf{X}^{ST}$ (Step 8), and can subsequently train a multiclass classifier using $(\mathbf{X}^{ST}, Y^S)$ to classify target samples $\mathbf{X}^T$.

6.8.2 Evaluation on Cuprite Imagery

We now evaluate the performance of the MARTIAL algorithm on the *Av97⇒Hyp11* and *Hyp11⇒Av97* scenarios described in Section 6.5. As in Section 6.7.2, we consider the unwhitened, L^2 normalized Av97 and Hyp11 spectra. In each scenario, we measure the source-to-target (ST) classification accuracy without domain adaptation, which provides a baseline accuracy we seek to improve. We then measure the classification accuracy using the transformations produced by the MARTIAL Seed (Algorithm 6.3, Step 6), Align and Improve (Imp., Algorithm 6.3, Step 7) steps. We select $N_i^P = 250$ candidate pivots from each class, and evaluate classification accuracy for seed sizes

$Q_k \in \{10, 12, 15, 20, 24, 30, 36, 40, 42{,}50, 75, 100\}$. We compare our results to those produced using the Procrustes alignment technique (abbreviated *Proc.*) described in Section 6.5.2. In fact, the MARTIAL seed alignment step can be interpreted as applying the Procrustes alignment algorithm to the pivots representing each class. We also provide results after mapping the source and target spectra to the R-space using source samples in their original feature space (R_S) and the source samples produced after applying the MARTIAL Seed (R_{Seed}), Align (R_{Align}) and Improve ($R_{Imp.}$) steps. We use the same Q_k pivots from each class used in the MARTIAL Seed alignment step for the Procrustes and the R-space mappings. Our classifier is the multiclass linear Support Vector Machine (SVM) implemented in the LIBSVM package [Chang and Lin, 2011], evaluated using five-fold cross-validation. We select the SVM slack parameter $C \in \{10^{-3}, \ldots, 10^{3}\}$ that yields the highest accuracy on the training data.

Figure 6.9 shows the classification accuracy vs. the number of seed samples Q_k for each algorithm in the Av97⇒Hyp11 (left) and Hyp11⇒Av97 (right) scenarios. In the Av97⇒Hyp11 scenario, we observe that classifying source samples after each of the MARTIAL Seed, Align and Improve steps produces accuracies significantly better than the baseline (8-11%). The poor performance by the Procrustes alignment algorithm for most Q_k values implies that the single global transformation computed using the pivot samples does not adequately resolve the class-specific differences between the images. We also observe dramatic improvements over the Procrustes alignment using MARTIAL in the Hyp11⇒Av97 scenario. However, as noted in [Bue and Thompson, 2011], because the classes are better separated in the Av97 image than in the Hyp11 image, we achieve high classification accuracy ($\approx 94\%$) in the Hyp11⇒Av97 scenario with the baseline classifier. The remaining classes are challenging to separate, as indicated by the roughly comparable performance to the baseline using each of the domain adaptation algorithms. On average, however (as shown in Table 6.5 below), classifying source samples transformed by MARTIAL yields slightly better accuracies than the baseline.

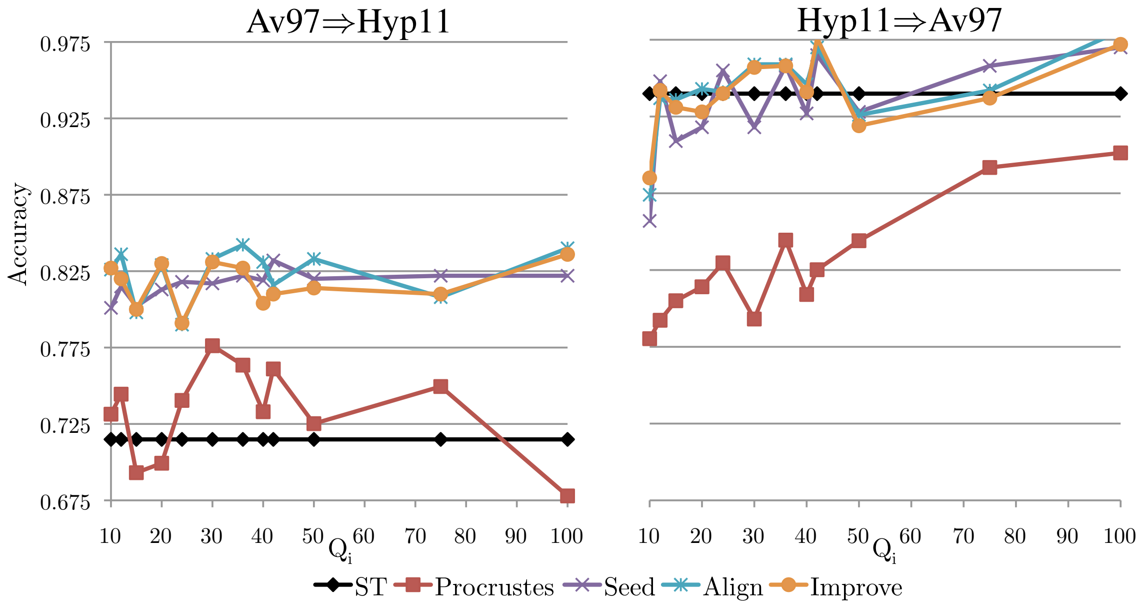


Figure 6.9 : Classification accuracy vs. number of seed samples Q_k for the Av97⇒Hyp11 (left) and Hyp11⇒Av97 (right) scenarios with the baseline (ST, black $\diamond$), Procrustes alignment (red $\Box$), and MARTIAL Seed (purple $\times$), Align (turquoise $*$), and Improve (orange $\circ$). The feature spaces produced using MARTIAL are better reconciled than the original (ST) and Procrustes-aligned feature spaces, as evidenced by the increase in classification accuracy.

We observe more substantial improvements in classification accuracy when we classify our data in the R-space (Equation (5.1)) after applying MARTIAL. These results are shown in Figure 6.10. In the Av97⇒Hyp11 scenario, classifying the target samples in the R-space using the source data transformed by MARTIAL produces uniformly better results for all Q_k than in the R-space with the original source features (R_S), indicating that the domains are better reconciled after applying the MARTIAL transformations. The R-space classification results using MARTIAL are also better than those given in Figure 6.9 for all $Q_k \neq 100$. Not surprisingly, as the classification accuracies in the Hyp11⇒Av97 scenario are already high, the R_S and the MARTIAL

$\mathrm{R}_{\mathrm{Align}}$ and $\mathrm{R}_{\mathrm{Imp.}}$ cases produce comparable, but not significantly better accuracies (±1%). We also observe that the most-accurate MARTIAL results shown in Figure 6.10 approach the *supervised* domain adaptation (R-ST) results reported in Table 6.1, with MARTIAL yielding 85.10% vs. 86.10% R-ST accuracy in the Av97⇒Hyp11 scenario, and 97.61% vs. 99.18% R-ST in the Hyp11⇒Av97 scenario.

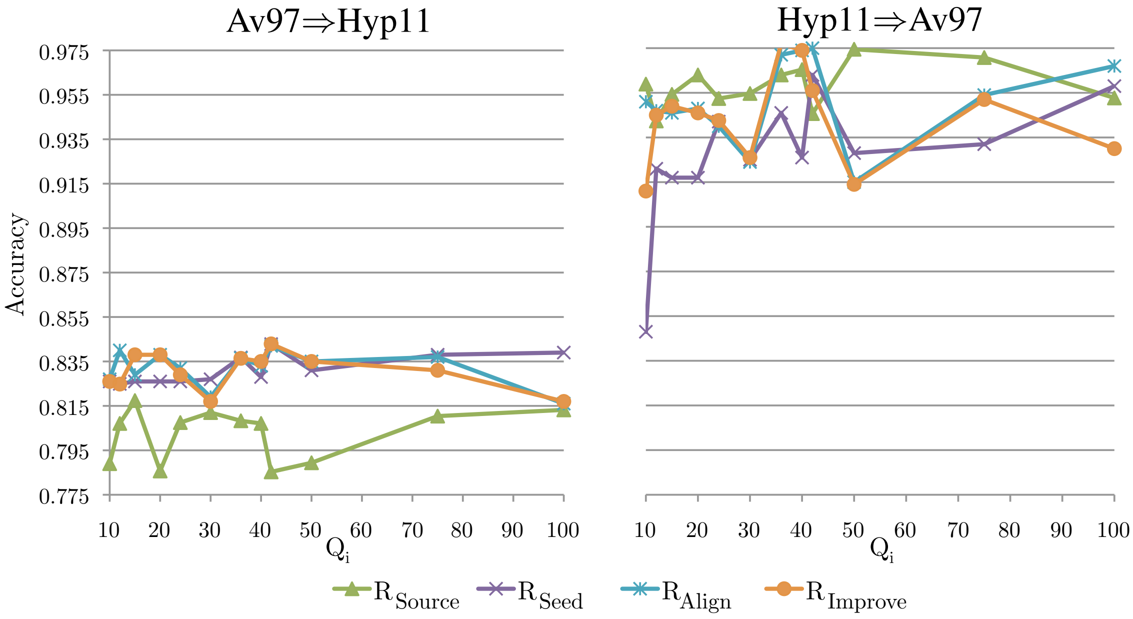


Figure 6.10 : R-space classification accuracy vs. number of seed samples Q_k for the Av97⇒Hyp11 (left) and Hyp11⇒Av97 (right) scenarios using source samples from the original source feature space (R_{S}, green △) vs. the MARTIAL seed ($\mathrm{R}_{\mathrm{Seed}}$, purple ×), align ($\mathrm{R}_{\mathrm{Align}}$, turquoise ∗) and improve ($\mathrm{R}_{\mathrm{Imp.}}$, orange ◦) feature spaces. We observe comparable or better performance in the R-space using the feature spaces produced by MARTIAL over the original feature space.

Table 6.5 provides a summary of the classification accuracies of each method, averaged over the range of Q_k values. We see that the MARTIAL feature space produced by the Align step yield the most accurate results in the Av97⇒Hyp11 scenario, and perform comparably to MCCL in the Hyp11⇒Av97 scenario. We also note that the accuracies produced after applying the Align step are typically equal or slightly better than those produced after the subsequent Improve step. This may be somewhat surprising, as one may expect that incorporating additional samples in the Improve step would produce a more robust alignment between the domains. However, since the pivots from each class are highly-correlated, using a large number of redundant pivots often produces worse results than using a smaller set of less-redundant pivots.

	ST	**Proc.**	**Seed**	**Align**	**Imp.**	$\mathbf{R_{S}}$	$\mathbf{R_{Seed}}$	$\mathbf{R_{Align}}$	$\mathbf{R_{Imp.}}$
Av97⇒Hyp11	71.49	73.29	81.69	82.35	81.67	80.27	*83.10*	*83.20*	83.08
Hyp11⇒Av97	93.99	82.75	93.43	94.28	94.05	*95.82*	92.69	*95.11*	94.36

Table 6.5 : Average accuracy over the range of selected Q_k values for each technique. The first and second most accurate results are given in red and blue italics, respectively.

Chapter 7

Conclusion

This thesis has advocated an *adaptive* approach to measuring similarity between spectral signatures for material identification tasks. By considering characteristics of spectral data, and accounting for the class-specific relationships most relevant to the given task, adaptive similarity measures can improve material identification accuracy over task-agnostic similarity measures and classification techniques that consider all spectral features equally relevant.

7.1 Contributions

We have made significant contributions to the field of automated spectral material identification. In Chapter 2, we demonstrated the feasibility of automated material identification using hyperspectral imagery by matching L^2-normalized spectra representing a diverse set of material classes to lab-measured material signatures using several distinct spectral similarity measures. Labels derived by our proposed material identification approach are determined by the contents of the library, the quality of the segmentation, and the similarity measure used to compare spectral signatures. We showed that spectral similarity measures that emphasize diagnostic absorption features can greatly improve material identification accuracy over baseline, task-agnostic similarity measures. Based upon these results, we proposed a new, hybrid similarity measure that accounts for both Continuum-Intact (CI) spectral shape and the positions/widths of diagnostic absorption features captured by Continuum-Removed (CR) signatures. We demonstrate that our novel measure, CICR, produces more accurate material identification results than the task-agnostic Euclidean Distance and Spectral Information Divergence similarity measures, both in terms of information-theoretic criteria, and visual inspection of the resulting library matches (Section 2.4.4).

We subsequently developed a technique to automatically determine a weighting between CI vs. CR distances for our CICR similarity measure using small amounts of labeled data (Section 3.1). We show that our technique yields improved classification accuracy in comparison to classification using CI or CR Euclidean distance measurements alone, and yields competitive performance to brute-force computation of the CI vs. CR weight parameter, at much reduced computational cost. We also demonstrate competitive classification performance using the adaptive CICR measure to several canonical feature selection techniques. We then generalized our technique to exploit the functional nature of spectral data by calculating the weighted relevances of spectral derivates using an adaptive form of the Sobolev distance (Section 3.3). Our analysis showed that the adaptive Sobolev metric produces more accurate results than the Euclidean baseline when distances between higher-order derivatives of spectral signatures are uncorrelated.

We evaluated similarity measures that assigned weights to individual spectral features in Chapter 4. We focused on the problem of learning low-rank Mahalanobis metrics from data, and provided a comprehensive evaluation of state-of-the-art Mahalanobis metric learning algorithms. We considered a diverse set of hyperspectral image classification problems, and our results indicated that, when properly regularized, multiclass LDA produced competitive or better classification performance, at significantly lower computational cost, than current algorithms. We also demonstrated that we can improve hyperspectral image segmentation results by augmenting a segmentation algorithm with a Mahalanobis measure learned from a small amount of labeled data (Section 4.3). We showed that the fidelity of the resulting image segments with respect to the labeled classes is improved, while we also observed a reduction in the number of spurious segments produced by noise or by features irrelevant to the labeled data. Our results also provided further evidence of the superiority of regularized LDA as a technique for low-rank Mahalanobis metric learning, in comparison to the Euclidean baseline, and the state-of-the-art Information Theoretic Metric Learning (ITML) algorithm.

In Part III, we broadened the scope of the material identification problem to inter-domain problems, where training (or source) and test (target) spectra are captured under different conditions – e.g., by different sensors, at different spatial locations or at different capture times. We proposed a novel, similarity-based domain adaptation framework, RelTrans, which calculates a mapping between a set of *source* domain spectra to a set of *target* domain spectra captured under similar, but not identical, conditions (Section 5.3). RelTrans captures structured, relative relationships between classes that are present in both the source and target domains by mapping them to a common feature space defined by relative distances to a set of canonical *pivot samples* representing identical classes in both domains. This mapping, applied as a similarity measure, allows us to classify samples from the target domain using a classifier trained using labeled source domain samples.

We considered the supervised domain adaptation setting in Chapter 5, where a small quantity of labeled target samples are available define the pivot sample-based mapping between the source and target domains. We provided a proof-of-concept of our RelTrans framework, RelSim, which adapted the MinDist classifier to the domain adaptation setting (Section 5.5.1). We applied RelSim to a multisensor domain adaptation task using a classifier trained on a set of synthetic hyperspectral image to classify materials from a spatially-overlapping multispectral image, and demonstrated improvements in classification accuracy ranging from 10-15% using the RelSim classifier over MinDist

(Section 5.6.2). We presented an extension to RelSim that automatically computed a threshold for its decision function based upon the set of source and target pivot samples, and illustrated effective outlier detection capabilites in the aforementioned synthetic multisensor domain adaptation problem, and on a hyperspectral domain adaptation problem involving synthetic imagery captured under varying atmospheric conditions (Section 5.7). We showed that our methodology enabled a classifier trained using synthetic spectra to classify similar materials in real hyperspectral imagery (Section 5.8). Additionally, we demonstrated the generality of the RelTrans framework by using several different classifiers trained using atmospherically-calibrated spectral reflectance signatures to classify uncalibrated spectra in radiance units (and vice-versa, Section 5.9.1). Our results showed competitive or better performance than several related multi-task learning algorithms.

We extended RelTrans to unsupervised domain adaptation settings in Chapter 6. We proposed the Multiclass Continuous Correspondence Learning (MCCL) algorithm in Section 6.2, which automatically selects pivot samples from the unlabeled target domain data that reflect the relative inter-class distances of the source pivots. When the source and target feature spaces are similar, in terms of the relative distances between classes, these target pivots typically represent the same classes as their corresponding source pivots. We also proposed a model-selection algorithm, Pdiv, which allows us to choose how many pivot samples are necessary to best reconcile the source and target domains. We applied MCCL and Pdiv to a synthetic four-class domain adaptation problem, and to a challenging multisource/multitemporal hyperspectral class knowledge transfer problem, and demonstrated comparable results to the supervised domain adaptation setting. In Section 6.5.2, we compared RelTrans to several baseline and related techniques on the aforementioned hyperspectral class knowledge transfer problem. Our results indicated that RelTrans outperforms each of the considered techniques in the unsupervised setting. In the supervised setting, we observed slightly improved accuracies over the baseline techniques when the target classes are better separated than the source classes, but slightly decreased accuracy when the target classes are less separable than the source classes. We showed in Section 6.7 that we can potentially improve our domain adaptation results by using a pivot selection strategy that leverages the functional nature of spectral signatures. Finally, we applied a manifold alignment approach based upon the TRIAL protein structure alignment algorithm to learn rigid transformations on a per-class basis to reconcile the source and target domains (Section 6.8).

7.2 Future Work

We can envision a number of directions for future research. In both intra-domain and inter-domain settings, to ensure the robustness of the material identification techniques we have developed, additional validation is essential on real spectral image data sets captured by different sensors, under varying enviornmental conditions, and containing diverse sets of material classes. Methods to incorporate additional contextual information in measuring spectral similarity, such as spatial relationships, can potentially improve material identification results [Hsieh and Landgrebe, 1999; Kim et al., 2008; Tarabalka, 2010], and recent work has demonstrated how such context can be incorporated into a similarity measure (e.g., [Lunga and Ersoy, 2012; Yang and Crawford, 2012]).

To conclude, we now reflect on several important open problems, and discuss potential directions for future research.

Theoretical Foundations of Similarity-based Domain Adaptation: Our experimental results from Part III suggest that, in the case when between-class distances are relatively preserved across domains, we can define a mapping from the source to the target domains based on distances between samples representing similar classes in both domains. Furthermore, as we demonstrated in Chapter 6, we can evaluate the quality of this mapping by measuring the H-divergence [Ben-David et al., 2010a] between the source and target pivots in the R-space (Algorithm 6.2). Our motivation for this approach was the generalization bound for domain adaptation problems proposed by Ben-David et al. [2010a, 2007], which gave a generalization bound on target domain accuracy for inter-domain classification problems based upon the generalization performance in the source domain, the H-divergence between the domains, and the complexity of the classification problem. Their bound could potentially be combined with the classification bounds proposed by Balcan et al. [Balcan et al., 2006; Balcan et al., 2008a,b] for distance and kernel-based transformations, such as our R-transform Equation (5.1). However, the bounds proposed by Balcan et al. do not extend directly to the inter-domain problems described in Section 5.2. Moreover, their applicability to multiclass domain adaptation settings remains an open question, as the bounds derived in [Ben-David et al., 2010a] and [Balcan et al., 2008b] assume the classification problem is binary. While it is possible to decompose the multiclass problem into multiple binary classification problems, as we showed empirically in Section 5.9.2, such decompositions may not yield good performance in domain adaptation settings, and suggest that generalization bounds using binary decompositions would be particularly loose. However, even if such bounds are not directly applicable in practical settings, as we discussed in Section 5.9.3, they often provide valuable insight into the conditions where domain adaptation is possible.

Feature-weighted Metrics for Domain Adaptation: The connections between Mahalanobis metric learning and domain adaptation could be further explored. One avenue of particular interest is exploring the relationship between the rigid transformations computed using the MARTIAL algorithm, and the corresponding intra-domain and inter-domain Mahalanobis metrics induced by these transformations. Specifically, the Euclidean distance between source samples $\mathbf{x}_i^S$ and $\mathbf{x}_j^S$ from the same class k after applying MARTIAL transformation matrix $\mathbf{T}_k$ is equivalent to the Mahalanobis distance parameterized by $\mathbf{M} = \mathbf{T}_k^T \mathbf{T}_k$. More interestingly, the Euclidean distance between transformed source sample $\mathbf{T}_k \mathbf{x}_i^S$ and target sample $\mathbf{x}_j^T$ from class k roughly approximates the Euclidean distance between target samples in the same class, by virtue of the fact that $\mathbf{T}_k$ maps the source samples from class k to the target domain feature space. This relationship largely explains the improvement in the R-space classification accuracy shown in Figure 6.10, as the relative distances between the source and target classes are better resolved in the MARTIAL feature spaces. However, we could potentially improve our results by incorporating constraints to make the $\mathbf{T}_k$ reflect the relative distances between the source classes.

Another possible direction is to apply feature-weighted metric learning/feature selection techniques to emphasize the most relevant dimensions of the R-space. By learning the most relevant dimensions of the R-space, we can eliminate uninformative pivot samples and reduce the dimensionality of the R-space, which can potentially improve classification performance in cases when the number of pivots is large. Recent work by Quattoni et al. [2008] has demonstrated similar approaches can improve multitask image classification results, but additional work is necessary to determine how well such techniques generalize to multiclass domain adaptation settings.

Object-level Material Identification: In this work, we concentrated on identification of materials of unlabeled spectra according to their relationships to spectra with known material labels. However, as mentioned in Section 2.4.5, spectra are often labeled according to the objects to which they belong, rather than their material composition, and require manual inspection to translate object to material labels. We can potentially infer the material composition of spectra with object labels by cross-referencing them with spectral libraries, while constraining the set of candidate materials for each object using Natural Language Processing (NLP) techniques to measure the *semantic* similarity between labels.

Autonomous Material Identification Onboard Spacecraft: A more long-term objective of this work is to deploy our material identification techniques directly onboard spacecraft. However, spacecraft platforms present unique challenges due to limited communication bandwidth and computational capacity [McGovern and Wagstaff, 2011], and are subject to extreme conditions that can potentially cause measurement errors [Wagstaff and Bornstein, 2009]. Consequently, algorithms deployed onboard spacecraft must be capable of robust anomaly detection, while also operating efficiently in terms of CPU and memory resources. We have demonstrated that the algorithms developed in this thesis meet the efficiency requirements, and future efforts will incorporate our algorithms into ongoing automated onboard material identification efforts, such as those described by Bornstein et al. [2011]; Thompson et al. [2012].

Author Bibliography

BJ Bornstein, DR Thompson, D Tran, BD Bue, SA Chien, and R Castaño. "Efficient spectral endmember detection onboard the EO-1 spacecraft". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (2011) (cit. on pp. 4, 219).

BD Bue. *Low-rank Mahalanobis Metric Learning for Hyperspectral Image Classification: A Comparative Survey.* Tech. rep. Rice University (in preparation), 2013 (cit. on p. 88).

BD Bue and C Jermaine. "Multiclass Domain Adaptation with Iterative Manifold Alignment". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS) (to appear)* (2013) (cit. on p. 175).

BD Bue and E Merényi. "An Adaptive Similarity Measure for Classification of Hyperspectral Signatures". *IEEE Geoscience and Remote Sensing Letters* 10.2 (2012), pp. 381–385 (cit. on p. 65).

— "Using spatial correspondences for hyperspectral knowledge transfer: evaluation on synthetic data". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (June 2010) (cit. on pp. 133, 141).

BD Bue and DR Thompson. "Multiclass Continuous Correspondence Learning". *NIPS Domain Adaptation Workshop* (Dec. 2011) (cit. on pp. 175, 208).

BD Bue, E Merényi, and B Csathó. "An Evaluation of Class Knowledge Transfer from Real to Synthetic Imagery". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (June 2011) (cit. on p. 133).

— "Automated Labeling of Materials in Hyperspectral Imagery". *IEEE Trans. on Geoscience and Remote Sensing* 48.11 (2010), pp. 4059–4070 (cit. on pp. 31, 65).

— "Automated Labeling of Segmented Hyperspectral Imagery via Spectral Matching". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (Aug. 2009) (cit. on pp. 31, 65).

BD Bue, DR Thompson, MS Gilmore, and R Castaño. "Metric Learning for Hyperspectral Image Segmentation". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (2011) (cit. on pp. 88, 93).

DR Thompson, BJ Bornstein, SA Chien, S Schaffer, D Tran, BD Bue, RC no, D Gleeson, and A Noell. "Autonomous Spectral Discovery and Mapping Onboard the EO-1 Spacecraft". *IEEE Trans. on Geoscience and Remote Sensing* (2012) (cit. on pp. 4, 219).

Bibliography

M Abrams, R Ashley, L Rowan, A Goetz, and A Kahle. “Mapping of hydrothermal alteration in the Cuprite mining district, Nevada, using aircraft scanner images for the spectral region 0.46 to 2.36 m”. *Geology* 5.12 (1977), pp. 713–718 (cit. on p. 1).

DH Ackley, GE Hinton, and TJ Sejnowski. “A learning algorithm for Boltzmann machines”. *Cognitive Science* 9.1 (1985), pp. 147–169 (cit. on p. 21).

J Adams and A Gillespie. *Remote sensing of landscapes with spectral images: A physical modeling approach*. Cambridge Univ Pr, 2006 (cit. on pp. 2, 12, 16, 140).

J Adams, M Smith, and A Gillespie. “Simple models for complex natural surfaces- A strategy for the hyperspectral era of remote sensing”. *Quantitative remote sensing: An economic tool for the Nineties* (1989), pp. 16–21 (cit. on pp. 15–16).

B Alipanahi, M Biggs, and A Ghodsi. “Distance metric learning vs. Fisher discriminant analysis”. *Proceedings AAAI 2008* (2008) (cit. on pp. 6, 91, 120).

A Argyriou, T Evgeniou, and M Pontil. “Multi-task feature learning”. *Advances in Neural Information Processing Systems* 19 (2007), p. 41 (cit. on p. 166).

M Balcan, A Blum, and S Vempala. “Kernels as features: On kernels, margins, and low-dimensional mappings”. *Machine Learning* (2006), pp. 79–94 (cit. on pp. 5, 30, 217).

MF Balcan, A Blum, and N Srebro. “A theory of learning with similarity functions”. *Machine Learning* 72.1 (2008), pp. 89–112 (cit. on pp. 5, 20, 217).

MF Balcan, A Blum, and S Vempala. “On Kernels, Margins and Low-dimensional Mappings”. *Proc. of Algorithmic Learning Theory 2008* (2008), pp. 1–12 (cit. on p. 217).

AM Baldridge, SJ Hook, CI Grove, and G Rivera. “The ASTER spectral library version 2.0”. *Remote Sensing of Environment* 113.4 (Jan. 2009), pp. 711–715 (cit. on p. 31).

T Bandos, L Bruzzone, and G Camps-Valls. “Efficient regularized LDA for hyperspectral image classification”. *Proc. of SPIE: Image and Singnal Processing for remote Sensing XIII* 6748.1 (2007) (cit. on p. 29).

TV Bandos, L Bruzzone, and G Camps-Valls. “Classification of Hyperspectral Images With Regularized Linear Discriminant Analysis”. *IEEE Trans. on Geoscience and Remote Sensing* 47.3 (2009), pp. 862–873 (cit. on p. 91).

R Basedow, D Carmer, and M Anderson. “HYDICE system: implementation and performance”. *Proceedings of SPIE* 2480 (1995), pp. 258–267 (cit. on p. 144).

S Ben-David. “Inductive transfer via embeddings into a common feature space”. *Open House on Multi-Task and Complex Outputs Learning*. July 2006 (cit. on p. 184).

S Ben-David, J Blitzer, K Crammer, A Kulesza, F Pereira, and JW Vaughan. “A theory of learning from different domains”. *Machine Learning* 79.1-2 (May 2010), pp. 151–175 (cit. on pp. 168–169, 217).

— “A theory of learning from different domains”. *Mach Learn* 79.1-2 (2010), pp. 151–175 (cit. on p. 176).

S Ben-David, J Blitzer, K Crammer, and F Pereira. “Analysis of Representations for

Domain Adaptation". *Advances in Neural Information Processing Systems* (2007) (cit. on p. 217).

J Benediktsson, J Sveinsson, and K Arnason. "Classification and Feature-Extraction of AVIRIS Data". *IEEE Trans. on Geoscience and Remote Sensing*. 1995, pp. 1194–1205 (cit. on p. 5).

A Berg and A Jensen. "Robust classification of hyperspectral images". *Proc. of SPIE* (2007) (cit. on p. 5).

J Blitzer, R McDonald, and F Pereira. "Domain adaptation with structural correspondence learning". *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (2006) (cit. on p. 136).

L Bruzzone and M Marconcini. "Domain Adaptation Problems: A DASVM Classification Technique and a Circular Validation Strategy". *IEEE Trans. on Pattern Analysis and Machine Intelligence* 32.5 (2010), pp. 770–787 (cit. on pp. 137, 195).

G Camps-Valls and L Bruzzone. "Kernel-based methods for hyperspectral image classification". *Geoscience and Remote Sensing, IEEE Transactions on* 43.6 (2005), pp. 1351–1362 (cit. on pp. 4, 25).

R Caruana. "Multitask learning". *Machine Learning* 28.1 (1997), pp. 41–75 (cit. on p. 138).

L Cazzanti. "Generative Models for Similarity-based Classification". PhD thesis. University of Washington, 2007. ISBN: 9780549375517 (cit. on p. 20).

C Chang and C Lin. "LIBSVM: a library for support vector machines". *ACM Transactions on Intelligent Systems and Technology (TIST)* 2.3 (2011), p. 27 (cit. on pp. 162, 178, 208).

CI Chang. "An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis". *IEEE Trans. on Information Theory* 46.5 (2000), pp. 1927–1932 (cit. on pp. 25, 34, 36, 38, 58).

M Chen, KQ Weinberger, and J Blitzer. "Co-Training for Domain Adaptation". *Proceedings of NIPS* (2011) (cit. on p. 139).

Y Chen, E Garcia, M Gupta, A Rahimi, and L Cazzanti. "Similarity-based classification: Concepts and algorithms". *The Journal of Machine Learning Research* 10 (2009), pp. 747–776 (cit. on p. 20).

SA Chien, R Sherwood, D Tran, B Cichy, G Rabideau, R Castaño, A Davies, D Mandl, S Frye, and B Trout. "Using autonomy flight software to improve science return on Earth Observing One". *Journal of Aerospace Computing, Information, and Communication* 2.4 (2005), pp. 196–216 (cit. on p. 4).

M Ciznicki, K Kurowski, and A Plaza. "Graphics processing unit implementation of JPEG2000 for hyperspectral image compression". *Journal of Applied Remote Sensing* 6.1 (Jan. 2012), p. 061507 (cit. on p. 2).

RN Clark. "Chapter 1: Spectroscopy of Rocks and Minerals, and Principles of Spectroscopy". *Manual of Remote Sensing*. John Wiley and Sons, 1999, pp. 3–58 (cit. on p. 13).

RN Clark and T Rousch. “Reflectance spectroscopy: quantitative analysis techniques for remote sensing applications”. *Journal of Geophysical Research-Solid Earth* 89.B7 (1984), pp. 6329–6340 (cit. on pp. 15, 35, 53).

RN Clark, T King, and N Gorelick. “Automatic continuum analysis of reflectance spectra”. *JPL Proceedings of the 3rd Airborne Imaging Spectrometer Data Analysis Workshop p 138-142(SEE N 88-13755 05-42)* (1987) (cit. on pp. 13, 35).

RN 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* 108.E12 (2003), p. 5131 (cit. on pp. 3, 6, 26–27).

RN Clark, AJ Gallagher, and GA Swayze. *Material absorption-band depth mapping of imaging spectrometer data using the complete band shape least-squares algorithm simultaneously fit to multiple spectral features from multiple materials.* Tech. rep. USGS, 1990 (cit. on pp. 6, 26).

RN Clark, G Swayze, R Wise, E Livo, T Hoefen, R Kokaly, and S Sutley. *USGS digital spectral library splib06a: U.S. Geological Survey, Digital Data Series 231.* Tech. rep. USGS, 2007 (cit. on pp. 31, 42).

T Cocks, R Jenssen, A Stewart, I Wilson, and T Shields. “The HyMap airborne hyperspectral sensor: the system, calibration and performance”. *Proceedings of the 1st EARSeL workshop on Imaging Spectroscopy* (1998), pp. 37–42 (cit. on p. 144).

C Cortes and V Vapnik. “Support-Vector Networks”. *Machine Learning* 20.3 (1995), pp. 273–297 (cit. on p. 23).

B Csathó, W Krabill, J Lucas, and T Schenk. “A multisensor data set of an urban and coastal scene”. *International Archives of Photogrammetry and Remote Sensing* XXXII (3/2) (1998), pp. 26–31 (cit. on p. 40).

H Daume. “Frustratingly Easy Domain Adaptation”. *Annual Meeting - Association for Computational Linguistics* (2007) (cit. on pp. 139, 189).

AG Davies et al. “Monitoring active volcanism with the Autonomous Sciencecraft Experiment on EO-1”. *Remote Sensing of Environment* 101.4 (Apr. 2006), pp. 427–446 (cit. on p. 1).

J Davis, B Kulis, P Jain, S Sra, and I Dhillon. “Information-theoretic metric learning”. *Proc. of the 24th International Conference on Machine Learning* (2007) (cit. on pp. 6, 28, 89–90, 101–103, 120, 125).

J Demšar. “Statistical comparisons of classifiers over multiple data sets”. *The Journal of Machine Learning Research* 7 (2006), pp. 1–30 (cit. on p. 40).

TG Dietterich. “Ensemble methods in machine learning”. *Multiple Classifier Systems* (2000), pp. 1–15 (cit. on p. 76).

Q Du. “Modified Fisher’s Linear Discriminant Analysis for Hyperspectral Imagery”. *IEEE Geoscience and Remote Sensing Letters* 4.4 (2007), pp. 503–507 (cit. on p. 29).

Q Du and CI Chang. “Hidden Markov model approach to spectral analysis for hyperspectral imagery”. *Optical Engineering* 40.10 (2001), p. 2277 (cit. on pp. 34, 38, 59).

Y Du, CI Chang, H Ren, CC Chang, JO Jensen, and FM D’amico. “New hyperspectral discrimination measure for spectral characterization”. *Optical Engineering* 43.8 (2004), p. 1777 (cit. on pp. 38, 59).

KB Duan and SS Keerthi. “Which is the best multiclass SVM method? An empirical study”. *Multiple Classifier Systems* (2005), pp. 278–285 (cit. on p. 165).

RO Duda and PE Hart. *Pattern recognition and scene analysis*. Wiley, New York, 1973 (cit. on p. 20).

P Felzenszwalb and D Huttenlocher. “Efficient graph-based image segmentation”. *International Journal of Computer Vision* 59.2 (2004), pp. 167–181 (cit. on p. 122).

R Fisher. “The statistical utilization of multiple measurements.” *Annals of Eugenics* 8 (1938), pp. 376–386 (cit. on p. 93).

— “The use of multiple measurements in taxonomic problems”. *Annals of eugenics* 7.2 (1936), pp. 179–188 (cit. on pp. 29, 68, 92).

A Ghodsi, M Biggs, and B Alipanahi. “Distance metric learning vs. Fisher discriminant analysis”. *Proceedings AAAI 2008* (2008) (cit. on pp. 29, 92).

MS Gilmore, R Castaño, BJ Bornstein, and J Greenwood. “Autonomous mineral detectors for visible/near-infrared spectrometers at Mars”. *Seventh International Conference on Mars(LPI Contribution No. 1353)* (2007) (cit. on p. 1).

MS Gilmore, BJ Bornstein, MD Merrill, R Castaño, and JP Greenwood. “Generation and performance of automated jarosite mineral detectors for visible/near-infrared spectrometers at Mars”. *Icarus* 195.1 (May 2008), pp. 169–183 (cit. on p. 3).

DF Gleeson, RT Pappalardo, SE Grasby, MS Anderson, B Beauchamp, R Castaño, SA Chien, T Doggett, L Mandrake, and KL Wagstaff. “Characterization of a sulfur-rich Arctic spring site and field analog to Europa using hyperspectral data”. *Remote Sensing of Environment* 114.6 (June 2010), pp. 1297–1311 (cit. on p. 4).

A Globerson and S Roweis. “Metric learning by collapsing classes”. *Advances in Neural Information Processing Systems* 18 (2006), p. 451 (cit. on pp. 28–29, 89–91, 98, 120).

A Goetz, G Vane, J Solomon, and B Rock. “Imaging spectrometry for earth remote sensing”. *Science* 228.4704 (1985), pp. 1147–1153 (cit. on p. 15).

J Goldberger, S Roweis, G Hinton, and R Salakhutdinov. “Neighbourhood components analysis”. *Advances in Neural Information Processing Systems* (2005) (cit. on pp. 28, 89–90).

— “Neighbourhood components analysis”. *Advances in Neural Information Processing Systems* 17 (2005), pp. 513–520 (cit. on p. 97).

R Green, M Eastwood, C Sarture, T Chrien, M Aronsson, B Chippendale, J Faust, B Pavri, C Chovit, and M Solis. “Imaging spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)”. *Remote Sensing of Environment* 65.3 (1998), pp. 227–248 (cit. on p. 40).

A Gretton, A Smola, J Huang, M Schmittfull, K Borgwardt, and B Schölkopf. "Covariate shift by kernel mean matching". *Dataset shift in machine learning* (2009), pp. 131–160 (cit. on pp. 169, 195).

JA Gualtieri and RF Cromp. "Support vector machines for hyperspectral remote sensing classification". *Proc. SPIE* 3584 (Jan. 1999), pp. 221–232 (cit. on p. 4).

I Guyon and A Elisseeff. "An introduction to variable and feature selection". *The Journal of Machine Learning Research* (2003) (cit. on pp. 5, 77).

I Guyon, J Weston, S Barnhill, and V Vapnik. "Gene selection for cancer classification using support vector machines". *Machine Learning* 46.1 (2002), pp. 389–422 (cit. on p. 77).

B Hammer and T Villmann. "Generalized relevance learning vector quantization". *Neural Networks* 15.8-9 (2002), pp. 1059–1068 (cit. on p. 21).

L Hansen and P Salamon. "Neural Network Ensembles". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.10 (1990), pp. 993–1001 (cit. on p. 76).

T Hastie, R Tibshirani, and JH Friedman. *The Elements of Statistical Learning*. 2nd ed. Springer, Feb. 2011. ISBN: 0387848576 (cit. on pp. 68–69).

DS Hayden, S Chien, D Thompson, and R Castaño. "Using Clustering and Metric Learning to Improve Science Return of Remote Sensed Imagery (preprint)". *ACM Trans. on Intelligent Systems and Technology: Special Issue: Intelligent Systems in Space* (2011), pp. 1–20 (cit. on p. 92).

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.* 91 (2004), pp. 304–319 (cit. on pp. 1, 42, 51, 53, 55).

T Hertz. "Learning Distance Functions: Algorithms and Applications". PhD thesis. Hebrew University, 2006 (cit. on pp. 6, 24).

S Hoi, W Liu, M Lyu, and WY Ma. "Learning Distance Metrics with Contextual Constraints for Image Retrieval". *CVPR* (2006) (cit. on pp. 95, 120).

SJ Hook, J Myers, K Thome, M Fitzgerald, and A Kahle. "The MODIS/ASTER airborne simulator (MASTER) - a new instrument for earth science studies". *Remote Sensing of Environment* 76.93 (2001), p. 102 (cit. on p. 144).

JJ Hopfield. "Neural networks and physical systems with emergent collective computational abilities". *Proceedings of the National Academy of Sciences* 79.8 (1982), pp. 2554–2558 (cit. on p. 21).

RA Horn and CA Johnson. *Matrix Analysis*. 1st. Cambridge University Press, 1985 (cit. on p. 69).

E Howell, E Merényi, and L Lebofsky. "Classification of asteroid spectra using a neural network". *Journal of Geophysical Research* 99.E5 (1994), pp. 10847–10,865 (cit. on pp. 1, 35).

P Hsieh and DA Landgrebe. "Statistics enhancement in hyperspectral data analysis using spectral-spatial labeling, the EM algorithm, and the leave-one-out covariance estimator". *Proc. SPIE* 3438 (1999), pp. 183–190 (cit. on p. 216).

C Hsu and C Lin. “A comparison of methods for multiclass support vector machines”. *Neural Networks, IEEE Transactions on* 13.2 (2002), pp. 415–425 (cit. on p. 23).

G Hughes. “On the mean accuracy of statistical pattern recognizers”. *IEEE Trans. on Information Theory* 14.1 (1968), pp. 55–63 (cit. on p. 18).

G Hunt and J Salisbury. “Visible and near-infrared spectra of minerals and rocks: I. Silicate Minerals”. *Modern Geology* 1 (1970), pp. 283–300 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks: II. Carbonates”. *Modern Geology* 2 (1971), pp. 23–30 (cit. on p. 12).

— “Visible and near infrared spectra of minerals and rocks. XI- Sedimentary rocks.” *Modern Geology* 5 (1976), pp. 211–217 (cit. on p. 12).

— “Visible and near infrared spectra of minerals and rocks. XII- Metamorphic rocks”. *Modern Geology* 5 (1976), pp. 219–228 (cit. on p. 12).

G Hunt, J Salisbury, and C Lenhoff. “Visible and near-infrared spectra of minerals and rocks: III. Oxides and Hydroxides”. *Modern Geology* 2 (1971), pp. 195–205 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks: IV. Intermediate Igneous Rocks”. *Modern Geology* 4 (1973), pp. 237–244 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks: IV. Sulphides and sulphates”. *Modern Geology* 3 (1971), pp. 1–14 (cit. on p. 12).

— “Visible and near infrared spectra of minerals and rocks: IX. Basic and ultrabasic igneous rocks”. *Modern Geology* 5 (1974), pp. 15–22 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks. V. Halides, phosphates, arsenates, vanadates, and borates”. *Modern Geology* 3 (1972), pp. 121–132 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks: VI. Additional silicates”. *Modern Geology* 4 (1973), pp. 85–106 (cit. on p. 12).

— “Visible and near-infrared spectra of minerals and rocks: VII. Acidic Igneous Rocks”. *Modern Geology* 4 (1973), pp. 217–224 (cit. on p. 12).

P Jain, B Kulis, JV Davis, and IS Dhillon. “Metric and Kernel Learning using a Linear Transformation”. *arXiv.org* cs.LG (Oct. 2009) (cit. on p. 90).

S Ji and J Ye. “An accelerated gradient method for trace norm minimization” (2009), pp. 457–464 (cit. on p. 166).

R Jin, S Wang, and Y Zhou. “Regularized Distance Metric Learning: Theory and Algorithm”. *Proc. 22nd Advances in Neural Information Processing Systems* (2009) (cit. on p. 100).

W Kabsch. “A discussion of the solution for the best rotation to relate two sets of vectors”. *Acta Crystallographica Section A: Crystal Physics* 34 (Sept. 1978), pp. 827–828 (cit. on p. 204).

P Kar and P Jain. “Similarity-based Learning via Data Driven Embeddings”. *Advances in Neural Information Processing Systems* (Dec. 2011) (cit. on p. 20).

M Kästner, B Hammer, M Biehl, and T Villmann. “Generalized functional relevance learning vector quantization”. *ESANN 2011* (2011) (cit. on p. 22).

N Keshava. “Distance Metrics and Band Selection in Hyperspectral Processing with Applications to Material Identification and Spectral Libraries”. *IEEE Trans. on Geoscience and Remote Sensing* (2004) (cit. on pp. 4, 25, 34).

N Keshava and J Mustard. “Spectral unmixing”. *Signal Processing Magazine, IEEE* 19.1 (2002), pp. 44–57 (cit. on p. 16).

D Kifer, S Ben-David, and J Gehrke. “Detecting change in data streams”. *Proc. of the International Conference on Very Large Databases* 30 (2004), pp. 180–191 (cit. on p. 169).

W Kim and MM Crawford. “Adaptive classification for hyperspectral image data using manifold regularization kernel machines”. *Geoscience and Remote Sensing* (2010) (cit. on p. 137).

W Kim, MM Crawford, and J Ghosh. “Spatially Adapted Manifold Learning for Classification of Hyperspectral Imagery with Insufficient Labeled Data”. *Proc. 2008 International Geosci. and Sens. Symposium (IGARSS08)* (2008) (cit. on pp. 137, 216).

M King, D Herring, and D Diner. “Earth observing system: a space-based program for assessing mankind’s impact on the global environment”. *Optics and Photonics News* 6.1 (1995), pp. 34–39 (cit. on p. 1).

T Kohonen. *Self Organizing Maps.* Ed. by T Huang, T Kohonen, and MR Schroeder. 1st. Springer-Verlag, 1995 (cit. on p. 21).

FA Kruse, J Boardman, and J Huntington. “Comparison of airborne hyperspectral data and EO-1 Hyperion for mineral mapping”. *IEEE Trans. on Geoscience and Remote Sensing* 41.6 (2003), pp. 1388–1400 (cit. on p. 180).

B Kulis, M Sustik, and I Dhillon. “Low-Rank Kernel Learning with Bregman Matrix Divergences”. *The Journal of Machine Learning Research* 10 (2009), pp. 341–376 (cit. on p. 100).

S Kullback and R Leibler. “On information and sufficiency”. *The Annals of Mathematical Statistics* (1951), pp. 79–86 (cit. on p. 25).

BC Kuo and DA Landgrebe. “Improved statistics estimation and feature extraction for hyperspectral data classification”. PhD thesis. Purdue University ECE, 2001 (cit. on p. 5).

DA Landgrebe. “On information extraction principles for hyperspectral data”. *White Paper, Purdue University* (1997) (cit. on p. 5).

— “The Application of pattern recognition techniques to a remote sensing problem”. *Adaptive Processes, 1968. Seventh Symposium on* 7 (1968), p. 22 (cit. on p. 3).

C Lee and DA Landgrebe. “Analyzing high-dimensional multispectral data”. *IEEE Trans. on Geoscience and Remote Sensing* 31.4 (1993), pp. 792–800 (cit. on p. 3).

WJ Lee, RP Duin, A Ibba, and M Loog. “An Experimental Study On Combining Euclidean Distances”. *The 2nd International Workshop on Cognitive Information Processing* (2010) (cit. on pp. 59, 77, 83).

C Lenart, P Burai, A Smailbegovic, T Biro, Z Katona, and R Andricevic. “Multi-Sensor Integration And Mapping Strategies For The Detection And Remediation

Of The Red Mud Spill In Kolontar, Hungary: Estimating The Thickness Of The Spill Layer Using Hyperspectral Imaging And Lidar". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (Feb. 2011), pp. 1–4 (cit. on p. 1).

J Li, D Hibbert, S Fuller, and G Vaughn. "A comparative study of point-to-point algorithms for matching spectra". *Chemometrics and Intelligent Laboratory Systems* (2006) (cit. on p. 25).

D Lunga and O Ersoy. *Nonlinear Dynamic Field Embedding: On Hyperspectral Scene Visualization*. Tech. rep. 439. 2012 (cit. on p. 216).
URL: http://docs.lib.purdue.edu/ecetr/439/

PC Mahalanobis. "On the Generalized Distance in Statistics". *Proceedings of the National Institute of Sciences of India* 2.1 (1936), pp. 49–55 (cit. on pp. 28, 60).

L Mandrake, DR Thompson, MS Gilmore, R Castaño, and E Dobrea. "Automated Neutral Region Detection Using Superpixels". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (2010) (cit. on p. 105).

CD Manning, P Raghavan, and H Schutze. *Introduction to information retrieval*. Vol. 1. Cambridge University Press Cambridge, 2008 (cit. on p. 77).

A McGovern and KL Wagstaff. "Machine learning in space: extending our reach". *Machine Learning* 84.3 (Apr. 2011), pp. 335–340 (cit. on p. 219).

MJ Mendenhall and E Merényi. "On the evaluation of synthetic hyperspectral imagery". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (Aug. 2009), pp. 1–4 (cit. on p. 173).

MJ Mendenhall. "A Neural Relevance Model for Feature Extraction from Hyperspectral Images, and Its Application in the Wavelet Domain". PhD thesis. Rice University, 2006 (cit. on p. 5).

MJ Mendenhall and E Merényi. "Relevance-Based Feature Extraction for Hyperspectral Images". *Neural Networks, IEEE Transactions on* 19.4 (2008), pp. 658–672 (cit. on p. 5).

E Merényi. ""Precision Mining" of High-Dimensional Patterns with Self-Organizing Maps: Interpetation of Hyperspectral Images". *Quo Vadis Computational Intelligence: New Trends and Approaches in Computational Intelligence (Studies in Fuzziness and Soft Computing)*. 54 (2000), pp. 1–15 (cit. on pp. 15, 90).

— "Self-organizing ANNs for planetary surface composition research". *Proc. 6th European Symposium on Artificial Neural Networks, ESANN* 98 (1998), pp. 22–24 (cit. on pp. 4, 15, 21).

E Merényi, WH Farrand, JV Taranik, and TB Minor. "Classification of hyperspectral imagery with neural networks: Comparison to conventional tools". *Machine Learning Reports* 5 (2011), pp. 1–15 (cit. on p. 21).

E Merényi, K Tasdemir, and WH Farrand. "Intelligent information extraction to aid science decision making in autonomous space exploration". *Proc. of SPIE* 6960 (2009) (cit. on p. 145).

E Merényi, B Csathó, and K Tasdemir. “Knowledge discovery in urban environments from fused multi-dimensional imagery”. *Proc. 4th IEEE GRSS/ISPRS Joint Workshop on Remote Sensing Data Fusion over Urban Areas* (2007), pp. 1–13 (cit. on pp. 41–43, 57, 71, 155).

E Merényi, WH Farrand, L Stevens, T Melis, and K Chhibber. “Mapping Colorado River ecosystem resources in Glen Canyon: Analysis of hyperspectral low-altitude AVIRIS imagery”. *Proc. of ERIM, 14th Int’l Conf. and Workshops on Applied Geologic Rem. Sens.* (2000) (cit. on pp. 1, 4, 43).

E Merényi, R Singer, and J Miller. “Mapping of spectral variations on the surface of mars from high spectral resolution telescopic images”. *Icarus* 124.1 (1996), pp. 280–295 (cit. on p. 1).

F Morgan, F Seelos, and S Murchie. “CAT tutorial”. *CRISM Data User’s Workshop, Lunar Planetary Sci. Conf.* 2009 (cit. on p. 105).

S Murchie et al. “CRISM (Compact Reconnaissance Imaging Spectrometer for Mars) on MRO (Mars Reconnaissance Orbiter)”. *Journal of Geophysical Research* 112.E05 (2007) (cit. on pp. 2, 105).

E Mwebaze, P Schneider, FM Schleif, JR Aduwo, JA Quinn, S Haase, T Villmann, and M Biehl. “Divergence-based classification in learning vector quantization”. *Neurocomputing* 74.9 (Apr. 2011), pp. 1429–1435 (cit. on p. 25).

RR Nidamanuri and B Zbell. “Normalized Spectral Similarity Score (NS3) as an Efficient Spectral Library Searching Method for Hyperspectral Image Classification”. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 4.1 (2011), pp. 226–240 (cit. on p. 54).

X Niyogi. “Locality preserving projections”. *Advances in Neural Information Processing Systems 16: Proceedings of the 2003 Conference* 16 (2003), p. 153 (cit. on p. 93).

G Obozinski, B Taskar, and MI Jordan. “Joint covariate selection and joint subspace selection for multiple classification problems”. *Statistics and Computing* 20.2 (Jan. 2009), pp. 231–252 (cit. on p. 166).

E Pekalska. “Dissimilarity representations in pattern recognition”. PhD thesis. Delft University, 2005 (cit. on p. 20).

S Pelkey, J Mustard, S Murchie, R Clancy, M Wolff, M Smith, R Milliken, J Bibring, A Gendrin, and F Poulet. “CRISM multispectral summary products: Parameterizing mineral diversity on Mars from reflectance”. *Journal of Geophysical Research* 112 (2007), pp. 1–18 (cit. on p. 1).

C Persello and L Bruzzone. “A novel active learning strategy for domain adaptation in the classification of remote sensing images”. *IEEE Geoscience and Remote Sensing Symposium* (2011), pp. 3720–3723 (cit. on p. 137).

M Pieters. *Reflectance Experiment Laboratory Description and User’s Manual.* Brown University. 1990 (cit. on p. 31).

E Pkalska and RP Duin. “Dissimilarity representations allow for building good classifiers”. *Pattern Recognition Letters* 23.8 (2002), pp. 943–956 (cit. on p. 29).

G Pouch and D Campagna. “Hyperspherical direction cosine transformation for separation of spectral and illumination information in digital scanner data”. *Photogrammetric Engineering and Remote Sensing* 56.4 (1990), pp. 475–479 (cit. on p. 105).

A Quattoni, M Collins, and T Darrell. “Transfer learning for image classification with sparse prototype representations”. *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008* (2008), pp. 1–8 (cit. on p. 218).

S Rajan, J Ghosh, and MM Crawford. “Exploiting Class Hierarchies for Knowledge Transfer in Hyperspectral Data”. *IEEE Trans. on Geoscience and Remote Sensing* 44.11 (2006), pp. 3408–3417 (cit. on pp. 138, 148).

C Rao. “The utilization of multiple measurements in problems of biological classification”. *Journal of the Royal Statistical Society* 10.2 (1948), pp. 159–203 (cit. on pp. 29, 68).

P Rauss, J Daida, and S Chaudhary. “Classification of spectral imagery using genetic programming”. *Proc. Genetic Evolutionary Computation Conf.* 1001 (2000), p. 48109 (cit. on p. 3).

SA Robila. “An analysis of spectral metrics for hyperspectral image processing”. *IEEE Geoscience and Remote Sensing Symposium* 5 (Sept. 2004), pp. 3233–3235 (cit. on p. 25).

HP Ross, JEM Adler, and GR Hunt. “A Statistical Analysis of the Reflectance of Igneous Rocks from 0. 2 to 2. 65 Microns”. *Icarus* 11 (July 1969), p. 46 (cit. on p. 16).

TL Roush and RB Singer. “Gaussian analysis of temperature effects on the reflectance spectra of mafic minerals in the 1-m region”. *Journal of Geophysical Research* 91.B (1986), pp. 10301–10308 (cit. on p. 16).

L Rowan, S Hook, M Abrams, and J Mars. “Mapping hydrothermally altered rocks at Cuprite, Nevada, using the advanced spaceborne thermal emission and reflection radiometer (ASTER), a new satellite-imaging system”. *Economic Geology* 98.5 (2003), pp. 1019–1027 (cit. on p. 1).

RSI. *ENVI 4.6 User’s Guide*. 4.6. Research Systems Inc. 2008 (cit. on pp. 105, 144, 152).

DE Rumelhart, G Hinton, and RJ Williams. “Learning Internal Representations by Error Propagation”. *Parallel Distributed Processing, Volume 1*. Ed. by JME D E Rumelhart. Cambridge, MA: MIT Press, 1986, pp. 318–362 (cit. on p. 21).

FG Sadowski and SJ Covington. *Processing and analysis of commercial satellite image data of the nuclear accident near Chernobyl USSR (US geological survey bulletin 1785)*. Tech. rep. 1785. USGS, 1988 (cit. on p. 1).
URL: http://www.tandfonline.com/doi/abs/10.1080/10106048809354173

J Salisbury, G Hunt, and C Lenhoff. “Visible and near-infrared spectra. X- Stony meteorites”. *Modern Geology* 5 (1975), pp. 115–126 (cit. on p. 12).

C Salvaggio, L Smith, and E Antoine. “Megacollect 2004: hyperspectral collection experiment of terrestrial targets and backgrounds of the RIT Megascene and surrounding area (Rochester, New York)". *Proceedings of SPIE* (2005), p. 555 (cit. on p. 144).

A Sato and K Yamada. "Generalized learning vector quantization". *Advances in Neural Information Processing Systems* (1996), pp. 423–429 (cit. on pp. 21–22).

J Schott. *Remote sensing: the image chain approach*. Oxford University Press, USA, 2007 (cit. on pp. 16–17).

J Schott, S Brown, R Raqueno, H Gross, and G Robinson. "Advanced synthetic image generation models and their application to multi/hyperspectral algorithm development". *Proceedings of SPIE* 3584 (1999), p. 211 (cit. on p. 144).

G Shaw and H Burke. "Spectral imaging for remote sensing". *MIT Lincoln Laboratory Journal* 14.1 (2003), pp. 3–28 (cit. on pp. 5, 26).

C Shen, J Kim, L Wang, and A van den Hengel. "Positive Semidefinite Metric Learning with Boosting". *Arxiv preprint arXiv:0910.2279* (2009) (cit. on p. 100).

N Singh-Miller, M Collins, and T Hazen. "Dimensionality reduction for speech recognition using neighborhood components analysis". *Eighth Annual Conference of the International Speech Communication Association* (2007) (cit. on p. 97).

S Slavney and S Murchie. *CRISM Spectral Library*. Tech. rep. JHU-APL, 2006 (cit. on p. 31).

M Sobhan. "Species Discrimination from a Hyperspectral Perspective". PhD thesis. Wageningen University, Wageningen, The Netherlands, 2007 (cit. on p. 59).

SL Sobolev. "On a theorem of functional analysis". *Transl. Amer. Math. Soc. 34 (2): 39–68* 34.2 (1963), pp. 39–68 (cit. on p. 60).

M Strickert. "Enhancing [M—G]RLVQ by quasi step discriminatory functions using 2nd order training". *MIWOCI 2011, Mittweidaer Workshop on Computational Intelligence*. Ed. by FM Schleif and T Villmann. University of Applied Sciences Mittweida, 2011, xx–yy (cit. on p. 162).

M Sugiyama. "Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis". *The Journal of Machine Learning Research* 8 (2007), p. 1061 (cit. on pp. 90, 93, 99, 120).

G Swayze. "The hydrothermal and structural history of the Cuprite Mining District, Southwestern Nevada: an integrated geological and geophysical approach". PhD thesis. University of Colorado, Boulder, 1997 (cit. on p. 15).

GA Swayze, RN Clark, A Goetz, N Gorelick, and T Chrien. "Spectral Identification of Surface Materials using Imaging Spectrometer Data: Evaluation of the Effects of Detector Sampling, Bandpass, and Signal to Noise Ratio using the USGS Tricorder Algorithm ". *Journal of Geophysical Research* (1999) (cit. on p. 3).

J Sweet. "The spectral similarity scale and its application to the classification of hyperspectral remote sensing data". *IEEE Trans. on Geoscience and Remote Sensing* (2004), pp. 92–99 (cit. on p. 34).

Y Tarabalka. "Classification of hyperspectral data using spectral-spatial approaches". PhD thesis. University of Iceland and Grenoble Institute of Technology, 2010 (cit. on p. 216).

Y Tarabalka, J Chanussot, and J Benediktsson. “Classification of Hyperspectral Images using Automatic Marker Selection and Minimum Spanning Forest”. *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (Sept. 2009), pp. 1–4 (cit. on p. 59).

Y Tarabalka, J Benediktsson, and J Chanussot. “Spectral-spatial Classification of Hyperspectral Imagery Based on Partitional Clustering Techniques”. *IEEE Trans. on Geoscience and Remote Sensing* 47.8 (2009), pp. 2973–2987 (cit. on p. 34).

DR Thompson, L Mandrake, MS Gilmore, and R Castaño. “Superpixel endmember detection”. *IEEE Trans. on Geoscience and Remote Sensing* 48.11 (Nov. 2010), pp. 4023–4033 (cit. on pp. 32, 105, 122, 124, 182).

IW Tsang, PM Cheung, and JT Kwok. “Kernel Relevant Component Analysis for Distance Metric Learning”. *Proceedings of International Joint Conference on Neural Networks* (May 2005), pp. 954–959 (cit. on pp. 28, 89–90).

SL Ustin, D DiPietro, K Olmstead, E Underwood, and G Scheer. “Hyperspectral remote sensing for invasive species detection and mapping”. *2002 IEEE International Geoscience and Remote Sensing Symposium, 2002. IGARSS'02* 3 (2002) (cit. on p. 4).

P van Otterloo and I Young. “A distribution-free geometric upper bound for the probability of error of a minimum distance classifier”. *Pattern Recognition* 10.4 (1978), pp. 281–286 (cit. on p. 21).

LJ van der Maaten. *An introduction to dimensionality reduction using matlab*. Tech. rep. MICC 07-07. Universiteit Maastricht, 2007 (cit. on pp. 98–100). URL: http://homepage.tudelft.nl/19j49/Matlab_Toolbox_for_Dimensionality_Reduction.html

FD van der Meer. “Analysis of spectral absorption features in hyperspectral imagery”. *International Journal of Applied Earth Observation and Geoinformation* 5.1 (Feb. 2004), pp. 55–68 (cit. on p. 34).

— “Spectral curve shape matching with a continuum removed CCSM algorithm”. *Remote Sensing* 21.16 (2000), pp. 3179–3185 (cit. on pp. 6, 27).

— “The effectiveness of spectral similarity measures for the analysis of hyperspectral imagery”. *Applied Earth Observations and Geoinformation* 8.1 (2006), pp. 3–17 (cit. on pp. 25, 34, 50).

G Vane and A Goetz. “Terrestrial imaging spectrometry: current status, future trends”. *Remote Sensing of Environment* 44.2 (1993), pp. 117–126 (cit. on p. 16).

V Vapnik and A Chervonenkis. “On the uniform convergence of relative frequencies of events to their probabilities”. *Theory of Probability and its Applications* 16.2 (1971), pp. 264–280 (cit. on p. 169).

J Venkateswaran, B Song, T Kahveci, and C Jermaine. “TRIAL: A Tool for Finding Distant Structural Similarities”. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 8.3 (2011), pp. 819–831 (cit. on pp. 201, 207).

T Villmann. “Sobolev metrics for learning of functional data-mathematical and theoretical aspects”. *Machine Learning Reports* 1 (2007), pp. 1–15 (cit. on p. 85).

T Villmann and B Hammer. "Functional Principal Component Learning Using Oja's Method and Sobolev Norms". *Advances in Self-Organizing Maps: 7th International Workshop on Self-Organizing Maps (WSOM 2009)* (2009) (cit. on p. 79).

T Villmann, E Merényi, and B Hammer. "Neural maps in remote sensing image analysis". *Neural Networks* 16.3-4 (2003), pp. 389–403 (cit. on pp. 4, 204).

U von Luxburg. "Statistical Learning with Similarity and Dissimilarity Functions". PhD thesis. Dissertation, Tech. Univ. Berlin, 2004 (cit. on p. 24).

KL Wagstaff and BJ Bornstein. "How much memory radiation protection do onboard machine learning algorithms require". *Proceedings of the IJCAI-09/SMC-IT-09/IWPSS-09 workshop on artificial intelligence in space* (2009) (cit. on p. 219).

C Wang and S Mahadevan. "A general framework for manifold alignment". *AAAI Fall Symposium on Manifold Learning and its Applications* (2009) (cit. on p. 188).

— "Manifold alignment using Procrustes analysis". *Proceedings of the 25th international conference on Machine learning* (2008), pp. 1120–1127 (cit. on pp. 188, 204).

KQ Weinberger and LK Saul. "Distance metric learning for large margin nearest neighbor classification". *The Journal of Machine Learning Research* 10 (2009), pp. 207–244 (cit. on p. 100).

KQ Weinberger, J Blitzer, and LK Saul. "Distance Metric Learning for Large Margin Nearest Neighbor Classification". *Advances in Neural Information Processing Systems* (2006) (cit. on pp. 6, 28, 89–91, 99, 120).

L Weizman and J Goldberger. "Classification of hyperspectral remote-sensing images using discriminative linear projections". *International Journal of Remote Sensing* 30.21 (2009), pp. 5605–5617 (cit. on pp. 29, 89–91).

F Wilcoxon. "Individual comparisons by ranking methods". *Biometrics Bulletin* (1945), pp. 80–83 (cit. on p. 39).

— "Probability tables for individual comparisons by ranking methods". *Biometrics* (1947), pp. 119–122 (cit. on p. 39).

E Xing, A Ng, M Jordan, and S Russell. "Distance Metric Learning with Application to Clustering with Side-Information". *Advances in Neural Information Processing Systems* (2003) (cit. on pp. 28–29, 89–90).

HL Yang and MM Crawford. "Exploiting spectral-spatial proximity for classification of hyperspectral data on manifolds". *Geoscience and Remote Sensing* (2012) (cit. on p. 216).

HL Yang and MM Crawford. "Manifold Alignment For Classification Of Multitemporal Hyperspectral Data". *Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (Apr. 2011), pp. 1–4 (cit. on pp. 140, 204).

L Yang and R Jin. *Distance metric learning: A comprehensive survey.* Tech. rep. Michigan State University, 2006 (cit. on pp. 90, 97).
URL: http://www.cs.cmu.edu/~liuy/frame_survey_v2.pdf

L Yang, R Jin, L Mummert, and R Sukthankar. "A boosting framework for visualitypreserving distance metric learning and its Application to Medical Image Retrieval". *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2010) (cit. on p. 100).

W Yang, K Wang, and W Zuo. "Fast neighborhood component analysis". *Neurocomputing* 83.C (Apr. 2012), pp. 31–37 (cit. on p. 98).

H Yu and H Yang. "A direct LDA algorithm for high-dimensional data - with application to face recognition". *Pattern Recognition* 34.10 (2001), pp. 2067–2070 (cit. on p. 96).

R Yuhas, A Goetz, and J Boardman. "Discrimination among semi-arid landscape endmembers using the Spectral Angle Mapper (SAM) algorithm". *Summaries of the 3rd Annual JPL Airborne Geoscience Workshop*. 1992, pp. 147–149 (cit. on p. 25).

Y Zhen and C Li. "Cross-domain knowledge transfer using semi-supervised classification". *AI 2008: Advances in Artificial Intelligence* (2008), pp. 362–371 (cit. on p. 139).

E Zhong, W Fan, Q Yang, O Verscheure, and J Ren. "Cross validation framework to choose amongst models and datasets for transfer learning". *Machine Learning and Knowledge Discovery in Databases* (2010), pp. 547–562 (cit. on p. 195).

J Zhou, J Chen, and J Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. 1.1. Arizona State University. 2011 (cit. on p. 166).
URL: http://www.public.asu.edu/~jye02/Software/MALSAR