SPECTRAL ANGLE-BASED FEATURE EXTRACTION AND SPARSE REPRESENTATION-BASED CLASSIFICATION OF HYPERSPECTRAL IMAGERY

A Dissertation

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

in Electrical and Computer Engineering

By

Minshan Cui

December 2015

SPECTRAL ANGLE-BASED FEATURE EXTRACTION AND SPARSE REPRESENTATION-BASED CLASSIFICATION OF HYPERSPECTRAL IMAGERY

Minshan Cui

Approved:

Chair of the Committee
Dr. Saurabh Prasad, Assistant Professor,
Dept. of Electrical and Computer Engineering

Committee Members:

Dr. Badri Roysam, Department Chair,
Dept. of Electrical and Computer Engineering

Dr. Jose L. Contreras-Vidal, Professor,
Dept. of Electrical and Computer Engineering

Dr. Demetrio Labate, Professor,
Dept. of Mathematics

Dr. Emanuel Papadakis, Professor,
Dept. of Mathematics

Dr. Suresh K. Khator, Associate Dean,
Cullen College of Engineering

Dr. Badri Roysam, Department Chair,
Dept. of Electrical and Computer Engineering

Acknowledgements

I am grateful to the following people who guided and supported me throughout my research.

I would first like to thank my advisor, Dr. Saurabh Prasad who guided me in choosing the right topic for my research. He also provided me valuable advice and support throughout my research. I would not have been able to successfully finish my Ph.D. degree without his support and guidance. I would also like to thank my committee members Dr. Badri Roysam, Dr. Demetrio Labate and Dr. Manos Papadakis for their support and valuable advices to improve this dissertation.

I want to thank all of my colleagues Xiong Zhou, Hao Wu, Yuhang Zhang, Jielian Guo, Tanu Priya, and Lifeng Yan for their help and advice during my studies at University of Houston.

I also thank my parents and grandmother who gave me the chance to study in the United States and always encouraged me with their best wishes.

Finally, I would like to thank my wife, Yehong Piao and my daughter, Sophie Xiaoxiang Cui. They were always there standing by me through the bad and good times.

SPECTRAL ANGLE-BASED FEATURE EXTRACTION AND SPARSE REPRESENTATION-BASED CLASSIFICATION OF HYPERSPECTRAL IMAGERY

An Abstract

of a

Dissertation

Presented to

the Faculty of the Department of Electrical and Computer Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

in Electrical and Computer Engineering

By

Minshan Cui

December 2015

Abstract

Remote sensing involves measuring and analyzing objects of interests through data collected by a remote imaging modality without physical contact with the objects. Hyperspectral sensors have become increasingly popular for a variety of remote sensing applications. Hyperspectral data are composed of densely sampled reflectance values over a wide range of the electromagnetic spectrum. Such a wealth of spectral information can provide unique spectral signatures of different materials present in a scene, which makes it especially suitable for classification tasks. In this dissertation, we present new dimensionality reduction (feature extraction) and classification algorithms for high-dimensional hyperspectral data. Specifically, we develop the theory and validate a new dimensionality reduction approach that maximizes angular separation in the lower dimensional subspace. We also propose and develop its "local" and "nonlinear kernel" variants for robust feature extraction of hyperspectral data. By preserving angular properties, the resulting subspaces demonstrate robustness to a variety of sources of variability that are commonly encountered in remote sensing applications. We also extend this approach to its "spatial variant" by incorporating spatial-contextual information along with spectral information from the hyperspectral images. We also optimize and develop a suitable sparse representation based classification framework for hyperspectral images. By extensive experiments on several real-world hyperspectral datasets, we demonstrate that the proposed algorithms significantly outperform the state-of-the-art methods. Further, we also demonstrate the applicability of the proposed methods for a practical environmental remote sensing task.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Remote sensing and its applications

Remote sensing involves measuring and analyzing the characteristics of objects of interests through the data collected by sensors on a remote platform, without physical contact with the objects. A majority of the modern remote sensing systems measure energy in the 400 nm to 2500 nm region of the electromagnetic spectrum, although there are some sensors that operate at larger wavelengths. Electromagnetic energy reaching the target suffers scattering and absorption. Scattering happens when there are interactions between dust particles and electromagnetic radiation, which consequently reduce the intensity of radiation. The two most common forms of scattering are Rayleigh scattering and Mie scattering. Absorption of electromagnetic radiation is due to absorptions in energy from specific molecules (e.g., gases or water) in the atmosphere. The three main sources of absorptions are water vapor, carbon dioxide and ozone. After radiation reaches the target on the Earth, some portion of the radiation is absorbed by the target, some portion of the radiation passes through the target, and the remainder is reflected from the target. In remote sensing applications, we measure the radiation that is reflected from the target.

Recent developments in remote sensing technologies have led the way for the development of airborne and spaceborne remote sensors. These sensors are usually mounted on aircrafts and satellites to capture wide regions of interests. Remote sensing technologies

have shifted from multispectral images (with a few selective bands) to hyperspectral images (with hundreds or even thousands of bands). Hyperspectral data are composed of densely sampled reflectance values over a wide range of the electromagnetic spectrum. Such a wealth of spectral information in hyperspectral data can provide unique spectral signatures of different materials present in a scene, which makes it especially suitable for classification tasks. The wide availability of remotely sensed data, particularly hyperspectral data, is enabling advancements for various applications such as environmental monitoring, agriculture and mineral exploration.

1.2 Recent advances in techniques and limitations for hyperspectral image analysis

Hyperspectral imagery consists of hundreds or thousands of densely sampled spectral bands. The resulting spectral information can provide unique spectral signatures of different materials present in a scene, which makes hyperspectral imagery especially suitable for classification problems. Due to its high-dimensionality, we often require large quantities of training data to accurately model the data. To fully exploit the potentially discriminative information in remotely sensed data, there is a need to develop dimensionality reduction techniques, resulting in a lower dimensional "optimal" feature space. Dimensionality reduction is hence among the most important research areas for remote sensing image analysis, wherein the high-dimensional data is projected into a lower dimensional subspace subject to optimization of certain metrics or objective functions. For example, in compression tasks, an effective dimensionality reduction method should preserve the original data in the reduced subspace. For classification, a good dimensionality reduction method should find a

subspace where the class specific discriminant information is captured.

Popular linear dimensionality reduction methods include principal component analysis (PCA), linear discriminant analysis (LDA) and their many variants [1–3]. PCA and LDA are not designed to exploit potentially non-linear separability of data (e.g., data on manifolds). Several manifold learning methods, including *local* linear embedding [4], ISOMAP [5], laplacian eigenmap [6], locality preserving projection (LPP) [7], *local* Fisher discriminant analysis (LFDA) [8] etc. have been proposed in literature. These methods can effectively preserve the *local* (neighborhood) structure of data in the resulting embeddings by utilizing information about nearest neighbors of every point on a manifold. It has been shown in [9] that hyperspectral data is inherently a low-dimensional manifold embedded on a high-dimensional space. Recent work [10–13] has also demonstrated that learning and utilizing manifold-specific properties is beneficial for hyperspectral image classification.

A majority of the dimensionality reduction methods described above commonly employ Euclidean distance information. However, advantages of using angular information for hyperspectral image analysis have been demonstrated previously [14–16]. A key advantage of using angular distances (commonly known as spectral angles when used with hyperspectral imagery, wherein feature vectors correspond to spectral reflectance) stems from the fact that such a measure is sensitive to the shape of spectral signatures, while simultaneously being relatively invariant to changes in atmospheric, illumination and topographic conditions. It is well known that spectral reflectance shapes of samples from the same material often exhibit linear scaling due to various sources of variability, and that angular distances are more sensitive to shapes of spectral reflectance profiles than Euclidean distances [15, 17].

Realizing the potential relevance of angular information for hyperspectral classification

problems, a few feature extraction methods exploring the angular (correlation) relationships between data samples have been previously developed. In [18], canonical correlation analysis (CCA) is proposed to find two separate projections, where the correlations between two sets of multi-dimensional variables onto those projections are maximized. Since the projections found by CCA are class specific and not global, they can not be directly used for real-world classification problems, in which class label information of test samples is not available a-priori. Discriminant analysis of canonical correlation is presented in [19] for image sets classification. In [20], correlation discriminant analysis (CDA) has been proposed to find a transformation where between-class correlation is minimized while within-class correlation is maximized simultaneously. Different from CCA, the transformation found by CDA is *global* which is suitable for classification problems. However, CDA cannot reduce the dimensionality of the data and the optimization problem in CDA does not have a closed form expression, and is solved via a gradient ascent which is computationally very expensive and subject to related limitations.

Another important area of hyperspectral data analysis is classification. In literature, various classification techniques including parametric and non-parametric classifiers have been proposed in the literature to efficiently exploit discriminant information contained in these rich spectral channels. The K-nearest neighbor (KNN) classifier [13, 21] is one of the simplest classifier for HSI classification. It is a nonparametric classifier where a test sample is simply assigned to a class whose samples occur most commonly among its K nearest neighbors. Parametric classifiers such as Maximum Likelihood (ML) [22] and Gaussian mixture model (GMM) classifiers [12, 23] are also common — these are based on statistical models learned from training samples. Due to the high dimensional nature of

HSI, either a large number of training samples are required to estimate the statistics of the data or dimensionality reduction needs to be performed to ensure reliable performance of such parametric classifiers. Among the various popular classifiers, support vector machines (SVMs) [24, 25] have been shown to be robust for classifying high-dimensional HSI.

Recently, sparse representations of signals has received great attention. Signals can potentially have a compact representation in terms of linear combination of atoms in the dictionary. Based on the theory of sparse representation, Wright et al. proposed a sparse representation based classification (SRC) [26] for robust face recognition. It relies on the underlying assumption that a test sample can be linearly represented by a small number of training samples from the same class. Experiments conducted in [26] have demonstrated the efficacy of SRC under a variety of scenarios for face recognition problems. Later, Zhang et al. proposed a collaborative representation classifier (CRC) [27] which seeks to linearly represent the test sample using all of the training samples in a least-squares sense. It is demonstrated in [27] that with much less computational complexity, CRC can have a competitive classification performance compared with SRC. Due to the wide range of applications, SRC has also been actively adopted to analyze HSI. Chen et al. proposed a joint sparsity model (JSM) for HSI classification [28, 29] by exploiting the contextual information of test samples via rectangular analysis windows. A nearest regularized subspace (NRS) classifier which couples nearest-subspace classification with a distance-weighted Tikhonov regularization has been proposed in [30]. Also, in [31], the authors exploit the sparsity of HSI via a graphical model to perform the classification.

However, for the problem of hyperspectral data classification, pixels from different classes are often characterized by a relatively high correlation with each other, which makes SRC

challenging. As is known, all atoms in the dictionary need to be ℓ_2 normalized to avoid the bias caused by atoms with varying lengths in SRC. However, by performing ℓ_2 normalization, samples having high correlation with each other will be highly overlapped with each other, implying that the Euclidean distance information is lost or distorted. Another limitation of traditional SRC comes from the fact that it does not incorporate the class label (prior) information of the dataset when learning the representation. It only utilizes the class label information in post processing when calculating the residuals for each class, while ignoring it when calculating the representation coefficients. Due to the high correlation between samples in HSI, using the entire training dataset as the dictionary for SRC results in atoms potentially being selected from multiple classes. This contradicts the core assumption of SRC, that the support of a test sample should ideally be in a union of atoms from the same class as the test sample.

1.3 Dissertation contributions

In this dissertation, we present new dimensionality reduction methods for the problem of hyperspectral data classification. Specifically, we propose angular discriminant analysis (ADA) which finds a projection, where the angular separation of between-class samples is maximized and the within-class samples is minimized simultaneously in a low dimensional subspace. For data where class specific samples are not clustered into well-defined unimodal clusters on a unit hypersphere, projections based on ADA may not be able to capture the multi-modality structure in the resulting subspace. For such data, we propose *local* angular discriminant analysis (LADA), an approach which preserves the locality of data in the projected subspace through an affinity matrix, while simultaneously angularly separating

between class samples. With such a projection, it is expected that the classification performance of classifiers such as SRC and NN with cosine angle distance is improved. When samples from different classes are in the same direction or are angularly non-separable in the original space, both ADA and LADA will fail to find a subspace that can angularly separate between-class samples. We contend that formulating ADA and LADA in a reproducible kernel Hilbert space (RKHS) will overcome this limitation. We show that ADA and LADA can be easily extended to their *kernelized* variants, kernel angular discriminant analysis (KADA) and kernel *local* angular discriminant analysis (KLADA) by invoking the *kernel trick*.

We also employ the proposed dimensionality reduction methods to address the real-world application problems of remote sensing. Particularly, we address the classification problem when some part of the collected hyperspectral images are under shadows due to clouds or some other nearby objects. Due to the illumination-insensitive of the proposed dimensionality reduction methods, the projected hyperspectral data based on these methods exhibit robustness to illumination changes which leads to the higher classification accuracies.

In remote sensing data classification, the process of collecting labeled training data is very time consuming and expensive. We can however, easily obtain unlabeled samples without much effort. To be able to use this advantage, we propose an unsupervised variant of ADA which we call local similarity preserving projection (LSPP). Unlike ADA which requires the labeled training samples, LSPP does not require labeled training samples to learn the projection matrix. It is also well-known that utilizing the spatial information in hyperspectral data can dramatically improve the classification accuracies. This is based on the observation that neighboring pixels in hyperspectral images usually consist of the

same type of materials. It means they are usually from the same class and have similar spectral characteristics. To incorporate the spatial information of hyperspectral data, we extend LSPP into its spatial version of LSPP (SLSPP) which effectively use the contextual information around each pixel in hyperspectral images when learning the projection.

To improve the classification performance of SRC, we propose class-dependent sparse representation classifier (cdSRC) to effectively exploit the correlation and Euclidean distance information simultaneously. Different from traditional SRC, the proposed cdSRC is comprised of two components — class-dependent OMP (cdOMP) and class-dependent KNN (cdKNN) which perform the OMP and KNN in a classwise manner by incorporating the prior (class label) information. Specifically, in cdOMP, the residual for the i-th class is the norm of the difference between the test sample and an approximated test sample derived through OMP using the dictionary formed by training samples from the i-th class. In cdKNN, the i-th class distance is measured by the mean of Euclidean distances of the test sample and its K nearest neighbors. After calculating the residual and distance via cdOMP and cdKNN, a test sample is assigned a class label via a unified class membership function.

To further enhance the classification performance of cdSRC, we present cdOLS which is a class-dependent version of OLS. It is based on the observation that the recovery ability of OLS is generally better than OMP in terms of the least square error estimation under the same experimental setting (i.e., the same sparsity level). Therefore, it is expected that the classification performance of cdOMP can be further improved by replacing OMP with OLS. We also extend both cdSRC and cdOLS into a kernel cdSRC (KcdSRC) and kernel cdOLS (KcdOLS) to effectively deal with non-linearly separable data. By extensive experiments

on several real-world hyperspectral datasets, we demonstrate that the proposed subspace learning and classification algorithms can significantly increase the classification accuracies compared with state-of-art results.

Lastly, we effectively combine the simultaneous orthogonal matching pursuit (SOMP) and block orthogonal matching pursuit (BOMP) and propose simultaneous block orthogonal matching pursuit (SBOMP) to explore the block structure of test samples and training samples respectively. A classification technique based on the SBOMP is proposed. SBOMP can be employed after SLSPP to fully utilize the spatial contextual information of HSI for classification.

Chapter 2

Angle-Based Dimensionality Reduction

2.1 Introduction

In this work, we propose a dimensionality reduction method named angular discriminant analysis (ADA). ADA finds a projection, where the angular separation of between-class samples is maximized and the within-class samples is minimized simultaneously in a low dimensional subspace. We also propose a *local* angular discriminant analysis (LADA), which preserves the locality of data in the projected space through an affinity matrix, while angularly separating different class samples. The ADA and LADA are mainly used to improve the classification performance of NN classifier with cosine angle distance and SRC, in which the sparse representation coefficient is learned via orthogonal matching pursuit (OMP) [32], by learning an appropriate, lower dimensional subspace. With such a projection, it is expected that the classification performance of NN with cosine angle distance is improved. It can also enhance the accuracy of the coefficients recovered by OMP, which in turn results in a better classification performance of SRC. This is due to the fact that OMP selects an atom (training sample) from the dictionary that produces the largest normalized inner product with the residual of a signal (test sample) at each iteration, stopping before the number

of selected atoms becomes larger than the predefined sparsity level or the residual is lower than some predefined value. We note that ADA and its variants can also be used as pre-processing to emerging approaches such as subspace-based learning [33], wherein subspaces are utilized as basic elements for classification. Preliminary work with ADA and LADA was presented by us in [34, 35]. We also show that ADA and LADA can be easily extended to their *kernelized* variants, kernel angular discriminant analysis (KADA) and kernel *local* angular discriminant analysis (KLADA) by invoking the *kernel trick*.

In typical remote sensing data classification applications, the process of collecting labeled training data is often very time consuming and expensive. Unlabeled data on the other hand is easily available and hence unsupervised dimensionality reduction methods that effectively learn the most appropriate subspace can hence be readily utilized. With that in mind, we propose an unsupervised counterpart of our recently proposed supervised subspace algorithm, the angular discriminant Analysis (ADA). This unsupervised counterpart is referred to as local similarity preserving projection (LSPP) in this paper. Unlike ADA which requires labeled training samples, LSPP does not require labeled training samples to learn the projection matrix. Additionally, it is well-known that utilizing spatial information in hyperspectral data can dramatically improve the classification accuracies because any such method accounts for the spatial variability of spectral content in local spatial neighborhoods. This follows from the observation that spatially neighboring pixels are highly likely to belong to the same class and have similar spectral characteristics. To incorporate such spatial information of hyperspectral data into our unsupervised projection, we develop a spatial information driven variant of LSPP (SLSPP) which effectively uses the spatial contextual information around each pixel in hyperspectral images to learn

the *optimal projection*.

2.2 Related work

In this section, we will briefly describe some popular linear subspace learning methods such as linear discriminant analysis (LDA), local Fisher discriminant analysis (LFDA), and correlation discriminant analysis (CDA).

Let $\{\boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \{1, 2, \ldots, c\}, i = 1, 2, \ldots, n\}$ be the d-dimensional i-th training sample with an associated class label y_i, where c is the number of classes and n is the total number of training samples. $n = \sum_{l=1}^{c} n_l$ where n_l denotes the number of training samples from class l. Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$ denote the training data matrix and $\boldsymbol{T} \in \mathbb{R}^{d \times r}$ be the projection matrix, where r denotes the reduced dimensionality. We also denote symbols having ℓ_2 norm (unit norm) with a *tilde* and those corresponding to an optimal value of an objective function or the value in the projected space with *hat*. In the context of pattern recognition, our goal is to predict a label for a test sample $\boldsymbol{x}_{test} \in \mathbb{R}^d$.

2.2.1 Linear discriminant analysis

The within-class scatter matrix $\boldsymbol{S}^{(\mathrm{w})}$ and between-class scatter matrix $\boldsymbol{S}^{(\mathrm{b})}$ in LDA take the form

$$\boldsymbol{S}^{(\mathrm{w})} = \sum_{l=1}^{c} \sum_{i:y_i=l} (\boldsymbol{x}_i - \boldsymbol{\mu}_l)(\boldsymbol{x}_i - \boldsymbol{\mu}_l)^t \text{ and} \tag{2.1}$$

$$\boldsymbol{S}^{(\mathrm{b})} = \sum_{l=1}^{c} n_l (\boldsymbol{\mu}_l - \boldsymbol{\mu})(\boldsymbol{\mu}_l - \boldsymbol{\mu})^t, \tag{2.2}$$

where $\boldsymbol{\mu}_l = \frac{1}{n_l} \sum_{i:y_i=l} \boldsymbol{x}_i$ is l-th class sample mean and $\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i$ is the total mean.

The projection matrix of LDA is defined as the solution that maximizes the Fisher's

ratio between and within-class scatter matrices, and is determined to be

$$\boldsymbol{T}_{LDA} = \underset{\boldsymbol{T}\in\mathbb{R}^{d\times r}}{\operatorname{argmax}} \left[\operatorname{tr}\left(\left(\boldsymbol{T}^t \boldsymbol{S}^{(\mathrm{w})} \boldsymbol{T} \right)^{-1} \boldsymbol{T}^t \boldsymbol{S}^{(\mathrm{b})} \boldsymbol{T} \right) \right]. \tag{2.3}$$

2.2.2 Local Fisher discriminant analysis

It is shown in [8] that LDA can not well separate samples when they form several clusters in a class. LFDA is proposed in [8] to address this problem by preserving the multi-modal structure of class-conditional distributions in the projected subspace. It effectively combines the properties of LDA and LPP. LPP is an unsupervised dimensionality reduction method that is used to preserve the *local* structure of neighboring samples in a lower-dimensional projected subspace. LFDA finds an optimal subspace where between-class samples are well-separated, while simultaneously the *local* neighborhood structure of within-class samples is preserved.

In LFDA, the *local* within-class $\boldsymbol{S}^{(\mathrm{lw})}$ and between-class $\boldsymbol{S}^{(\mathrm{lb})}$ scatter matrices are defined as

$$\boldsymbol{S}^{(\mathrm{lw})} = \frac{1}{2} \sum_{i,j=1}^{n} W_{ij}^{(\mathrm{lw})} (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^t \text{ and} \tag{2.4}$$

$$\boldsymbol{S}^{(\mathrm{lb})} = \frac{1}{2} \sum_{i,j=1}^{n} W_{ij}^{(\mathrm{lb})} (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^t. \tag{2.5}$$

The $\boldsymbol{W}^{(\mathrm{lw})}$ and $\boldsymbol{W}^{(\mathrm{lb})}$ are $n \times n$ weight matrices defined as

$$W_{ij}^{(\mathrm{lw})} = \begin{cases} A_{ij}/n_l, & \text{if } y_i, y_j = l, \\ 0, & \text{if } y_i \neq y_j \text{ and} \end{cases} \tag{2.6}$$

$$W_{ij}^{(\mathrm{lb})} = \begin{cases} A_{ij}(1/n - 1/n_l), & \text{if } y_i, y_j = l, \\ 1/n, & \text{if } y_i \neq y_j. \end{cases} \tag{2.7}$$

The affinity matrix $A_{ij} \in [0, 1]$ between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is defined as

$$A_{ij} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\gamma_i \gamma_j}\right), \tag{2.8}$$

where $\gamma_i = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{(\text{knn})}\|$ denotes the *local* scaling of data samples in the neighborhood of $\boldsymbol{x}_i$, and $\boldsymbol{x}_i^{(\text{knn})}$ is the K-th nearest neighbors of $\boldsymbol{x}_i$. $\boldsymbol{A}$ is a symmetric affinity matrix that measures the distance between samples. Although other affinity matrices can be used, the heat kernel as defined in (2.48) has been shown to have very effective locality-preserving properties. If $A_{ij} = 1$ for all i and j, LFDA degenerates to traditional LDA. The projection matrix of LFDA can obtained via

$$\boldsymbol{T}_{LFDA} = \underset{\boldsymbol{T} \in \mathbb{R}^{d \times r}}{\operatorname{argmax}} \left[\operatorname{tr}\left(\left(\boldsymbol{T}^t \boldsymbol{S}^{(\text{lw})} \boldsymbol{T}\right)^{-1} \boldsymbol{T}^t \boldsymbol{S}^{(\text{lb})} \boldsymbol{T}\right)\right]. \tag{2.9}$$

2.2.3 Correlation discriminant analysis

The CDA [20] was recently proposed as a discriminant analysis approach based on correlation similarity. It seeks a transformation where the difference between the within-class and between-class samples correlation are maximized.

Let $\hat{C}^{(\text{w})}$ and $\hat{C}^{(\text{b})}$ be the within-class and between-class correlation in the CDA transformed space, which are defined as

$$\begin{aligned}
\hat{C}^{(\text{w})} &= \frac{1}{n^{(\text{w})}} \sum_{l=1}^{c} \sum_{i,j:y_i,y_j=l} \frac{(\boldsymbol{T}^t \boldsymbol{x}_i)^t (\boldsymbol{T}^t \boldsymbol{x}_j)}{\|\boldsymbol{T}^t \boldsymbol{x}_i\| \|\boldsymbol{T}^t \boldsymbol{x}_j\|} \\
&= \frac{1}{n^{(\text{w})}} \sum_{l=1}^{c} \sum_{i,j:y_i,y_j=l} \frac{\boldsymbol{x}_i^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_j}{\sqrt{\boldsymbol{x}_i^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_i \boldsymbol{x}_j^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_j}} \quad \text{and} \qquad (2.10) \\
\hat{C}^{(\text{b})} &= \frac{1}{n^{(\text{b})}} \sum_{l=1}^{c} \sum_{i,j:y_i \neq y_j} \frac{(\boldsymbol{T}^t \boldsymbol{x}_i)^t (\boldsymbol{T}^t \boldsymbol{x}_j)}{\|\boldsymbol{T}^t \boldsymbol{x}_i\| \|\boldsymbol{T}^t \boldsymbol{x}_j\|} \\
&= \frac{1}{n^{(\text{b})}} \sum_{l=1}^{c} \sum_{i,j:y_i \neq y_j} \frac{\boldsymbol{x}_i^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_j}{\sqrt{\boldsymbol{x}_i^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_i \boldsymbol{x}_j^t \boldsymbol{T} \boldsymbol{T}^t \boldsymbol{x}_j}}, \qquad (2.11)
\end{aligned}$$

where $n^{(\mathrm{w})}$ and $n^{(\mathrm{b})}$ denote the number of sample pairs from within-class and between-class respectively. Let $\boldsymbol{M} = \boldsymbol{T}\boldsymbol{T}^t$, then the optimization problem of CDA is defined as one that maximizes the difference between within-class and between-class correlation matrices, described below.

$$\boldsymbol{M}_{CDA} = \underset{\boldsymbol{M}\in\mathbb{R}^{d\times d}}{\operatorname{argmax}} \left[\hat{C}^{(\mathrm{w})} - \hat{C}^{(\mathrm{b})}\right], \quad \text{s. t.} \quad \boldsymbol{M} \geq 0. \tag{2.12}$$

The optimization problem in (2.12) is solved using a gradient ascent approach followed by an iterative projection method.

2.3 Proposed angle-based dimensionality reduction

2.3.1 Angular discriminant analysis

We propose ADA, which can be considered as an angular variant of LDA, utilizing angular separation as opposed to Euclidean distance separation. Similar to LDA, ADA projects samples into a lower dimensional subspace, where the angular separation of between-class samples is maximized, while the within-class samples is minimized. The resulting formulations make it a uniquely beneficial pre-processing to classifiers utilizing the angular relationships of samples such as NN with cosine angle distance, SRC with OMP as the recovery method etc. In the following, we describe the proposed ADA in detail.

Let $\hat{I}^{(\mathrm{w})}$ and $\hat{I}^{(\mathrm{b})}$ be the within-class and between-class normalized inner product in the ADA projected subspace, which are defined as

$$\hat{I}^{(\mathrm{w})} = \sum_{l=1}^{c} \sum_{i:y_i=l} (\boldsymbol{T}^t\tilde{\boldsymbol{x}}_i)^t(\boldsymbol{T}^t\tilde{\boldsymbol{\mu}}_l) \text{ and} \tag{2.13}$$

$$\hat{I}^{(\mathrm{b})} = \sum_{l=1}^{c} n_l(\boldsymbol{T}^t\tilde{\boldsymbol{\mu}}_l)^t(\boldsymbol{T}^t\tilde{\boldsymbol{\mu}}), \tag{2.14}$$

where $\tilde{\boldsymbol{\mu}}_l = \frac{1}{n_l}\sum_{i:y_i=l}\tilde{\boldsymbol{x}}_i$ is the normalized mean of l-th class samples, and $\tilde{\boldsymbol{\mu}} = \frac{1}{n}\sum_{i=1}^{n}\tilde{\boldsymbol{x}}_i$

is defined as a normalized total mean.

Based on the properties of trace operator tr, $\hat{I}^{(\mathrm{w})}$ and $\hat{I}^{(\mathrm{b})}$ can be converted into

$$\begin{aligned}
\hat{I}^{(\mathrm{w})} &= \mathrm{tr}\left(\hat{I}^{(\mathrm{w})}\right) \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} \mathrm{tr}\left((\boldsymbol{T}^t \tilde{\boldsymbol{x}}_i)^t (\boldsymbol{T}^t \tilde{\boldsymbol{\mu}}_l)\right) \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} \mathrm{tr}\left(\boldsymbol{T}^t \tilde{\boldsymbol{\mu}}_l (\boldsymbol{T}^t \tilde{\boldsymbol{x}}_i)^t\right) \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} \mathrm{tr}\left(\boldsymbol{T}^t \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{x}}_i^t \boldsymbol{T}\right) \\
&= \mathrm{tr}\left(\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{w})} \boldsymbol{T}\right) \text{ and} \qquad (2.15) \\
\hat{I}^{(\mathrm{b})} &= \mathrm{tr}\left(\hat{I}^{(\mathrm{b})}\right) \\
&= \sum_{l=1}^{c} \mathrm{tr}\left(n_l (\boldsymbol{T}^t \tilde{\boldsymbol{\mu}}_l)^t (\boldsymbol{T}^t \tilde{\boldsymbol{\mu}})\right) \\
&= \sum_{l=1}^{c} \mathrm{tr}\left(n_l \boldsymbol{T}^t \tilde{\boldsymbol{\mu}} (\boldsymbol{T}^t \tilde{\boldsymbol{\mu}}_l)^t\right) \\
&= \sum_{l=1}^{c} \mathrm{tr}\left(\boldsymbol{T}^t n_l \tilde{\boldsymbol{\mu}} \tilde{\boldsymbol{\mu}}_l^t \boldsymbol{T}\right) \\
&= \mathrm{tr}\left(\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{b})} \boldsymbol{T}\right), \qquad (2.16)
\end{aligned}$$

where $\boldsymbol{O}^{(\mathrm{w})}$ and $\boldsymbol{O}^{(\mathrm{b})}$ are the within-class and between-class matrices obtained by outer product of samples in the original (input) space, defined as

$$\boldsymbol{O}^{(\mathrm{w})} = \sum_{l=1}^{c} \sum_{i:y_i=l} \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{x}}_i^t \text{ and} \qquad (2.17)$$

$$\boldsymbol{O}^{(\mathrm{b})} = \sum_{l=1}^{c} n_l \tilde{\boldsymbol{\mu}} \tilde{\boldsymbol{\mu}}_l^t. \qquad (2.18)$$

The projection matrix $\boldsymbol{T}_{ADA}$ of ADA can be obtained by solving the following trace

ratio problem

$$\boldsymbol{T}_{ADA} = \underset{\boldsymbol{T}\in\mathbb{R}^{d\times r}}{\operatorname{argmin}} \left[\frac{\operatorname{tr}\left(\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{b})} \boldsymbol{T}\right)}{\operatorname{tr}\left(\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{w})} \boldsymbol{T}\right)} \right]. \tag{2.19}$$

Although there is no closed form solution for this trace-ratio problem, an approximate solution can be easily obtained by solving the corresponding ratio-trace problem [36, 37], which is defined as

$$\boldsymbol{T}_{ADA} \approx \underset{\boldsymbol{T}\in\mathbb{R}^{d\times r}}{\operatorname{argmin}} \left[\operatorname{tr}\left((\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{w})} \boldsymbol{T})^{-1} \boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{b})} \boldsymbol{T} \right) \right]. \tag{2.20}$$

The projection matrix $\boldsymbol{T}_{ADA}$ in (2.20) can be obtained by solving the generalized eigenvalue problem involving $\boldsymbol{O}^{(\mathrm{w})}$ and $\boldsymbol{O}^{(\mathrm{b})}$. As can be seen from (2.18), the rank of $\boldsymbol{O}^{(\mathrm{b})}$ is at most $c-1$. It implies that only $c-1$ meaningful principal directions can be obtained through ADA.

Although ADA and CDA are similar in that they exploit the angular and correlation information of samples respectively, they differ in several ways — (1) the objective function of ADA minimizes the ratio of between-class to within-class normalized inner products of samples in the projected space, while CDA is based on their cosine angle difference; (2) CDA inherently is not a dimensionality reduction algorithm, since it primarily searches for a transformation to a space of the same dimensionality where angular separation is enhanced; (3) The optimization problem of ADA can be converted into a simple generalized eigenvalue problem, while the optimization problem in CDA is solved based on an iterative gradient-based optimization method which can potentially be trapped in a locally optimal solution. Furthermore, there are several parameters that need to be tuned in an iterative gradient based method, such as the initial random projection matrix, gradient step size etc., and the computational complexity of CDA is much higher than ADA. Finally, the proposed

formulation can be easily extended to a *localized* variant, which will be developed later in this paper.

Proposition 1 *Let $\hat{S}^{(w)}$ and $\hat{S}^{(b)}$ be the within-class and between-class scatter matrices of LDA for the projected samples. They can be reformulated as*

$$\hat{S}^{(w)} = \sum_{l=1}^{c} \sum_{i:y_i=l} (\| \boldsymbol{T}^t \boldsymbol{x}_i\|^2 + \| \boldsymbol{T}^t \boldsymbol{\mu}_l\|^2) - 2\,\mathrm{tr}(\boldsymbol{T}^t \bar{\boldsymbol{O}}^{(w)} \boldsymbol{T}) \text{ and} \tag{2.21}$$

$$\hat{S}^{(b)} = \sum_{l=1}^{c} n_l (\| \boldsymbol{T}^t \boldsymbol{\mu}_l\|^2 + \| \boldsymbol{T}^t \boldsymbol{\mu}\|^2) - 2\,\mathrm{tr}(\boldsymbol{T}^t \bar{\boldsymbol{O}}^{(b)} \boldsymbol{T}), \tag{2.22}$$

where $\bar{\boldsymbol{O}}^{(w)}$ and $\bar{\boldsymbol{O}}^{(b)}$ are defined as

$$\bar{\boldsymbol{O}}^{(w)} = \sum_{l=1}^{c} \sum_{i:y_i=l} \|\boldsymbol{x}_i\| \|\boldsymbol{\mu}_l\| \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{\mu}}_l^t \text{ and} \tag{2.23}$$

$$\bar{\boldsymbol{O}}^{(b)} = \sum_{l=1}^{c} n_l \|\boldsymbol{\mu}_l\| \|\boldsymbol{\mu}\| \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{\mu}}^t. \tag{2.24}$$

The proof of Proposition 1 is provided in Appendix A.1. Based on Proposition 1, it is observed that the scatter matrices utilized in the LDA formulation can be rewritten as having two key additive components — a Euclidean distance based terms (that utilizes the norm of samples), and a term that quantifies angular separation (similar to that used in ADA). ADA would hence be more favorable compared to LDA for datasets (e.g., hyperspectral imagery) where source of variability manifest themselves as changes in energy/norm of the samples.

Fig. 2.1 demonstrates ADA and LDA projections using a three-dimensional three-class synthetic dataset that is generated from a unit variance Gaussian distribution. It can be seen that the two-dimensional subspace found by ADA indeed yields much better angular separation than LDA.

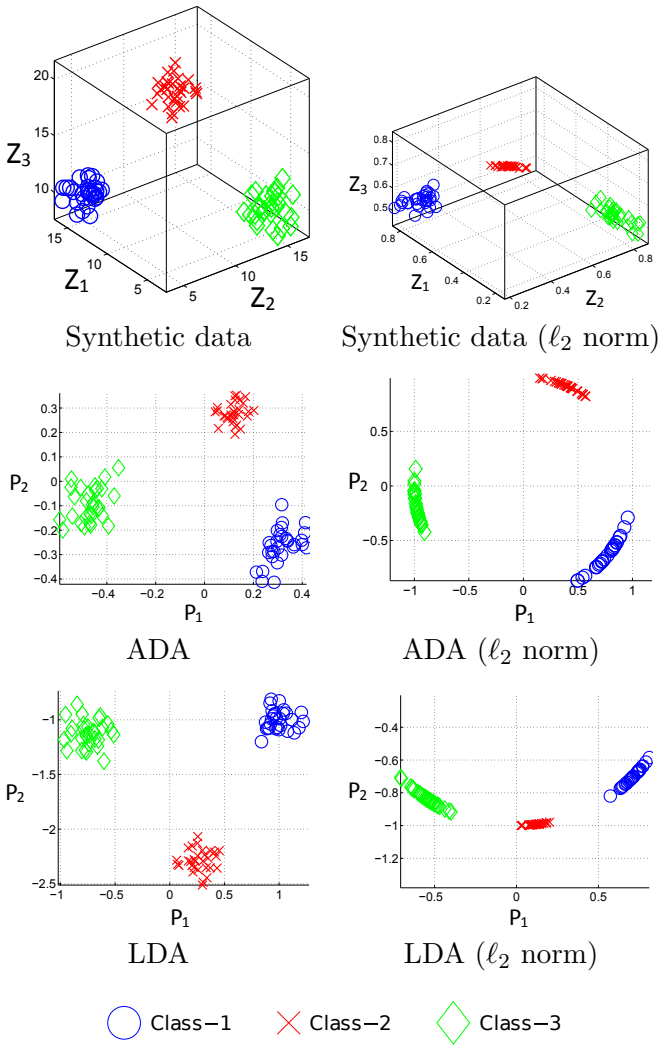


Figure 2.1: Evaluation of ADA and LDA using a three-dimensional three-class synthetic data generated from a unit variance Gaussian distribution.

2.3.2 Local angular discriminant analysis

For data where class specific samples are not clustered into well-defined unimodal clusters on a unit hypersphere, projections based on ADA may not be able to capture the multi-modality structure in the resulting subspace. For such data, we propose LADA — an approach which preserves the locality of data in the projected subspace through an affinity matrix, while simultaneously angularly separating between-class samples.

For the *local* variant of ADA, we modify the normalized outer-product matrices in ADA with a locality preserving constraint. Before defining these *local* within and between-class outer-product matrices, we provide the derivations of within and between-class outer product matrices of ADA in a pairwise manner.

$$\begin{aligned}
\boldsymbol{O}^{(\mathrm{w})} &= \sum_{l=1}^{c} \sum_{i:y_i=l} \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{x}}_i^t \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} (\frac{1}{n_l} \sum_{j:y_j=l} \tilde{\boldsymbol{x}}_j) \tilde{\boldsymbol{x}}_i^t \\
&= \sum_{l=1}^{c} \frac{1}{n_l} \sum_{i,j:y_i,y_j=l} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \\
&= \sum_{i,j=1}^{n} W_{ij}^{(\mathrm{w})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t, \qquad (2.25)
\end{aligned}$$

where

$$W_{ij}^{(\mathrm{w})} = \begin{cases} 1/n_l, & \text{if } y_i, y_j = l, \\ 0, & \text{if } y_i \neq y_j. \end{cases} \qquad (2.26)$$

Let us define the total outer product matrix as

$$\begin{aligned}\boldsymbol{O}^{(\mathrm{t})} &= \sum_{i=1}^{n} \tilde{\boldsymbol{\mu}} \tilde{\boldsymbol{x}}_i^t \\ &= \sum_{i=1}^{n} (\frac{1}{n} \sum_{j=1}^{n} \tilde{\boldsymbol{x}}_j) \tilde{\boldsymbol{x}}_i^t \\ &= \frac{1}{n} \sum_{i,j=1}^{n} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t, \end{aligned} \tag{2.27}$$

which consequently yields between-class outer product matrix

$$\begin{aligned}\boldsymbol{O}^{(\mathrm{b})} &= \boldsymbol{O}^{(\mathrm{t})} - \boldsymbol{O}^{(\mathrm{w})} \\ &= \sum_{i,j=1}^{n} (\frac{1}{n} - W_{ij}^{(\mathrm{w})}) \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \\ &= \sum_{i,j=1}^{n} W_{ij}^{(\mathrm{b})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t, \end{aligned} \tag{2.28}$$

where

$$W_{ij}^{(\mathrm{b})} = \begin{cases} 1/n - 1/n_l, & \text{if } y_i, y_j = l, \\ 1/n, & \text{if } y_i \neq y_j. \end{cases} \tag{2.29}$$

After reformulating ADA into a pairwise manner, the within and between-class outer product matrices of LADA are obtained by multiplying the normalized weight matrices

$$\boldsymbol{O}^{(\mathrm{lw})} = \sum_{i,j=1}^{n} \tilde{W}_{ij}^{(\mathrm{lw})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \text{ and} \tag{2.30}$$

$$\boldsymbol{O}^{(\mathrm{lb})} = \sum_{i,j=1}^{n} \tilde{W}_{ij}^{(\mathrm{lb})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t, \tag{2.31}$$

where the normalized weight matrices are defined as

$$\tilde{W}_{ij}^{(\mathrm{lw})} = \begin{cases} \tilde{A}_{ij}/n_l, & \text{if } y_i, y_j = l, \\ 0, & \text{if } y_i \neq y_j, \end{cases} \tag{2.32}$$

$$\tilde{W}_{ij}^{(\mathrm{lb})} = \begin{cases} \tilde{A}_{ij}(1/n - 1/n_l), & \text{if } y_i, y_j = l, \\ 1/n, & \text{if } y_i \neq y_j. \end{cases} \tag{2.33}$$

The normalized affinity matrix $\tilde{A}_{ij} \in [0, 1]$ between $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$ is defined as

$$\tilde{A}_{ij} = \exp\left(-\frac{(2 - 2\tilde{\boldsymbol{x}}_i^t \tilde{\boldsymbol{x}}_j)}{\tilde{\gamma}_i \tilde{\gamma}_j}\right), \tag{2.34}$$

where $\tilde{\gamma}_i = \sqrt{2 - 2\tilde{\boldsymbol{x}}_i^t \tilde{\boldsymbol{x}}_i^{(\mathrm{knn})}}$ denotes the *local* scaling of data samples in the neighborhood of $\tilde{\boldsymbol{x}}_i$, and $\tilde{\boldsymbol{x}}_i^{(\mathrm{knn})}$ is the K-th nearest neighbors of $\tilde{\boldsymbol{x}}_i$.

We will demonstrate with synthetic (and real-world hyperspectral) data that this locality preserving constraint will be particularly beneficial when class specific samples are not clustered into well-defined unimodal clusters on a unit hypersphere. Similar to ADA, the projection matrix of LADA can be defined as

$$\boldsymbol{T}_{LADA} = \underset{\boldsymbol{T} \in \mathbb{R}^{d \times r}}{\operatorname{argmin}} \left[\operatorname{tr}\left((\boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{lw})} \boldsymbol{T})^{-1} \boldsymbol{T}^t \boldsymbol{O}^{(\mathrm{lb})} \boldsymbol{T} \right) \right]. \tag{2.35}$$

As with ADA, this can be solved via the generalized eigenvalue problem. We note that LADA significantly departs from LFDA, in that while LFDA is built based on the principle of preserving locality while pushing between-class samples far apart in a Euclidean sense, LADA seeks to find compact angular clusters for within-class samples, while between-class samples are angularly maximized. Additionally, we note that by incorporating the affinity matrix $\tilde{A}_{ij}$ in $\boldsymbol{O}^{(\mathrm{lb})}$, the rank of $\boldsymbol{O}^{(\mathrm{lb})}$ is no longer limited to $c - 1$, implying that the dimensionality after an LADA can be chosen to be larger than $c - 1$.

We next investigate the benefit of the locality preserving component in the LADA formulation, for a problem where within-class samples possess a multi-modal distribution on a unit hypersphere. Assume there are several distinct clusters (multi-modal) in each class. Let $\tilde{\boldsymbol{\mu}}_{lk}$ and n_{lk} denote the normalized mean vector and the number of samples in k-th cluster of the l-th class.

Proposition 2 *Consider a scenario wherein the choice of affinity matrix accurately captures* local *neighborhood structures, such that within-class samples that belong to different clusters are not considered neighbors and vice-versa.* $\boldsymbol{O}^{(lw)}$ *and* $\boldsymbol{O}^{(lb)}$ *take the following form,*

$$\boldsymbol{O}^{(lw)} = \sum_{l=1}^{c} \frac{1}{n_l} \sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^{t} \text{ and} \tag{2.36}$$

$$\boldsymbol{O}^{(lb)} = \sum_{l=1}^{c} (\frac{1}{n_l} - \frac{1}{n}) \sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^{t} + \frac{1}{n} \sum_{l \neq m} n_l n_m \tilde{\boldsymbol{\mu}}_{l} \tilde{\boldsymbol{\mu}}_{m}^{t}. \tag{2.37}$$

The proofs of Proposition 2 is provided in Appendix A.2. It can be noticed from this proposition, LADA preserves locality by ensuring that within-class samples that belong to different clusters ($k \neq q$) do not contribute to the objective function.

To highlight the locality preserving property of LADA, we evaluate various dimensionality reduction methods using a three-dimensional, two-class multi-modal synthetic (classes are no longer uni-modal clusters) dataset. In Fig. 2.2, it can be observed that owing to the multi-modal structure of samples from class-1, LADA and LFDA can well preserve the *local* structure of class-1 samples on a two-dimensional subspace owing to the locality preserving property. We further observe that LADA provides a much better angular separation compared with LFDA.

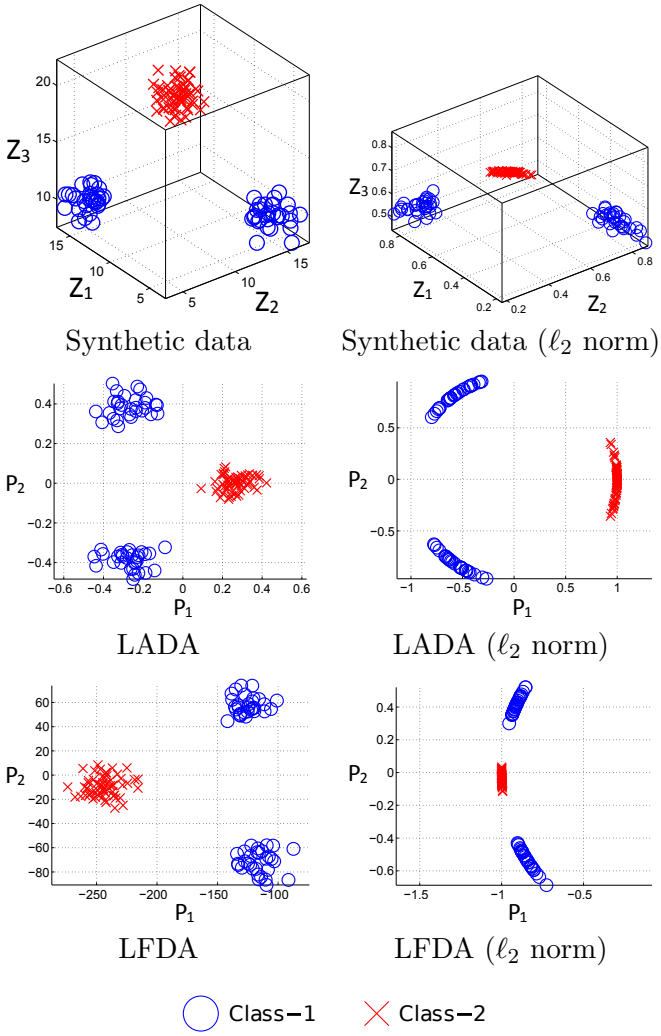


Figure 2.2: Evaluation of LADA, and LFDA using a three-dimensional three-class synthetic data generated from a unit variance Gaussian distribution.

2.3.3 Kernel variant of angular discriminant analysis

When samples from different classes are in the same direction or are angularly non-separable in the original space, both ADA and LADA will fail to find a subspace that can angularly separate between-class samples. We contend that formulating ADA / LADA in a Reproducible Kernel Hilbert Space (RKHS) $\mathcal{H}$ will overcome this limitation.

By invoking the *kernel trick* [38], ADA can be extended to its kernel variant. Specifically, $\boldsymbol{O}^{(\mathrm{w})}$ and $\boldsymbol{O}^{(\mathrm{b})}$ can be represented as

$$\boldsymbol{O}^{(\mathrm{w})} = \boldsymbol{X}\,\boldsymbol{W}^{(\mathrm{w})}\boldsymbol{X}^{t} \text{ and} \tag{2.38}$$

$$\boldsymbol{O}^{(\mathrm{b})} = \boldsymbol{X}\,\boldsymbol{W}^{(\mathrm{b})}\boldsymbol{X}^{t}. \tag{2.39}$$

The generalized eigenvalue problem in ADA can be defined as

$$\boldsymbol{X}\,\boldsymbol{W}^{(\mathrm{b})}\boldsymbol{X}^{t}\boldsymbol{\psi} = \lambda\boldsymbol{X}\,\boldsymbol{W}^{(\mathrm{w})}\boldsymbol{X}^{t}\boldsymbol{\psi}. \tag{2.40}$$

Since $\boldsymbol{\psi}$ can be represented as a linear combination of columns of $\boldsymbol{X}$, it can be formulated using a vector $\boldsymbol{\varphi} \in \mathbb{R}^n$ as

$$\boldsymbol{X}^{t}\boldsymbol{\psi} = \boldsymbol{X}^{t}\boldsymbol{X}\boldsymbol{\varphi} = \boldsymbol{K}\boldsymbol{\varphi}, \tag{2.41}$$

where $\boldsymbol{K}$ is a $n \times n$ symmetric kernel matrix. Here $K_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$ represents a simple linear kernel, although it can be replaced with any valid (nonlinear) Mercer kernel. A commonly used non-linear kernel function is the Gaussian radial basis function (RBF) which is defined as

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\Big(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\Big), \tag{2.42}$$

where σ is a free parameter.

By multiplying $\boldsymbol{X}^t$ on both sides of (2.40), we obtain the following generalized eigenvalue problem.

$$\boldsymbol{K}\,\boldsymbol{W}^{(\mathrm{b})}\boldsymbol{K}\boldsymbol{\varphi} = \lambda \boldsymbol{K}\,\boldsymbol{W}^{(\mathrm{w})}\boldsymbol{K}\boldsymbol{\varphi}. \tag{2.43}$$

Let $\boldsymbol{\Phi} = \{\boldsymbol{\varphi}_k\}_{k=1}^{r}$ be the r generalized eigenvectors associated with the r smallest eigenvalues $\lambda_1 \leq \lambda_2, \ldots, \leq \lambda_r$. A test sample $\boldsymbol{x}_{test}$ can be embedded in $\mathcal{H}$ via

$$(\boldsymbol{X}\boldsymbol{\Phi})^t \boldsymbol{x}_{test} = \boldsymbol{\Phi}^t \boldsymbol{X}^t \boldsymbol{x}_{test} = \boldsymbol{\Phi}^t \boldsymbol{K}_{\boldsymbol{X},\boldsymbol{x}_{test}}, \tag{2.44}$$

where $\boldsymbol{K}_{\boldsymbol{X},\boldsymbol{x}_{test}}$ is a $n \times 1$ vector.

Similar to KADA, the generalized eigenvalue problem of KLADA can be obtained by simply replacing the weight matrices $\boldsymbol{W}^{(\mathrm{w})}$ and $\boldsymbol{W}^{(\mathrm{b})}$ in (2.43) with their kernel versions, where the affinity matrix $\boldsymbol{A}$ is calculated in the kernel feature space.

2.3.4 Experimental results and analysis

We first describe the two hyperspectral datasets that will be used in the rest of this dissertation. The first dataset is acquired using an ITRES-CASI (Compact Airborne Spectrographic Imager) 1500 hyperspectral imager over the University of Houston campus and the neighboring urban area. This image has a spatial dimension of 1905×349 with a spatial resolution of 2.5 m. There are 15 number of classes and 144 spectral bands over the $380 - 1050$ nm wavelength range. Fig. 2.4 shows the true color image of University of Houston dataset inset with the ground truth.

The second hyperspectral data was acquired using the ProSpecTIR instrument in May 2010 over an agriculture area in Indiana, USA. This image (covering agriculture fields) has a 1342×1287 spatial dimension with 2 m spatial resolution. It has 360 spectral bands over $400 - 2500$ nm wavelength range with approximately 5 nm spectral resolution. The

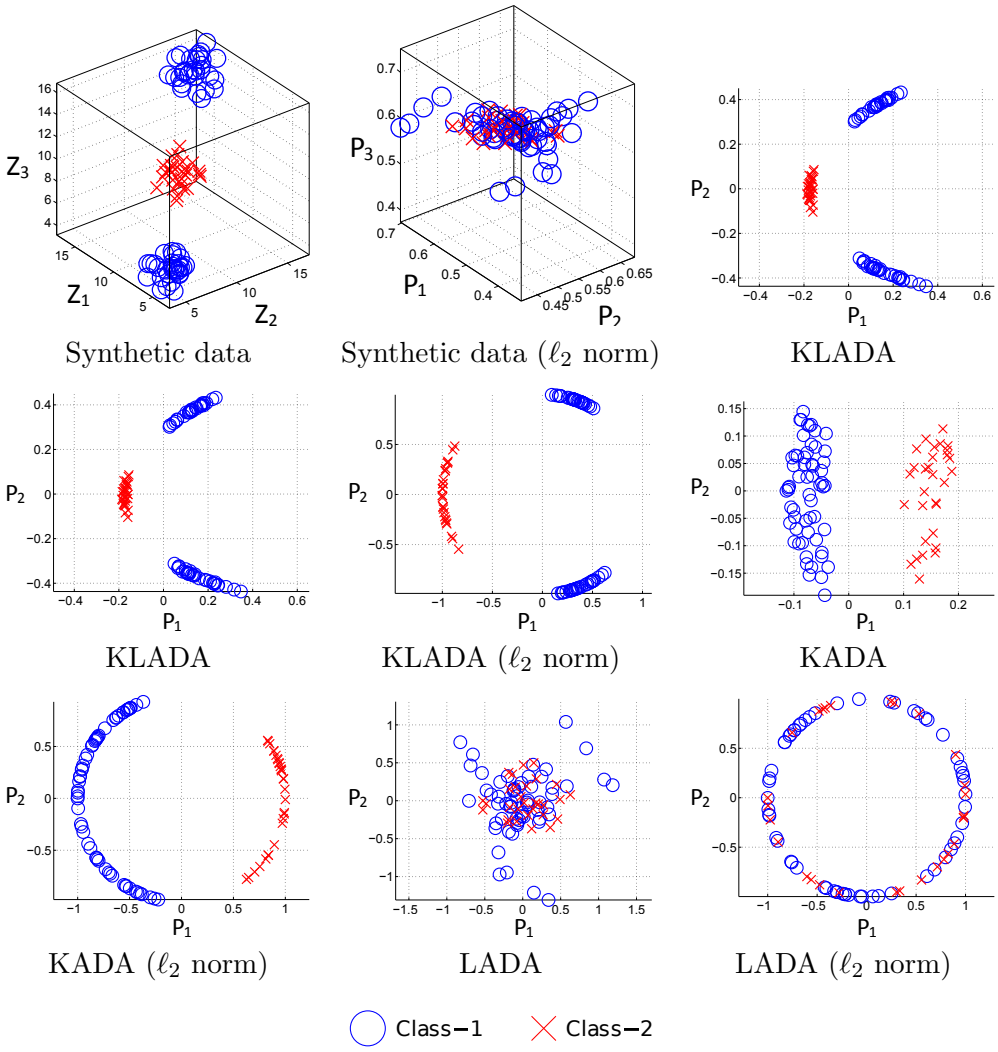


Figure 2.3: Evaluation of KLADA, KADA and LADA using a three-dimensional three-class synthetic data generated from a unit variance Gaussian distribution.

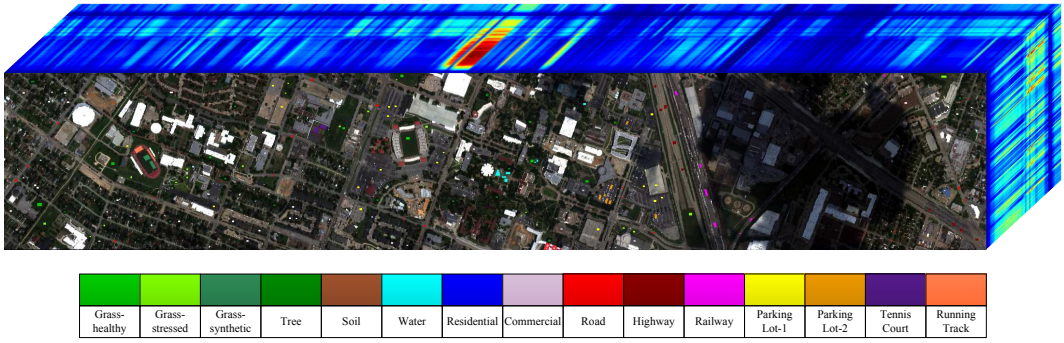


Figure 2.4: True color University of Houston hyperspectral image inset with ground truth.

19 classes consist of agriculture fields with different residue cover. Fig. 2.5 shows the true color image of the Indian Pines dataset with the corresponding ground truth.

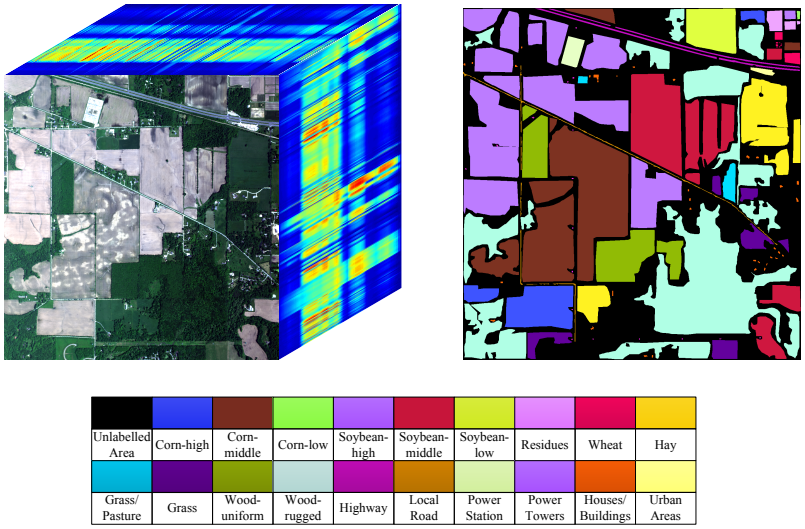


Figure 2.5: True color image (top left) and ground-truth (top right) of the Indian Pines hyperspectral image.

We now evaluate the classification performance of the proposed and existing dimensionality reduction algorithms, and show that the proposed methods outperform other existing methods including CDA, LDA, LFDA, generalized discriminant analysis (GDA) [2], kernel *local* Fisher discriminant analysis (KLFDA) [8], traditional NN, SRC, SRC-ℓ_1 and nonlinear kernel based support vector machine (SVM). In SRC, the sparse coefficient vector is

learned via OMP, and SRC-ℓ_1 uses a gradient project [39] to obtain the sparse coefficient. Note that the atom selection process in OMP used in this work is based on the maximal normalized inner product instead of maximal absolute normalized inner product between the residual vector and atoms in the dictionary. This is due to the fact that the angular separation between samples may potentially be larger than 90^o after the projection, and the normalized inner product considers angles between 0^o to 180^o (the normalized absolute inner product on the other hand restricts the range of angles between 0^o and 90^o). The time complexity of NN and SRC (with OMP as the recovery method) is $\mathcal{O}(dn)$ and $\mathcal{O}(dnS)$ where S is the sparsity level in OMP. Kernel functions used in KLADA, KADA, KLFDA, GDA and SVM are all based on the RBF kernel defined in (2.42). For both hyperspectral datasets described above, 200 samples per class are used for evaluation, and 10, 30, 50 and 100 samples per class are used for training respectively. Both testing and training samples are drawn randomly from the ground truth without overlapping with each other. Each experiment has been repeated 10 times and the average accuracy with its standard deviation are reported. The parameter values of each algorithm are tuned by searching through a wide range of the parameter space, and the performance reported here represents the "optimal" parameters.

Table 2.1 shows the mean classification accuracies along with the corresponding standard deviations as a function of training sample size for the University of Houston and Indian Pines datasets. Since LFDA and LDA are Euclidean distance based dimensionality reduction methods, the distance used in these methods is based on Euclidean distance, while others are based on spectral angle distance. It can be seen from these results that the proposed methods generally outperform other existing methods, especially when the

Table 2.1: Classification accuracy (%) and standard deviation (in parenthesis) as a function of number of training samples per class.

Dataset		*University of Houston*				*Indian Pines*			
Algorithm / Sample Size		*10*	*30*	*50*	*100*	*10*	*30*	*50*	*100*
KLADA	***NN***	84.7 (1.5)	93.9 (0.5)	96.9 (0.4)	98.7 (0.2)	80.3 (2.0)	88.9 (0.8)	90.9 (0.7)	93.3 (0.6)
	SRC	85.0 (1.3)	94.2 (0.6)	97.1 (0.4)	98.8 (0.2)	80.3 (1.9)	89.0 (0.8)	91.0 (0.7)	93.3 (0.6)
LADA	***NN***	81.8 (1.2)	91.4 (0.5)	95.5 (0.5)	98.4 (0.3)	73.3 (1.1)	82.0 (0.5)	85.2 (0.6)	88.8 (0.5)
	SRC	82.0 (1.1)	91.5 (0.5)	95.5 (0.5)	98.4 (0.3)	73.7 (1.1)	82.3 (0.5)	85.4 (0.5)	88.9 (0.5)
KADA	***NN***	82.4 (1.4)	90.2 (1.3)	94.0 (0.5)	97.4 (0.3)	72.0 (1.5)	80.3 (0.6)	83.3 (1.2)	86.9 (0.5)
	SRC	82.7 (1.2)	90.4 (1.2)	94.0 (0.4)	97.5 (0.3)	72.6 (1.5)	80.6 (0.7)	83.5 (1.2)	87.0 (0.5)
ADA	***NN***	78.9 (1.7)	87.6 (0.9)	91.6 (0.5)	95.9 (0.4)	69.3 (1.4)	77.9 (1.0)	81.2 (1.2)	85.6 (0.5)
	SRC	80.0 (1.7)	87.9 (1.1)	91.8 (0.9)	95.9 (0.5)	70.0 (1.3)	78.3 (1.1)	81.6 (0.9)	85.8 (0.6)
CDA	***NN***	75.5 (1.3)	86.2 (0.6)	91.2 (0.5)	96.3 (0.4)	63.6 (1.1)	73.6 (0.8)	76.9 (1.3)	81.2 (0.5)
	SRC	77.7 (0.8)	86.6 (0.6)	91.6 (0.6)	96.4 (0.4)	65.7 (1.1)	74.6 (0.8)	77.9 (1.4)	81.6 (0.4)
KLFDA	***NN***	77.6 (1.2)	86.0 (0.7)	90.3 (0.7)	95.4 (0.6)	75.1 (1.4)	81.9 (1.3)	87.7 (1.1)	90.4 (0.5)
LFDA	***NN***	73.5 (1.3)	82.2 (1.1)	88.4 (0.8)	94.9 (0.5)	69.2 (1.4)	74.8 (1.2)	86.2 (0.7)	89.6 (0.6)
GDA	***NN***	66.5 (3.3)	83.4 (1.1)	89.0 (1.2)	95.2 (0.9)	68.1 (3.5)	80.9 (2.9)	85.1 (2.1)	88.6 (0.9)
LDA	***NN***	35.6 (7.1)	81.3 (0.3)	86.4 (0.7)	93.6 (0.5)	16.0 (4.5)	74.9 (1.2)	84.0 (0.8)	88.9 (0.5)
NN		76.6 (1.1)	87.1 (0.6)	91.8 (0.5)	96.6 (0.4)	67.2 (0.7)	76.9 (0.9)	80.2 (1.2)	84.5 (0.6)
SRC		78.7 (0.8)	88.5 (0.5)	92.2 (0.6)	96.9 (0.4)	69.1 (1.2)	78.2 (0.7)	81.0 (1.0)	85.2 (0.6)
SRC-ℓ_1		80.4 (1.1)	89.3 (0.7)	92.7 (0.4)	96.0 (0.3)	75.4 (2.6)	82.1 (0.9)	84.3 (1.0)	86.7 (0.5)
SVM		79.1 (1.2)	88.8 (0.7)	92.9 (0.8)	97.2 (0.3)	71.4 (1.4)	83.1 (0.9)	86.9 (0.8)	91.3 (0.3)

number of training samples per class is small.

To provide insights on the benefits of the proposed dimensionality reduction methods on real-world hyperspectral data, we depict class-specific accuracies for the University of Houston dataset in Table 2.2. We draw attention to *difficult* classes, particularly, Road, Highway, Railway, Parking Lot-1 and Parking Lot-2. These classes are *difficult* in particular because they are spectrally very similar (with regards to their spectral shape) as shown in Fig. 2.6. From the table, we observe that the *local* variants of KLADA and LADA generally consistently outperform the *non-local* variants of ADA and KADA by preserving the *local* structure of the data. Further, the kernel variants outperform their non-kernel counterparts when the signatures of different classes are spectrally very similar (i.e., for difficult classes).

We next demonstrate the benefit of kernel variants of the proposed methods for robust classification of sub-pixel classes — a scenario commonly found in remotely sensed hyperspectral imagery, wherein classes of interest are often mixed with background due to low spatial resolution. The mixed pixels are (synthetically) generated from real "pure" pixels

by mixing the pure target spectra for each class with background spectra (from all other classes) via a linear mixing model. The larger the background abundance (BA), the smaller the fraction of the target in the pixel. Table 2.3 shows the classification accuracy with the University of Houston dataset under pixel mixing. 30 training and 200 test samples per class are used in this experiment. We use NN as the back-end classifier for the proposed (and baseline) dimensionality reduction methods. It can be seen that the proposed methods outperform other dimensionality reduction methods.

Table 2.2: Class-specific accuracies (%) for the University of Houston dataset.

Class / Algorithm	***KLADA***	***KADA***	***LADA***	***ADA***
Grass-healthy	99.9	98.4	98.4	98.7
Grass-stressed	97.9	98.9	97.7	96.3
Grass-synthetic	99.9	99.5	100.0	99.7
Tree	99.8	98.5	97.8	99.0
Soil	99.6	99.4	99.9	99.0
Water	98.4	97.7	98.8	98.1
Residential	93.9	88.2	90.9	83.1
Commercial	85.4	85.1	86.7	83.1
Road	88.3	76.9	79.0	72.9
Highway	94.8	82.9	86.6	79.8
Railway	96.1	86.0	79.3	73.0
Parking Lot 1	84.9	77.1	82.4	76.1
Parking Lot 2	70.8	66.7	73.5	56.3
Tennis Court	99.3	99.2	99.5	98.7
Running Track	99.0	98.4	99.9	99.5

Table 2.3: Classification accuracy (%) as a function of BA (%) for University of Houston dataset.

Algorithm / BA	***10***	***20***	***30***
KLADA	91.8 (1.2)	89.9 (0.9)	85.7 (1.1)
LADA	90.6 (0.7)	89.4 (0.7)	85.4 (0.8)
KADA	89.5 (1.4)	87.3 (1.0)	83.3 (1.5)
ADA	84.7 (1.1)	79.9 (1.0)	74.6 (0.7)
CDA	79.7 (0.5)	73.3 (1.0)	66.5 (0.9)
KLFDA	82.2 (1.1)	77.6 (1.1)	72.4 (1.2)
LFDA	80.8 (0.8)	79.8 (0.9)	77.3 (1.1)
GDA	81.2 (0.7)	75.3 (1.0)	68.6 (1.0)
LDA	80.8 (0.4)	80.0 (0.7)	78.6 (0.6)

In what follows, we demonstrate that the proposed approaches are suitable for SRC with OMP based coefficient recovery. We first define the reconstructive error caused by the

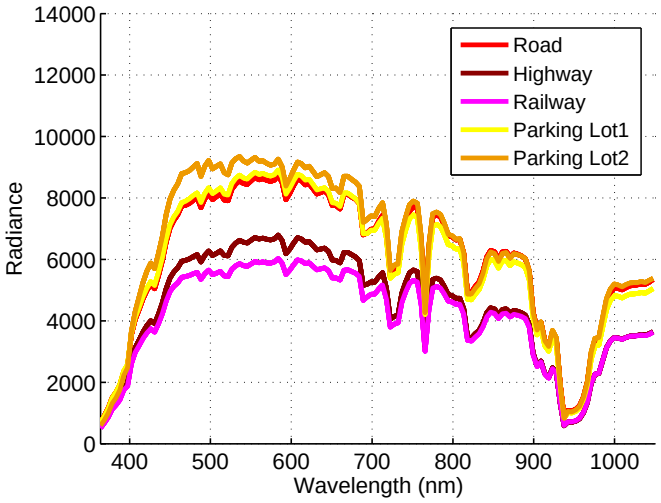


Figure 2.6: The mean spectral signatures of the most five confusing classes in the University of Houston dataset.

intra-class $E^{(\mathrm{w})}$ and inter-class data $E^{(\mathrm{b})}$ [40] as

$$E^{(\mathrm{w})} = \frac{1}{n}\sum_{l=1}^{c}\sum_{i:y_i=l}^{n_l} \|\boldsymbol{x}_i - \boldsymbol{X}\delta_l(\hat{\boldsymbol{\alpha}})\|^2 \text{ and} \tag{2.45}$$

$$E^{(\mathrm{b})} = \frac{1}{n(c-1)}\sum_{l=1}^{c}\sum_{i:y_i=l}^{n_l}\sum_{m\neq l} \|\boldsymbol{x}_i - \boldsymbol{X}\delta_m(\hat{\boldsymbol{\alpha}})\|^2, \tag{2.46}$$

where $\hat{\boldsymbol{\alpha}}$ is obtained via OMP.

The ratio of $E^{(\mathrm{b})}$ to $E^{(\mathrm{w})}$ is hence a reasonable heuristic to gauge the suitability of the subspace for SRC. We use the University of Houston dataset with 30 training samples per class to calculate the ratio of inter-class to intra-class reconstruction error in the projected subspace obtained by LADA, ADA, CDA, LFDA and LDA. The sparse coefficient $\hat{\boldsymbol{\alpha}}$ is calculated by OMP with an optimal sparsity level determined empirically for each algorithm. Based on Table 2.4, we can infer that the proposed approaches produce a larger ratio than the traditional approaches, which indicates that the classification ability of SRC is better in LADA and ADA projected subspaces compared with CDA, LFDA and LDA projected subspaces.

Table 2.4: Ratio of inter-class to intra-class reconstruction error calculated in the projected space

Algorithm	***LADA***	***ADA***	***CDA***	***LFDA***	***LDA***
$E^{(\mathrm{b})}/E^{(\mathrm{w})}$	12.1	9.4	6.8	5.8	3.5

Finally, we demonstrate the effect of the dimensionality of the projected subspace on the performance of the proposed methods as well as NN applied directly on the input space (without any dimensionality reduction) for the University of Houston dataset. In this experiment, we randomly choose 30 training samples per class and 200 test samples per class. The reduced dimensionality ranges from 5 to 140. Each experiment is repeated 10 times and the average accuracy is reported for each method. Fig. 2.7 shows the classification accuracies as a function of the number of dimensions retained after each projection. The accuracy for NN is constant as a function of dimensionality, since no dimensionality reduction is performed beforehand. Based on Fig. 2.7, the optimal dimensionality for KLADA, LADA, KADA, and ADA are found to be 45, 20, 20, and 30 respectively. Note that for ADA, although the upper limit on the number of relevant dimensions is $c-1$ (which is 14 for this dataset), utilizing additional dimensions indeed increases class separability by enhancing angular separability. We conjecture that this phenomenon is related to complex data distributions, wherein adding additional dimensions (that do not necessarily contribute to the objective function) enhances classification performance.

2.4 Proposed spatially-driven angle preserving projection

2.4.1 Local similarity preserving projection

In this paper, we seek to make two related contributions within the context of angular discriminant analysis — (1) developing an unsupervised approach to spectral angle based subspace learning, where local spectral angles are preserved following this unsupervised

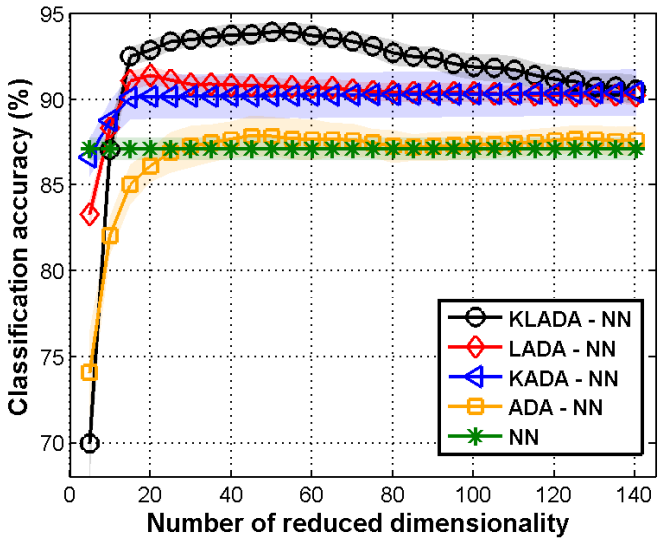


Figure 2.7: Effect of the dimensionality of the projected subspace on the performance of the proposed methods for University of Houston hyperspectral dataset.

project, and (2) developing a projection that incorporates spatial information when learning such an *optimal* projection. We first form a unsupervised version of ADA which we refer to as local similarity preserving projection (LSPP). It seeks a lower dimensional space where the correlation or angular relationship between samples that are neighbors in the feature space are preserved. We can also think of it as an angular equivalent of the commonly employed locality preserving projection (LPP) [7].

Let x_i be the i-th training sample and P be the $d \times r$ projection matrix, where r is the

reduced dimensionality. The objective function of LSPP can be simplified as

$$\begin{aligned} I &= \sum_{ij} W_{ij}(P^t x_i)^t (P^t x_j) \\ &= \sum_{ij} tr\left[W_{ij}(P^t x_i)^t (P^t x_j)\right] \\ &= \sum_{ij} tr\left[W_{ij} P^t x_j (P^t x_i)^t\right] \\ &= \sum_{ij} tr\left[P^t W_{ij} x_i x_j^t P\right] \\ &= tr\left[P^t X W X^t P\right]. \end{aligned} \tag{2.47}$$

The heat kernel $W_{ij} \in [0,1]$ between x_i and x_j is defined as

$$W_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma}\right), \tag{2.48}$$

where σ is the parameter in the heat kernel.

We impose a constraint ($P^t X D X^t P = 1$ where $D_{ii} = \sum_j W_{ij}$) to avoid biases caused by different samples. The bigger the value D_{ii} which is corresponding to i-th training sample, the more important i-th training sample is.

The final objective function is defined as

$$\underset{P}{\text{argmax}} \;\; tr\left[P^t X W X^t P\right] \qquad \text{s. t.} \quad P^t X D X^t P = I. \tag{2.49}$$

The problem in 2.49 can be solved as a generalized eigenvalue problem as

$$X W X^t P = \lambda X D X^t P. \tag{2.50}$$

The projection matrix P are the eigenvectors corresponding to the r largest eigenvalues.

2.4.2 Spatially-driven local similarity preserving projection

It is well understood from many recent works [28, 41–43], that by aking into account the spatial neighborhood information, hyperspectral image classification accuracy can be significantly increased. This is based on the observation that spatially neighboring samples in hyperspectral data often consist of similar materials,and hence they are spectrally correlated. In order to utilize spatially neighboring samples in a lower-dimensional subspace, the spatial neighborhood relationship of hyperspectral data should be preserved. To address this problem, we propose a spatial variant of LSPP (SLSPP) in this work to further improve the classification accuracies. Let $\{z_k, k \in \Omega_i\}$ be the spatial neighborhood samples around a training sample x_i, then the objective function of SLSPP can be reduced to

$$\begin{aligned} I &= \sum_i \sum_{k\in\Omega_i} W_{ik}(P^t x_i)^t (P^t z_k) \\ &= \sum_i \sum_{k\in\Omega_i} tr\left[W_{ik}(P^t x_i)^t (P^t z_k)\right] \\ &= \sum_i \sum_{k\in\Omega_i} tr\left[W_{ik} P^t z_k (P^t x_i)^t\right] \\ &= \sum_i \sum_{k\in\Omega_i} tr\left[P^t W_{ik} z_k x_i^t P\right] \\ &= tr\left[P^t M P\right], \end{aligned} \tag{2.51}$$

where $M = \sum_i \sum_{k\in\Omega_i} W_{ik} z_k x_i^t$.

The final objective function is defined as

$$\underset{P}{\operatorname{argmax}} \; tr\left[P^t M P\right] \qquad \text{s.t.} \quad P^t P = I. \tag{2.52}$$

Similar to LSPP, the projection matrix P are the eigenvectors corresponding to the r largest eigenvalues.

Note that both LSPP and SLSPP are unsupervised projections in the sense that they do not require labels when learning the projections. We also note that a projection such as SLSPP is particularly beneficial when the backend classifier utilizes the spatial structure in the spatially neighboring pixels for each test pixel during classification. Towards that end, we next propose a modified sparse representation based classifier that utilizes sparse representations of the entire spatial neighborhood of a test pixel simultaneously when making a decision.

2.4.3 Experimental results and analysis

The first experimental hyperspectral dataset employed was collected using the Reflective Optics System Imaging Spectrometer (ROSIS) sensor [44]. This image, covering the University of Pavia, Italy, has 103 spectral bands with a spatial coverage of 610×340 pixels, and 9 classes of interests are considered in this dataset. A three-band true color image and its ground-truth are shown in Fig. 2.8.

The second hyperspectral data used in this work was acquired by us in Galveston, Texas in October, 2014 which includes two different wetland scenes captured at ground-level (side-looking views) over wetlands in Galveston. The two image cubes are referred to as area 1, and area 2, representing different regions of the wetlands that were imaged — In addition to common wetland classes, area 2 has Black Mangrove (*Avicennia germinans*) trees in the scene, a species which is of particular interest in ecological studies of wetlands, in addition to Spartina. This data was acquired using a Headwall Photonics hyperspectral imager which provides measurements in 325 spectral bands with a spatial size of 1004×5130. The hyperspectral data uniformly spanned the visible and near-infrared spectrum from 400 nm - 1000 nm. The objects of interests are primarily vegetation species common in such wetlands.

Six different classes were identified in area-1 including soil, *symphyotrichum*, *schoenoplectus*, *spartina patens*, *borrichia* and *rayjacksonia*. The second area includes *Avicennia germinans*, *batis*, *schoenoplectus*, *spartina alterniflora*, *soil*, *water* and *bridge*. Since soil and *schoenoplectus* are included in both areas, the total number of classes in the combined library are eleven.

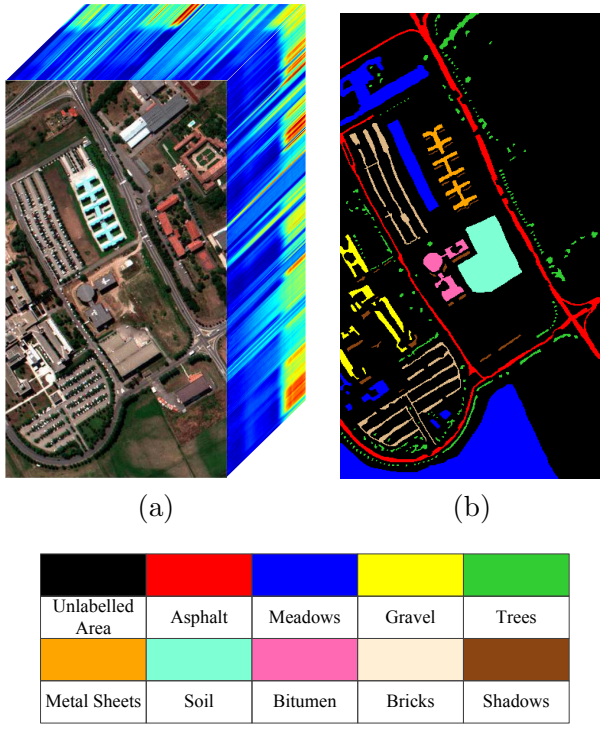
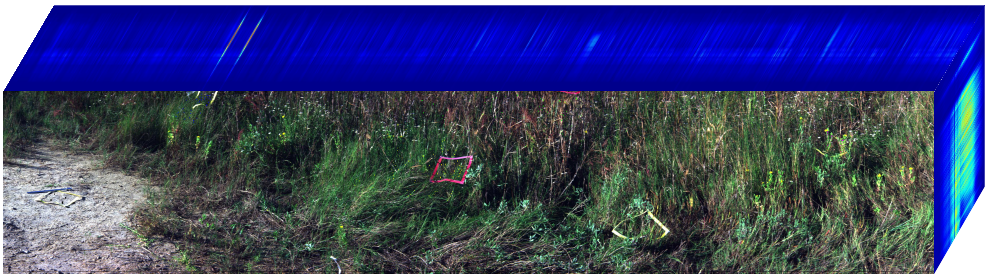


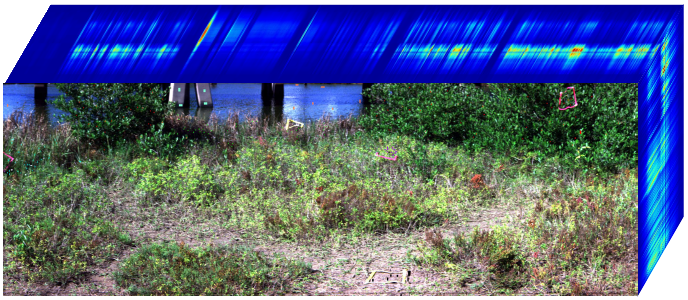
Figure 2.8: (a) True color image and (b) ground-truth of the University of Pavia hyperspectral data.

The efficacy of the proposed LSPP, SLSPP and SBOMP are evaluated as a function of training samples per class using the two practical hyperspectral datasets mentioned previously. SLSPP–SBOMP-C indicates the data is projected based on SLSPP, and SBOMP-C is employed as the backend classifier. A nearest-neighbor (NN) classifier with cosine angle

distance is used after LSPP and LADA projection, since they do not take spatial information into account when deriving the projection matrix. Similar to SLSPP–SOMP-C, LSPP–NN and LADA–NN. Each experiment is repeated 10 times using a repeated random subsampling validation technique, and the average accuracy is reported. The number of test samples per class is fixed to 100 for every random subsampling.



Wetland data, area - 1



Wetland data, area - 2

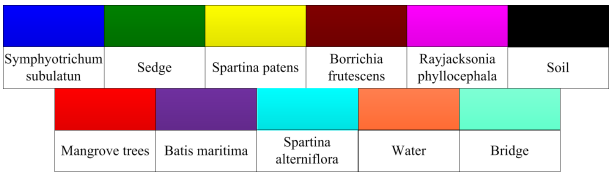


Figure 2.9: True color images of the wetland dataset inset with ground truth.

Fig. 2.10 shows the classification accuracies as a function of training samples for the University of Pavia hyperspectral data. The parameter for each algorithm is determined via searching through a wide range of the parameter space and the accuracies reported in this plot is based on the optimal parameter values. As can be seen from this figure, our proposed

SLSPP followed by SBOMP-C gives the highest classification accuracies consistently over a wide range of the training sample size. LSPP–NN also gives better classification result compared with LADA–NN.

We perform a similar analysis using the wetland hyperspectral data. Fig. 2.11 and Fig. 2.12 plot the classification accuracies with respect to the training sample size per class for wetland area - 1 and wetland area - 2 respectively. It can be seen from the two plots that the proposed methods generally outperform other baseline methods for the vegetation type of hyperspectral data which demonstrate the diverse applicability of the proposed methods.

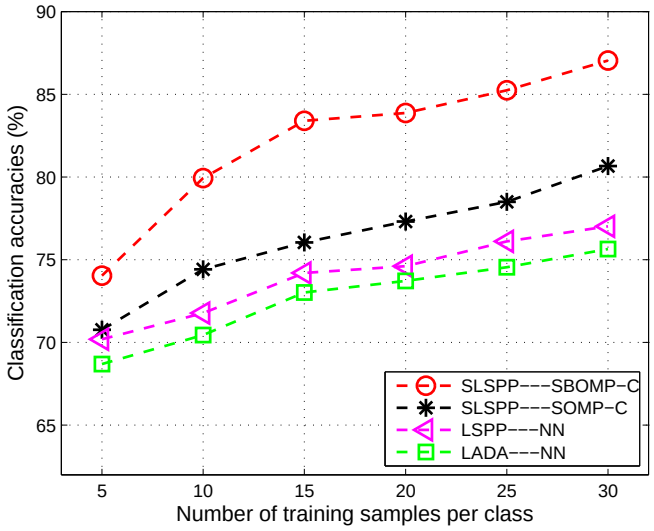


Figure 2.10: Overall classification accuracy (%) versus number of training samples for the University of Pavia data.

The class specific accuracies for different datasets are shown in Table 2.5, Table 2.6 and Table 2.7 respectively. In this experiment, the training sample size per class is fixed to 10 and the test sample size is 100 per class. Each experiment is repeated 10 times and the

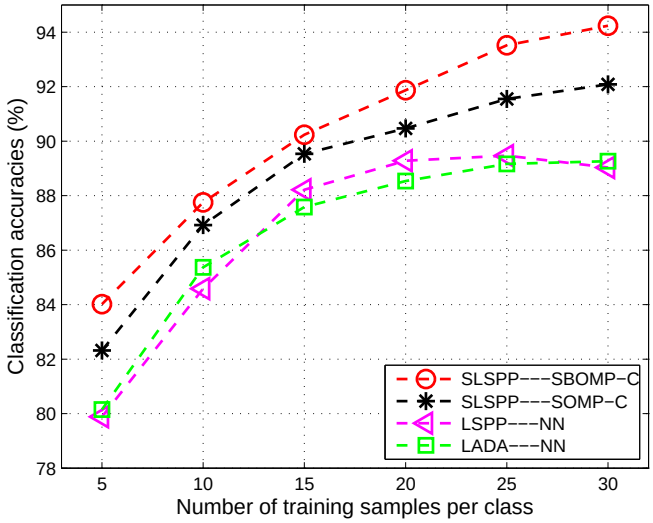


Figure 2.11: Overall classification accuracy (%) versus number of training samples for the wetland data, area - 1.

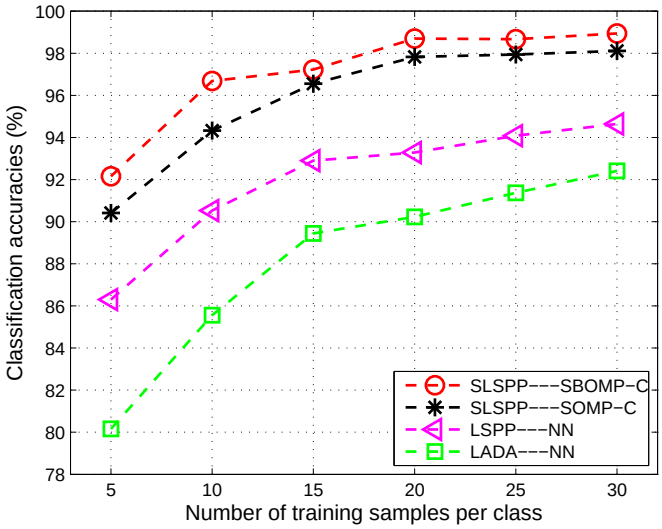


Figure 2.12: Overall classification accuracy (%) versus number of training samples for the wetland data, area - 2.

average accuracy is reported. As can be seen from the class-specific tables for the wetland data, the spatial information plays a crucial role, especially for the species with complex textures and shapes such as *Spartina-Patens*, *Sedge* and *Symphyotrichum*. These three species are shown in Fig. 2.13.

Table 2.5: Class-specific accuracies (%) for the University of Pavia Data.

Class Name / Algorithm	***LSPP–SBOMP-C***	***LSPP–SOMP-C***	***LSPP–NN***	***LADA–NN***
Asphalt	77.2	60.8	31.4	30.9
Meadows	64.4	62.5	60.2	59.6
Gravel	79.2	65.5	61.1	59
Trees	88.2	79	94.4	94.1
Metal Sheets	98.9	98.2	99.9	99.9
Soil	67.6	59.1	57.1	53.4
Bitumen	84.9	83	84.8	84.4
Bricks	60.1	64.8	66.2	62.9
Shadows	99.2	96.9	96.9	95.9
Overall Accuracy	80.0	74.4	72.4	71.1

Table 2.6: Class-specific accuracies (%) for the Wetland, Area-1 data.

Class Name / Algorithm	***LSPP–SBOMP-C***	***LSPP–SOMP-C***	***LSPP–NN***	***LADA–NN***
Soil	99.8	100	99.8	99.9
Symphyotrichum	86.8	85.5	78.1	76.1
Sedge	93.4	93.3	91.6	92.1
Spartina-Paten	68.4	59.1	64.2	56
Borrichia	91.6	91.2	91.7	93.7
Rayjacksonia	91.6	89	84.7	92.2
Overall Accuracy	88.6	86.4	85.0	85.0

Table 2.7: Class-specific accuracies (%) for the Wetland, Area-2 data.

Class Name / Algorithm	***LSPP–SBOMP-C***	***LSPP–SOMP-C***	***LSPP–NN***	***LADA–NN***
Soil	96.1	96.8	87.7	86.7
Mangrove Tree	97.1	95	97.1	98.5
Batis	99.2	99.1	97.9	97.2
Sedge	97	94.3	92.3	89.8
Spartina-Alterniflora	89.7	90.2	75.3	48
Water	99	99	98.5	98.7
Bridge	99.9	99.4	94.3	85.7
Overall Accuracy	96.9	96.3	91.9	86.4

Next, we analyze the effect of the window size (that defines the spatial neighborhood) for the SLSPP method. Fig. 2.14 depicts the classification accuracies as a function of different

Spartina-Paten Sedge Symphyotrichum

Figure 2.13: Three wetland species having complex spatial texture features.

window size using the University of Pavia dataset. From this figure, we can see that the optimal window size is 5.

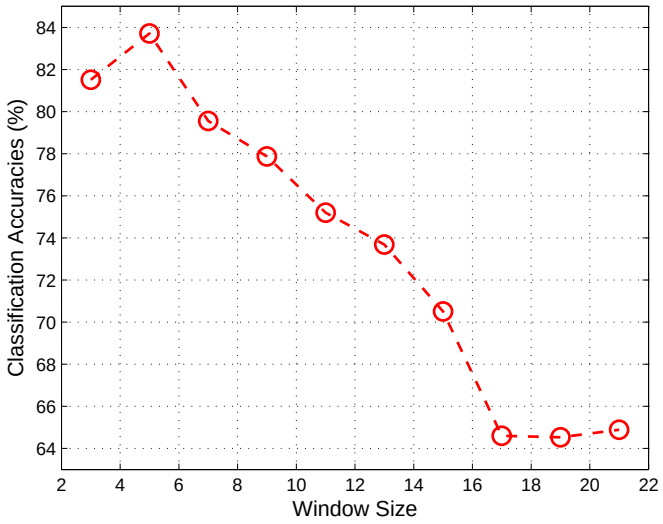


Figure 2.14: Overall classification accuracy (%) versus different window size for the University of Pavia data.

In this work, we also visualize the data distributions after projection for the proposed angle-based SLSPP and the Euclidean-based LPP methods. Figure 2.15 (a) shows the subset image for University of Pavia and Figure 2.15 (b) plots all the training samples used in this experiment on an ℓ_2 normalized sphere. Figure 2.15 (c) and (d) show the same samples after an SLSPP and LPP projection on an ℓ_2 normalized sphere respectively. U_1, U_2 and U_3 are the three projections found by SLSPP and LPP corresponding to the largest

eigenvalues. As can be seen from this figure, SLSPP is much more effective at preserving the inter-sample relationships in terms of spectral angle in the lower-dimensional subspace compared to LPP.

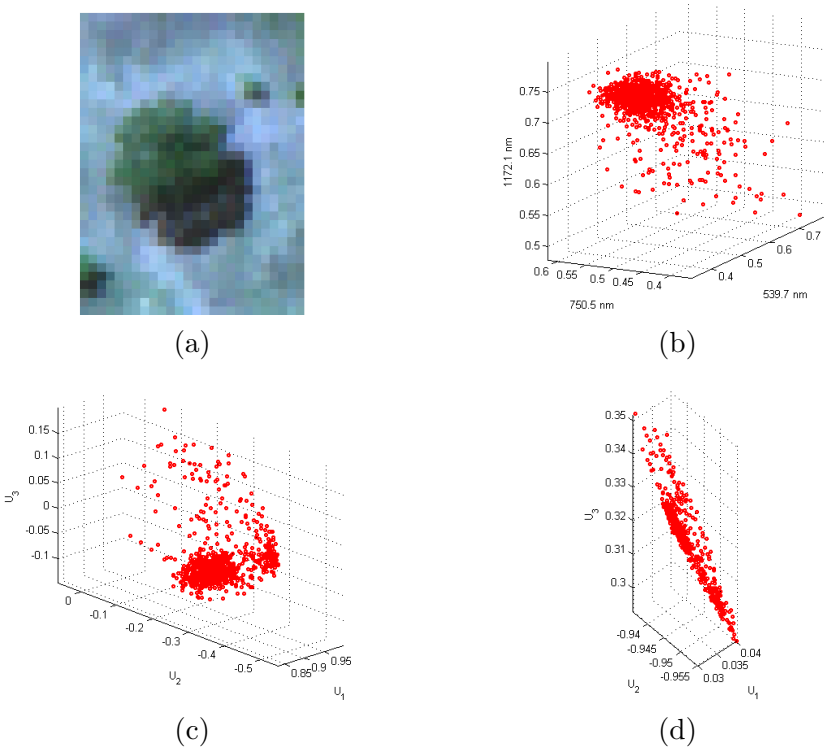


Figure 2.15: Illustrating the (a) subset image of the University of Pavia, (b) original samples, (c) SLSPP projected samples and (d) LPP projected samples on the sphere.

2.5 Conclusion

The proposed ADA seeks to learn an "optimal" projection in an angular sense, wherein the ratio of within-class to between class inner products after an ℓ_2 normalization is maximized in the projected space. The optimization problem formed by ADA can be solved by a simple generalized eigenvalue problem, and is readily extended to its locality-preserving and kernel variants (which are also developed in this paper). We also provide geometrical

insight for the proposed methods. In this work, ADA is used as a feature pre-processing with the goal of improving the classification ability of NN with cosine angle distance and SRC with OMP as the recovery method. Since OMP selects atoms based on the normalized inner products, it is expected that the accuracy of coefficient recovery will increase after an ADA projection. LADA is proposed to address the scenario wherein class specific samples are not clustered into well-defined unimodal clusters on a unit hypersphere, but are rather dispersed across multiple clusters. The nonlinear kernel variant proposed in this paper is beneficial when between-class samples are distributed along the same radial direction or angularly non-separable in the original space.

Besides the supervised angle-based dimensionality reduction, we have presented an unsupervised variant (LSPP) of the recently developed supervised dimensionality reduction method — ADA, as well as its spatial variant, SLSPP, that utilize spatial information around the samples when learning the projections. By incorporating spatial information in the dimensionality reduction projection, we are able to learn much more effective subspaces that not only preserves angular information among the training pixels in the feature space, but also their spatial neighbors.

Experimental results based on different benchmarking hyperspectral datasets show that the proposed dimensionality reduction methods outperform other existing traditional dimensionality reduction methods — the resulting classification performance is similar or better than the nonlinear SVM a common benchmarking algorithm for hyperspectral data.

Chapter 3

Sparse Representation-Based Classification

3.1 Introduction

In this work, we propose class dependent sparse representation classifier (cdSRC) to exploit the spectral content of HSI based on sparse representation. In essence, cdSRC effectively combines the ideas of SRC and KNN in a class-wise manner to exploit both correlation and Euclidean distance relationship between test and training samples. Towards this goal, a unified class membership function is developed, that utilizes residual and distance information simultaneously. In doing so, cdSRC not only utilizes the correlation information but also harnesses the Euclidean information which may be lost due to the ℓ_2 normalization (the length of each of atoms in the dictionary is unit length) preprocessing step in the SRC. Experimental results based on several real-world hyperspectral datasets demonstrate that cdSRC can dramatically increase the classification performance of traditional SRC and also outperform kernel based SVM. We also propose an extension of the proposed algorithm in a kernel induced space.

Finding the sparsest solution in SRC is a combinatorial problem as it involves searching through every combination of S atoms in a dictionary, where S denotes the optimal sparsity

level. There are two major approaches to approximate this problem. One is to relax this non-convex combinatorial problem into an ℓ_1 convex optimization problem — also known as basis pursuit. Several methods have been proposed to solve this ℓ_1-norm problem including interior-point method [45], gradient projection [39] etc. The other major category is based on iterative greedy pursuit algorithms such as matching pursuit, orthogonal matching pursuit (OMP) and orthogonal least square (OLS). These greedy approaches have been widely used due to their computational simplicity and easy implementation. They find an atom at a time based on different criterion and update the sparse solution iteratively. Among these approaches, the OMP algorithm is by far the most popular approach and is used in a wide range of applications. OLS is similar to OMP except for the atom selection process. A major difference between OMP and OLS relies on their atom selection procedure in that OMP selects an atom that best correlates with the current residual, while OLS selects an atom giving the smallest residual after orthogonalization. Note that the first atom selected by OMP is identical to OLS. For more detailed information about the differences between these two algorithms, readers can refer to [46, 47] and a k-step analysis of OMP and OLS can be found in [48].

OLS has been widely used in many applications [49–53], but it has not gained much attention for classification problems. In [54], the authors implement SRC in a classwise manner to improve the classification accuracy, in which the sparse coefficient is recovered by OMP. In this work, we implement A class-dependent version of OLS to perform classification. Since OLS produces lower signal reconstruction error compared to OMP under similar condition [46] (such as the same sparsity level, same dictionary etc.) — an observation that will be further analyzed and explained in the next section, we hypothesize that

more accurate signal estimation will further improve the classification performance of SRC. Compared with convex optimization based techniques such as interior point and gradient projection methods [39, 45], greedy pursuit-based approaches are more efficient and appropriate to recover the sparse coefficient in SRC due to their low time-complexity. By using the kernel trick, we extend the proposed cdOLS into its kernel variant to handle nonlinearly separable data as well.

In [28], the authors propose a join sparsity model to incorporate the contextual information of test samples to improve the classification performance of SRC. However, the contextual information of training samples have not been used. In this work, we also propose a sparse representation based classifier which takes into account the spatial information for both training and test samples in this work.

3.2 Related work

In this section, we introduce several popular classification methods used for remote sensed image classification such as nearest neighbor (NN) classifier, sparse representation-based classification (SRC), and support vector machine (SVM).

3.2.1 Nearest neighbor-based classification

The NN classifier is a nonparametric classification method that assigns a test sample $\boldsymbol{x}_{test}$ to the l-th class if its nearest (measured by an appropriate distance metric) training sample belongs to class l. The Euclidean distance D_E is a commonly used, though angular cosine distance D_C [14, 15] is also used to measure the similarity between a test sample

$\boldsymbol{x}_{test}$ and a training sample $\boldsymbol{x}_i$ which are defined as

$$D_E(\boldsymbol{x}_{test}, \boldsymbol{x}_i) = \|\boldsymbol{x}_{test} - \boldsymbol{x}_i\| \text{ and} \tag{3.1}$$

$$D_C(\boldsymbol{x}_{test}, \boldsymbol{x}_i) = 1 - \tilde{\boldsymbol{x}}_{test}^t \tilde{\boldsymbol{x}}_i.$$

3.2.2 Sparse representation-based classification

In SRC [26], a test sample $\boldsymbol{x}_{test}$ is represented as a linear combination of the available training samples in $\boldsymbol{X}$,

$$\boldsymbol{x}_{test} = \boldsymbol{X}\boldsymbol{\alpha}, \tag{3.2}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n]^t$ is a coefficient vector corresponding to all training samples. In an ideal case, if a test sample $\boldsymbol{x}_{test}$ belongs to the l-th class, the entries of $\boldsymbol{\alpha}$ are all zeros except those related to the training samples from the l-th class.

To obtain the *sparsest* solution in (3.2), one can solve the following optimization problem:

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_0, \quad \text{s. t.} \quad \boldsymbol{X}\boldsymbol{\alpha} = \boldsymbol{x}_{test}, \tag{3.3}$$

where the ℓ_0 norm $\|\cdot\|_0$ simply counts the number of nonzero entries in $\boldsymbol{\alpha}$. Problem (3.3) is NP hard — hence, as is common in other related applications [55, 56], the ℓ_0 norm can be relaxed with an ℓ_1 norm — the resulting approach is referred to as basis pursuit [57]. This can be cast as a linear programming problem and can be solved via gradient projection [39] or interior-point method [45]. The ℓ_0 norm problem can also be approximately solved by greedy pursuit algorithms such as OMP. Before computing the sparse coefficient based on the methods described above, atoms in the dictionary need to be ℓ_2 normalized to avoid biases caused by atoms with varying lengths [26].

After calculating the sparse coefficient vector $\hat{\boldsymbol{\alpha}}$, the residual of each class can be calculated via

$$\boldsymbol{r}_l(\boldsymbol{x}_{test}) = \|\boldsymbol{x}_{test} - \boldsymbol{X}\delta_l(\hat{\boldsymbol{\alpha}})\|, \qquad l = 1, 2, \ldots, c, \tag{3.4}$$

where $\delta_l(\hat{\boldsymbol{\alpha}})$ is a characteristic function whose only non-zero entries in $\hat{\boldsymbol{\alpha}}$ corresponding to l-th class training samples. Finally, $\boldsymbol{x}_{test}$ is assigned a class label l corresponding to the class that resulted in the minimal residual.

3.2.3 Support vector machines

Assume each data sample $\boldsymbol{x}_i$ belongs to one of its class label in $y_i \in \{+1, -1\}$ and n denotes the total number of training samples. SVM classifies binary data by solving the following objective function defined as

$$\min_{\omega, \xi_i, b} \left\{ \frac{1}{2}\|\omega\|^2 + \varsigma \sum_{i=1}^{n} \xi_i \right\}, \tag{3.5}$$

subject to the constraint

$$y_i\big(\langle \phi(\omega, \boldsymbol{x}_i)\rangle + b\big) \geq 1 - \xi_i, \tag{3.6}$$

where $\phi(\cdot)$ is a mapping function, ω is normal to the optimal hyperplane, b is the bias term, ς is the regularization parameter which controls the generalization capacity of the machine and $\xi_i \geq 0$ is a slack variable which measures the degree of misclassification of the data.

3.3 Proposed class-dependent sparse representation classifier

3.3.1 Limitations of SRC

Many works have been proposed to enhance the performance of SRC in which they rely on the spatial context of HSI or are based on collaborative representation. In this work, we propose class-dependent sparse representation classifier (cdSRC) to exploit the spectral

content of HSI. In essence, cdSRC effectively combines the ideas of SRC and KNN in a class-wise manner to exploit both correlation and Euclidean distance relationship between test and training samples. Towards this goal, a unified class membership function is developed, that utilizes residual and distance information simultaneously. In doing so, cdSRC not only utilizes the correlation information but also harnesses the Euclidean information which may be lost due to the ℓ_2 normalization (the length of each of atoms in the dictionary is unit length) preprocessing step in the SRC. Experimental results based on several real-world hyperspectral datasets demonstrate that cdSRC can dramatically increase the classification performance of traditional SRC and also outperform kernel based SVM. We also propose an extension of the proposed algorithm in a kernel induced space.

For hyperspectral data classification, pixels from different classes are usually characterized by relatively high correlation with each other and hence makes SRC challenging. This is because the recovered coefficient under such scenarios may potentially be inaccurate. Specifically, the locations of non-zero values in the recovered coefficients may not correspond to the training samples that have the same membership as the true membership of the test sample.

As we have mentioned before, the coefficient vector in Eq. (3.3) can be calculated via a greedy pursuit algorithm such as OMP. Since OMP only exploits correlation (measured by inner products) between a test sample and training samples, it suffers from a limitation in that it does not utilize the Euclidean distance information which can potentially be useful when samples from different classes possess high correlation.

In practical hyperspectral data classification, it is common to encounter problems where samples from two different classes are highly correlated with each other but separated in

the Euclidean space. In this scenario, OMP will be likely select atoms (training samples) whose class membership are different from the test sample which consequently results in classification errors. Fig. 3.1 illustrates this phenomenon based on synthetic data. From this figure, one can easily notice that although the test sample is likely to belong to class-1, it is more correlated with samples from class-2 than class-1, although with respect to the Euclidean distance, the test sample is closer to the samples from class-1 compared with samples from class-2.

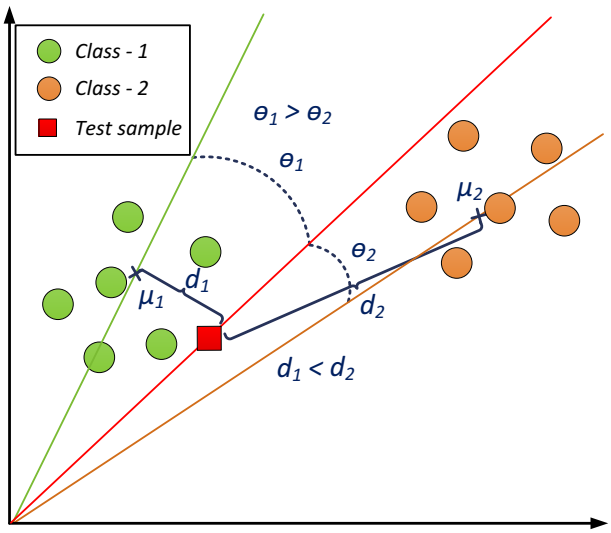


Figure 3.1: Illustrating the case where a test sample is highly correlated with samples from other class but separated in the Euclidean distance. Note that cross symbol denotes means for each class.

Let's explain this phenomenon in another way. It is well known that before solving Eq. (3.3) using OMP, all of atoms in the dictionary need to be ℓ_2 normalized to avoid the bias caused by atoms with varying lengths [26]. However, by performing ℓ_2 normalization, samples having high correlation with each other will be highly overlapped with each other which means the Euclidean distance information is lost or changed. Fig. 3.2 illustrates this phenomenon based on the same synthetic data used in Fig. 3.1. As seen in this figure, highly

correlated samples from two different classes which are well separated in the Euclidean space are completely overlapped after ℓ_2 normalization. In this scenario, OMP is more likely to select atoms from class-2 rather than class-1 which will lead to misclassification. Later, we will see that some real-world hyperspectral datasets also exhibit similar phenomenon.

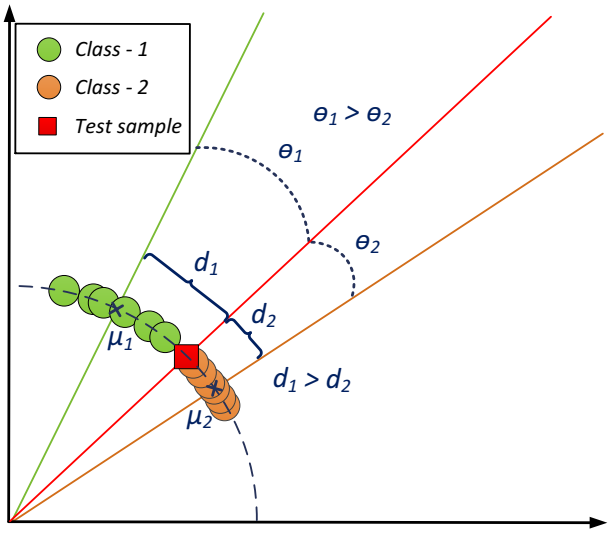


Figure 3.2: Illustrating the case where Euclidean distance information between a test sample and training samples is changed after ℓ_2 normalization.

Another limitation of traditional SRC comes from the fact that it does not incorporate the class label (prior) information of the dataset. It only utilizes the class label information in post processing when calculating the residuals for each class and ignores it when calculating the coefficients. In the supervised or semi-supervised hyperspectral data classification problems, we are given a set of training samples with corresponding labels, although the available number of training samples is very limited, since collecting hyperspectral data is very expensive and time-consuming. Due to the high correlation between samples in HSI, using the entire training dataset as the dictionary for SRC results in atoms potentially being selected from multiple classes. This contradicts the assumption of SRC that the support of a test sample should ideally be in a union of atoms from the same class as the test sample.

3.3.2 Class-dependent sparse representation classifier

To cope with these dilemmas, we propose a cdSRC algorithm to effectively exploit the correlation and Euclidean distance information simultaneously. Different from traditional SRC, the proposed cdSRC is comprised of two components — class-dependent OMP (cdOMP) and class-dependent KNN (cdKNN) which perform the OMP and KNN in a class-wise manner by incorporating the prior (class label) information. In essence, this approach unifies ideas from SRC and traditional classification information obtained via utilizing inter-sample distances. Additionally, by formulating the solver class-wise (i.e., cdOMP), we better condition the SRC. The proposed cdSRC algorithm is described in Algorithm 1. The class label of a test sample is not directly determined in cdOMP or cdKNN. Instead, a residual and Euclidean distance between a test sample and training samples in the dictionary is calculated via cdOMP and cdKNN respectively. Specifically, in cdOMP, the residual for the i-th class is the norm of the difference between the test sample and an approximated test sample derived through OMP using the dictionary formed by training samples from the i-th class. In cdKNN, the i-th class distance is measured by the mean of Euclidean distances of the test sample and its K nearest neighbors. It should be noted that cdOMP is used to sparsely represent the test sample by exploiting the correlation information between the test sample and training samples in every class, while cdKNN is used to exploit the Euclidean distance information. Note that in cdKNN, the data are not ℓ_2 normalized to avoid between-class samples being overlapped with each other, while cdOMP necessitates ℓ_2 normalization [26].

After calculating the residual and distance via cdOMP and cdKNN, a test sample is assigned class label ω via $\omega = \text{argmin}_{i=1,2,\ldots c}(r_i + \lambda d_i)$. Here, λ is a regularization parameter

to balance the relative importance between correlation and Euclidean distance information.

HSI data samples usually reside on manifolds, implying that Euclidean distance in the input space may not be an appropriate to describe inter-sample relationships [10]. Hence we adopt a manifold learning technique that exploits the nonlinear structure of HSI by embedding high-dimensional hyperspectral data into a lower dimensional transformed space where the neighborhood structure of the data is preserved. Various works have been proposed in literature such as ISOMAP, Locally Linear Embedding (LLE), Locality Preserving Projection (LPP) [7] and Local Fisher Discriminant Analysis (LFDA) [8]. In recent work, we have shown that LFDA provides a superior projection of the data into a lower dimensional subspace, where samples from different classes are well separated, and additionally, the local structure of point-clouds of each class is preserved [10, 12]. LFDA can be implemented via effectively combing the properties of Linear Discriminant Analysis (LDA) and LPP. Readers can refer to [8] for a detailed description of LFDA, and to [10, 12] for a description on how to effectively utilize LFDA for hyperspectral classification.

Inspired by these recent observations, we measure the Euclidean distance information used in cdKNN in an LFDA projected subspace such that the local structure of hyperspectral data is effectively captured. Let $\boldsymbol{T} \in \mathbb{R}^{\hat{d} \times d}$ be the LFDA projection matrix, where $\hat{d}$ is the reduced dimensionality. In this work, the Euclidean distance between a test sample $\boldsymbol{x}_{test}$ and training sample $\boldsymbol{x}_{ij}$ in cdKNN is calculated in the LFDA projected space via

$$D(\boldsymbol{x}_{test}, \boldsymbol{x}_{ij}) = \|\boldsymbol{T}\boldsymbol{x}_{test} - \boldsymbol{T}\boldsymbol{x}_{ij}\|_2. \tag{3.7}$$

By projecting the data into the lower dimensional LFDA projected space, we not only increase the inter-class separability but preserve the neighboring distances of samples. This implies that the neighboring samples of a test sample are likely to have the same membership

Algorithm 1 *Class-Dependent Sparse Representation Classifier*

Input: $d \times n$ training data $\boldsymbol{X} = \{\boldsymbol{X}_i\}_{i=1}^{c}$ where $\boldsymbol{X}_i = \{\boldsymbol{x}_{ij}\}_{j=1}^{n_i}$, test data $\boldsymbol{x}_{test} \in \mathbb{R}^d$, sparsity level S, number of nearest neighbors K and regularization parameter λ. Normalize columns of $\boldsymbol{X}$ and $\boldsymbol{x}_{test}$ to have unit norm.

---

for all $i \in 1, 2, \ldots c$ **do**

  Initialization: $\Lambda_i^0 = \emptyset$, $\boldsymbol{r}_i^0 = \boldsymbol{x}_{test}$, and $t = 1$.

  while $t \leq S$ **do**

    Update the support set $\Lambda_i^t = \Lambda_i^{t-1} \cup \lambda_i^t$ by solving $\lambda_i^t = \underset{j=1,2,\ldots,n_i}{\operatorname{argmax}} |\langle \boldsymbol{r}_i^{t-1}, \boldsymbol{x}_{ij} \rangle|$.

    Calculate the coefficient $\boldsymbol{\alpha}_i^t$ over the current support set Λ_i^t by solving

$$\boldsymbol{\alpha}_i^t = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{x}_{test} - (\boldsymbol{X}_i)_{\Lambda_i^t} \boldsymbol{\alpha}\|_2.$$

    Calculate the residual $\boldsymbol{r}_i^t$ by solving $\boldsymbol{r}_i^t = \boldsymbol{x}_{test} - (\boldsymbol{X}_i)_{\Lambda_i^t} \boldsymbol{\alpha}_i^t$.

    $t \leftarrow t + 1$

  end while

  Calculate the i-th class residual $r_i = \|\boldsymbol{r}_i^{t-1}\|_2$.

end for

---

for all $i \in 1, 2, \ldots c$ **do**

  for all $j \in 1, 2, \ldots n_i$ **do**

    Calculate the Euclidean distance D between $\boldsymbol{x}_{test}$ and $\boldsymbol{x}_{ij}$ $d_{ij} = D(\boldsymbol{x}_{test}, \boldsymbol{x}_{ij})$.

  end for

  Calculate the mean of K smallest d_{ij} as the i-th class distance d_i.

end for

Determine the class label of $\boldsymbol{x}_{test}$ based on $\omega = \underset{i=1,2,\ldots c}{\operatorname{argmin}}(r_i + \lambda d_i)$.

Output: A class label ω.

as the test sample and the corresponding cluster of training points is made compact, while the distance between the test sample and training samples whose membership are different from the test sample are increased.

In this section, we provide an argument for using the same sparsity level for each class in cdSRC algorithm. As is described in Algorithm 1, we fix the sparsity level for every class. For signal reconstruction problems, it is reasonable to assume that the sparsity level for different classes should be set individually to faithfully represent the test sample. However, for our cdSRC approach, we contend that the sparsity level for each class should be the same for each class, a motivation for which is provided below.

In HSI, in addition to the samples from the same class being highly correlated with each other, it is possible that samples from different classes are also highly correlated. This can be seen from the 9 class mean correlation matrix of University of Pavia hyperspectral dataset in Fig. 3.3. More details of this dataset can be found in Sec. V. This implies that

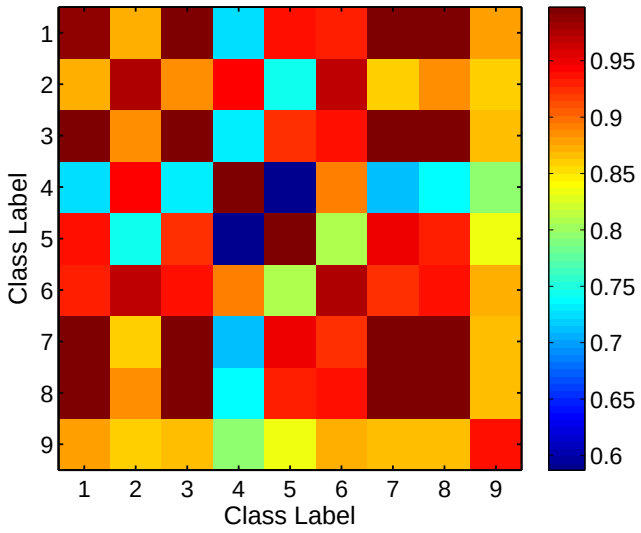


Figure 3.3: Illustrating the 9-class mean correlation matrix of University of Pavia dataset.

a test sample may be faithfully reconstructed using training samples from several different

classes. In other words, the residuals obtained for these classes are similar. In this scenario, it is likely to cause the residual for the class with high sparsity level smaller than the one with low sparsity level. This is due to the fact that in greedy pursuit algorithm such as OMP, the calculated residual monotonically decreases as the sparsity level increases. Hence we want to calculate the residual for every class under the same sparsity level to avoid bias.

We provide additional empirical evidence to illustrate this point. We have chosen the most correlated four classes out of 9 classes in the dataset, namely asphalt, gravel, bitumen and bricks from the University of Pavia dataset. Then we randomly select a test sample from the asphalt class and also select 50 training samples from each of the four classes, while ensuring that the test and training samples do not overlap. Following this, we calculate the residual for every class by gradually increasing the sparsity level. This experiment is repeated 50 times and the average residuals are reported in Fig. 3.4. As can be seen from this figure, for a fixed sparsity level, the residual for the asphalt class is always the smallest among the four classes, which leads to correct classification. But when the sparsity level for the asphalt class is lower than the other classes, the residuals calculated for the other classes may potentially be smaller than the asphalt class, resulting in incorrect classification.

3.3.3 Kernel class-dependent sparse representation classifier

By introducing the kernel trick, cdSRC can be easily extended to the kernelized version of cdSRC (KcdSRC). KcdSRC is especially useful when samples from different classes cannot be linearly distinguished in terms of both correlation and distance information in the input space. By projecting into a reproducing kernel Hilbert space, the underlying discriminant information can be effectively exploited for classification.

Let Φ be the nonlinear function, mapping the data to the kernel feature space and κ be

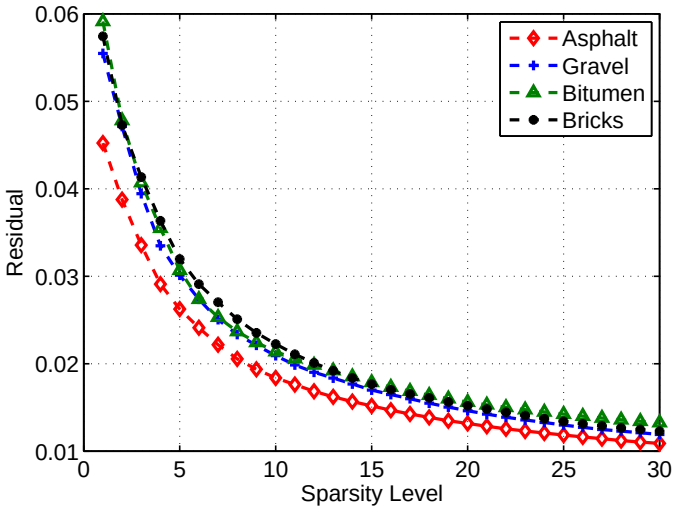


Figure 3.4: Plotting residuals as a function of sparsity level for the most correlated four classes from the University of Pavia dataset.

the corresponding kernel function. A data sample $\boldsymbol{x}$ in the input space $\mathcal{X}$ can be mapped into the feature space $\mathcal{F}$ via $\Phi : \boldsymbol{x} \in \mathcal{X} \rightarrow \Phi(\boldsymbol{x}) \in \mathcal{F}$. By invoking the kernel trick, the inner product of any two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the feature space can be represented as $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \Phi(\boldsymbol{x}_i), \Phi(\boldsymbol{x}_j) \rangle$. Eq. (3.2) can be represented in the feature space as

$$\Phi(\boldsymbol{x}_{test}) = \Phi(\boldsymbol{X})\tilde{\boldsymbol{\alpha}}, \tag{3.8}$$

where $\Phi(\boldsymbol{x}_{test})$ and $\Phi(\boldsymbol{X})$ represent the test and training samples in the feature space, and $\tilde{\boldsymbol{\alpha}}$ is the corresponding coefficient vector. The *sparse coefficient vector* $\tilde{\boldsymbol{\alpha}}$ in $\mathcal{F}$ can be obtained by solving

$$\hat{\boldsymbol{\alpha}} = \text{argmin} \, \|\tilde{\boldsymbol{\alpha}}\|_0, \quad \text{s. t.} \quad \Phi(\boldsymbol{X})\tilde{\boldsymbol{\alpha}} = \Phi(\boldsymbol{x}_{test}). \tag{3.9}$$

The problem posed in Eq. (3.9) can be solved by a "kernelized" variant of OMP (KOMP) [58]. In a manner similar to the cdOMP formulation proposed for cdSRC, we can construct a kernelized variant, KcdOMP, that serves as a class-dependent version of KOMP. The KcdSRC algorithm is described in Algorithm 2.

To capture Euclidean information of data points in $\mathcal{F}$, we have a KcdKNN component within the KcdSRC formulation. The Euclidean distance between a test sample $\boldsymbol{x}_{test}$ and training sample $\boldsymbol{x}_{ij}$ can be calculated in the kernel LFDA (KLFDA) projected feature space via

$$\tilde{D} = \|\tilde{\boldsymbol{T}}\boldsymbol{x}_{test} - \tilde{\boldsymbol{T}}\boldsymbol{x}_{ij}\|_2, \tag{3.10}$$

where $\tilde{\boldsymbol{T}}$ is the projection matrix of KLFDA. The motivation for the choice of KLFDA here is similar to that for using LFDA in cdSRC. KLFDA is an embedding suited for data possessing complex structures, such as data that is on a manifold and essentially implements LFDA in the kernel induced space $\mathcal{F}$. The reader is referred to [8] for further details on KLFDA, and to [59] for a description on the appropriateness of this preprocessing for hyperspectral classification. With these formulations, the remainder of the KcdSRC algorithm is similar to cdSRC.

3.3.4 Experimental results and analysis

In this section, the efficacy of the cdSRC and KcdSRC (as measured by overall classification accuracy) are evaluated as a function of training samples using three different real-world hyperspectral data. Several baseline approaches including SRC, Kernelized SRC (KSRC), CRC, Kernelized CRC (KCRC), KNN, NRS as well as SVM with radial basis function (RBF) as the kernel function are used to compare the efficacy of the proposed algorithms. The kernel functions used in KcdSRC, KSRC and KCRC are all based on the RBF. Each experiment is repeated 10 times using a repeated random sub-sampling validation technique, and the average accuracies are reported in this work. For KcdSRC and SVM based classification, an RBF kernel is used. A one-against-one strategy is used for the multi-class implementation of standard SVM. All free parameters of these algorithms

Algorithm 2 *Kernel Class Dependent Sparse Representation Classifier*

Input: $d \times n$ training data $\boldsymbol{X} = \{\boldsymbol{X}_i\}_{i=1}^{c}$ where $\boldsymbol{X}_i = \{\boldsymbol{x}_{ij}\}_{j=1}^{n_i}$, test data $\boldsymbol{x}_{test} \in \mathbb{R}^d$, kernel function κ, sparsity level S, number of nearest neighbors K and regularization parameter λ.

---

for all $i \in 1, 2, \ldots c$ **do**

  Initialization: Calculate i-th class kernel matrix $\boldsymbol{K}_{\boldsymbol{X}_i} \in \mathbb{R}^{n_i \times n_i}$ whose (j,k)-th entry is $\kappa(\boldsymbol{x}_{ij}, \boldsymbol{x}_{ik})$ and $\boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}} \in \mathbb{R}^{n_i}$ whose j-th entry is $\kappa(\boldsymbol{x}_{ij}, \boldsymbol{x}_{test})$. Set Λ_i^1 to be the index corresponding to the largest entry in $\boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}}$ and iteration counter $t = 2$.

  while $t \leq S$ **do**

    Calculate $\boldsymbol{\beta} \in \mathbb{R}^{n_i}$ via $\boldsymbol{\beta} = \boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}} - (\boldsymbol{K}_{\boldsymbol{X}_i})_{:, \Lambda_i^{t-1}} \Big((\boldsymbol{K}_{\boldsymbol{X}_i})_{\Lambda_i^{t-1}, \Lambda_i^{t-1}} \Big)^{-1} (\boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}})_{\Lambda_i^{t-1}}$.

    Update the support set $\Lambda_i^t = \Lambda_i^{t-1} \cup \lambda_i^t$ by solving $\lambda_i^t = \underset{j=1,2,\ldots n_i}{\operatorname{argmax}} |\boldsymbol{\beta}_j|$.

    $t \leftarrow t + 1$

  end while

  Let $\Lambda_i = \Lambda_i^{t-1}$, the i-th class coefficient $\tilde{\boldsymbol{\alpha}}_{\Lambda_i}$ whose nonzero entries indexed by Λ_i can be calculated via $\tilde{\boldsymbol{\alpha}}_{\Lambda_i} = \Big((\boldsymbol{K}_{\boldsymbol{X}_i})_{\Lambda_i, \Lambda_i} \Big)^{-1} (\boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}})_{\Lambda_i}$.

  Calculate i-th class residual $\tilde{r}_i = \boldsymbol{k}(\boldsymbol{x}_{test}, \boldsymbol{x}_{test}) - 2\tilde{\boldsymbol{\alpha}}_{\Lambda_i}^{'} (\boldsymbol{k}_{\boldsymbol{X}_i, \boldsymbol{x}_{test}})_{\Lambda_i} + \tilde{\boldsymbol{\alpha}}_{\Lambda_i}^{'} (\boldsymbol{K}_{\boldsymbol{X}_i})_{\Lambda_i, \Lambda_i} \tilde{\boldsymbol{\alpha}}_{\Lambda_i}$.

end for

---

for all $i \in 1, 2, \ldots c$ **do**

  for all $j \in 1, 2, \ldots n_i$ **do**

    Calculate the Euclidean distance in the feature space $\tilde{d}_{ij} = \tilde{D}(\boldsymbol{x}_{test}, \boldsymbol{x}_{ij})$.

  end for

  Calculate the mean of K smallest $\tilde{d}_{ij}$ as the i-th class distance $\tilde{d}_i$.

end for

Determine the class label of $\boldsymbol{x}_{test}$ based on $\omega = \underset{i=1,2,\ldots c}{\operatorname{argmin}} (\tilde{r}_i + \lambda \tilde{d}_i)$.

Output: A class label ω.

(e.g., sparsity level, regularization value, parametrization of the kernel function etc.) are determined via cross validation, using training data only.

Since the dictionary used in cdSRC is class dependent (i.e., for c classes, there are c sub-dictionaries), the dictionary size is much smaller than what is commonly encountered in traditional SRC, which is formed based on training samples overall c classes. Together with the high dimensionality of HSI, this results in class-specific sub-dictionaries in cdSRC to often be under-complete when the training sample size is small. To test the premise and performance of cdSRC with both under-complete and over-complete dictionaries, we have tested over a wide range of the number of training samples, varying from 5 samples per class through 150 samples per class. When using 150 samples per class, the dictionaries used in University of Pavia and University of Houston datasets are over-complete, and hence test out the performance under over-complete conditions. For the Indian Pines dataset, the dimensionality of the dataset is very high, and there are classes that do not have enough labeled samples to make the dictionary over-complete. Consequently, this test is only conducted with the University of Pavia and the University of Houston hyperspectral datasets.

The overall classification accuracies of the proposed methods and baseline algorithms for the University of Pavia are summarized in Table 3.1. The "optimal" parameter values for the algorithms (as determined using 50 training samples per class) are as follows. The sparsity levels for cdSRC and KcdSRC, SRC and KSRC are 10, 10, 1, and 30 respectively. The dimensionality of the LFDA/KLFDA feature space in cdSRC/KcdSRC is 30. The λ values of cdSRC, KcdSRC and NRS are 0.05 and 0.001 and 0.00001 respectively. The sigma values in the RBF kernel function of KcdSRC, KSRC, KCRC and SVM are 2, 0.1, 0.1

and 0.05 respectively. It can be observed from the results that the proposed cdSRC and KcdSRC outperform all other baseline approaches. We also note that cdSRC (and KcdSRC) consistently outperform baseline approaches with both under-complete and over-complete dictionaries. The proposed methods have a performance that is comparable to SVM when using very limited training data (5 and 10 samples per class). Other than that, in general, the proposed methods outperform SVM.

Table 3.1: Overall classification accuracies (%) and standard deviation (in bracket) as a function of the number of training samples per class for University of Pavia dataset.

	Number of training samples per class					
Algorithm	***5***	***10***	***30***	***50***	***100***	***150***
cdSRC	73.1 (2.7)	78.9 (2.4)	86.0 (1.0)	87.6 (0.9)	89.0 (0.8)	91.1 (0.8)
KcdSRC	72.8 (2.9)	79.8 (2.1)	87.6 (0.9)	89.6 (0.8)	91.4 (0.3)	92.3 (0.5)
SRC	68.0 (1.5)	70.6 (1.2)	75.3 (1.6)	77.3 (1.0)	79.0 (0.8)	80.4 (0.4)
KSRC	69.2 (1.1)	73.5 (1.9)	79.7 (1.8)	81.6 (1.1)	84.3 (0.8)	85.5 (0.5)
NRS	70.9 (2.4)	75.5 (1.8)	82.3 (2.3)	82.0 (2.4)	82.3 (1.4)	80.9 (2.2)
CRC	64.9 (3.6)	66.0 (4.2)	70.6 (0.6)	71.6 (0.9)	72.0 (0.8)	72.8 (0.5)
KCRC	69.6 (1.4)	72.2 (4.6)	79.8 (1.0)	80.5 (0.8)	82.9 (0.9)	83.9 (0.6)
KNN	70.4 (2.1)	73.0 (2.3)	78.2 (1.1)	80.1 (1.0)	82.6 (0.4)	84.0 (0.8)
SVM	73.5 (3.0)	79.7 (1.4)	85.8 (1.1)	87.4 (0.9)	88.9 (1.2)	90.4 (0.4)

To illustrate the impact of the regularization parameter λ for the performance of cdSRC, we first look at how the ℓ_2 normalization will affect the mean spectral signatures of samples from different classes. In University of Pavia data, four different classes such as Asphalt, Gravel, Bitumen and Bricks are highly correlated with each other but slightly separated in the ℓ_2 norm (Euclidean distance) sense. After ℓ_2 normalization which is shown in Fig. 3.5, these four classes are almost completely overlapped with each other. It makes classifying these four classes very challenging. Hence, in Table 3.1, we note that the traditional SRC and CRC give relatively low classification accuracies compared with the proposed methods which is designed to handle these overlapping.

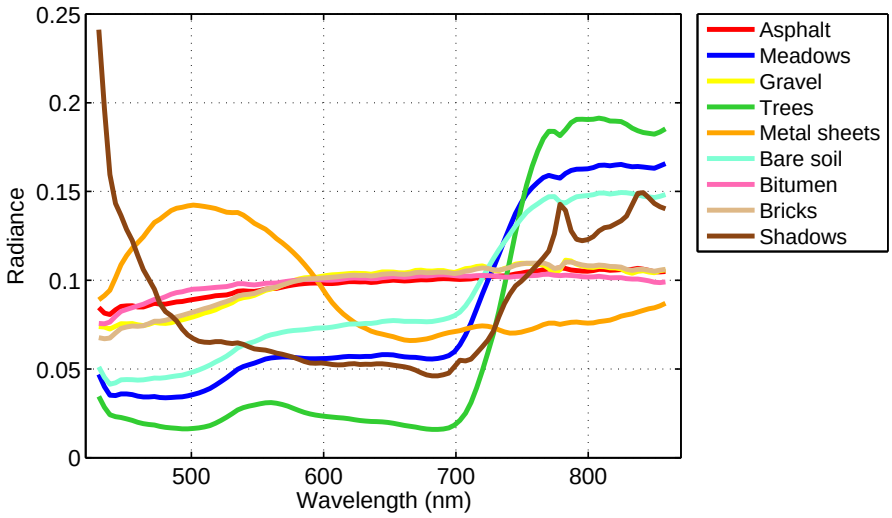


Figure 3.5: Mean spectral signatures of University of Pavia dataset after ℓ_2 normalization.

Fig. 3.6 shows the overall classification accuracies versus different values of λ. If λ value is set to 0, it means only correlation information is exploited during the classification stage. One can notice that as the λ value increases, the classification accuracy also increases until the λ value reaches around 0.06. Also, as the λ value keeps increasing, the accuracy drops. It is expected since the correlation information is gradually excluded and Euclidean distance information is dominated. Hence by properly adding the Euclidean distance information using cdKNN will improve the classification performance of cdSRC.

Next, we demonstrate the effect of sparsity level S and number of nearest neighbors K on the performance of cdSRC which is depicted in Fig. 3.7 as a mesh plot. By looking at this figure, the high classification accuracies are obtained when both S and K are small.

The classification maps of University of Pavia generated using the proposed methods and baseline algorithms are shown in Fig. 3.8 to test the generalization capability of these methods. 30 training samples per class are used in this experiment. It can be seen from Fig. 3.8 that the proposed methods (cdSRC/KcdSRC) result in more accurate and "smoother"

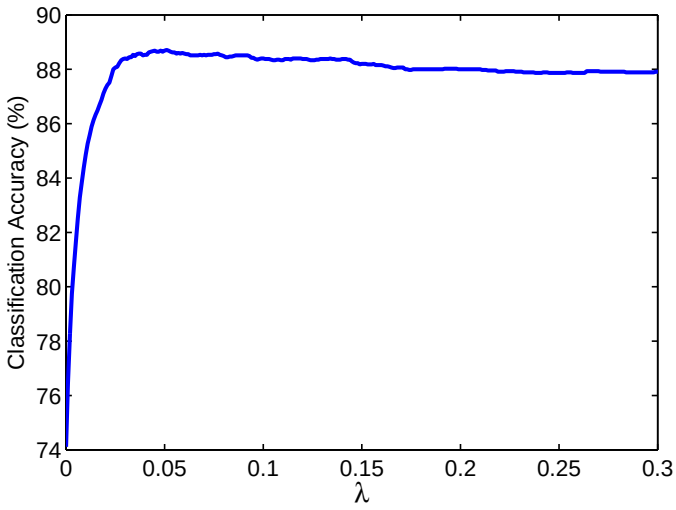


Figure 3.6: Overall classification accuracies (%) versus different values of λ for University of Pavia dataset. 50 number of training samples per class are employed in this experiment.

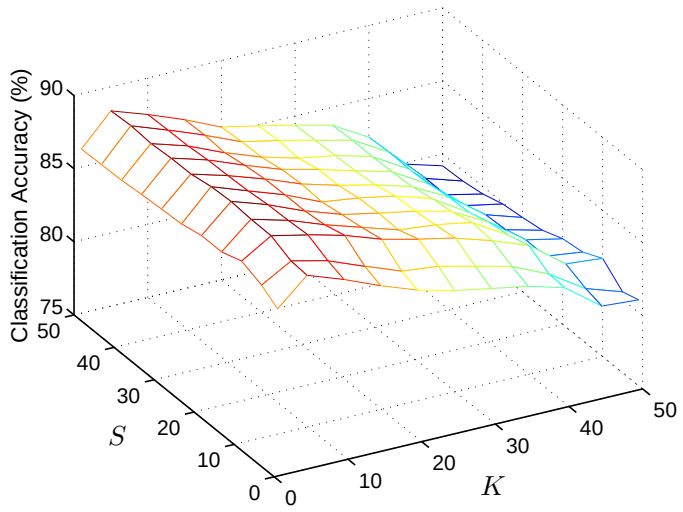


Figure 3.7: Overall classification accuracies versus sparsity level S and number of nearest neighbor K for University of Pavia dataset. 50 training samples per class are employed in this experiment.

classification maps (with reduced salt-and-pepper classification noise) compared with traditional SRC/KSRC/CRC/KCRC, and they have performance that is visually comparable to NRS and SVM — KcdSRC provides better classification for pixels corresponding to the Brick class, while SVM yields superior performance for pixels corresponding to the Meadows class. The actual improvements are quantified by accuracies discussed previously.

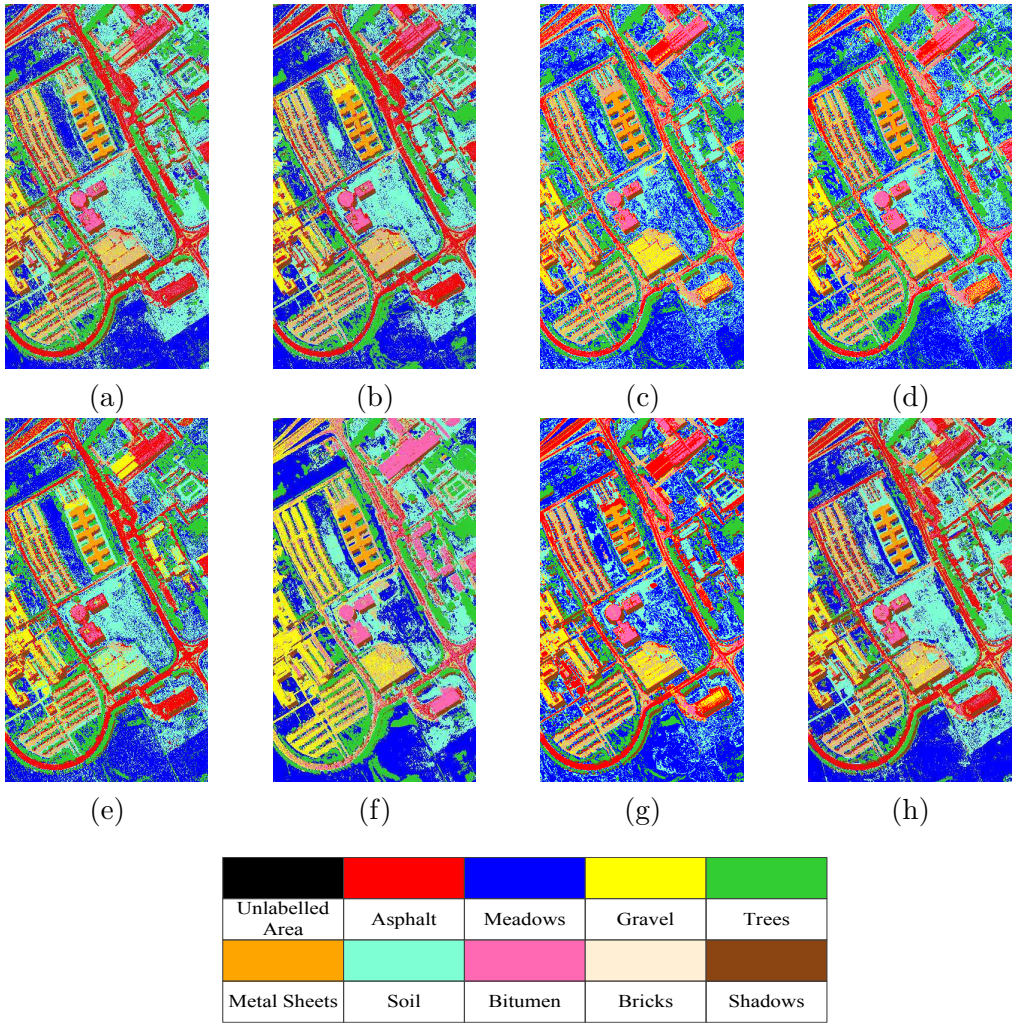


Figure 3.8: Classification maps of University of Pavia dataset generated using (a) cdSRC (b) KcdSRC (c) SRC (d) KSRC (e) NRS (f) CRC (g) KCRC (h) SVM.

The overall classification results for the Indian Pines data are summarized in Table 3.2.

The "optimal" parameter values for the algorithms (as determined using 50 training samples per class) are as follows. The sparsity levels of cdSRC and KcdSRC, SRC and KSRC are 10, 15, 12, and 45 respectively. The reduced dimensionality values of LFDA/KLFDA in cdSRC/KcdSRC are both 30. The λ values of cdSRC, KcdSRC and NRS are 0.05 and 0.00005 and 0.00005 respectively. The sigma values in the RBF kernel function of KcdSRC, KSRC, KCRC and SVM are 2, 0.1, 0.1 and 0.01 respectively. Based on the result in Table 3.2, we can easily see that the classification performance of cdSRC and KcdSRC are considerably and consistently better than other baseline algorithms under the different training sample sizes. Even with a small number of training samples (5 and 10 per class), the proposed methods have a much better classification performance than SVM.

Table 3.2: Overall classification accuracies (%) and standard deviation (in bracket) as a function of the number of training samples per class for Indian Pines dataset.

	Number of training samples per class					
Algorithm	***5***	***10***	***30***	***50***	***100***	***150***
cdSRC	73.1 (1.6)	81.6 (1.7)	89.4 (0.9)	92.1 (0.5)	94.2 (0.4)	95.0 (0.3)
KcdSRC	73.2 (3.1)	81.6 (2.1)	90.3 (0.7)	92.5 (0.4)	94.4 (0.3)	95.2 (0.4)
SRC	63.5 (1.6)	67.7 (2.0)	77.3 (2.0)	80.9 (0.9)	84.8 (0.3)	86.7 (0.3)
KSRC	67.8 (2.6)	75.3 (1.8)	84.5 (1.0)	87.9 (0.9)	91.2 (0.2)	92.7 (0.3)
NRS	64.0 (4.1)	79.4 (1.5)	87.8 (1.0)	89.8 (0.7)	90.2 (1.1)	89.8 (1.1)
CRC	65.3 (1.9)	70.8 (2.3)	77.9 (1.1)	80.6 (0.8)	82.9 (0.9)	84.1 (0.3)
KCRC	68.2 (2.2)	75.4 (3.0)	85.9 (0.8)	88.7 (0.5)	91.4 (0.3)	93.0 (0.4)
KNN	57.5 (1.7)	62.3 (0.7)	71.2 (0.9)	74.5 (0.7)	79.2 (0.4)	81.4 (0.4)
SVM	64.6 (2.4)	75.5 (1.6)	86.9 (1.0)	89.9 (0.5)	93.0 (0.2)	94.0 (0.3)

Fig. 3.9 illustrates the influence of λ on the performance of cdSRC. Again by gradually increasing the λ value (adding the Euclidean distance information), the classification performance of cdSRC increases. However, it degrades the performance of cdSRC if too much Euclidean information is included.

The overall classification accuracies versus sparsity level S and number of nearest neighbors K is depicted in Fig. 3.10 as a mesh plot. Slightly different with the University of Pavia data, the optimal performance of cdSRC is achieved at the high values of S and K.

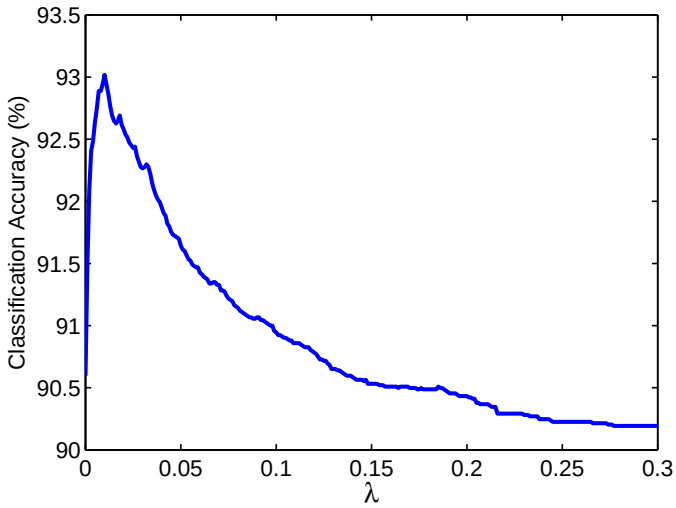


Figure 3.9: Overall classification accuracies (%) versus differet values of λ for Indian Pines dataset.

Fig. 3.11 shows the classification maps obtained using the proposed and baseline algorithms which is based on the 30 training samples per class. The proposed cdSRC and KcdSRC have a better or comparable performance compared with SVM. Specifically, KcdSRC and SVM correctly classify most of pixels in Soybean-middle class residing in the top-right corner of the map, while other methods have a poor performance for these pixels. Also, KcdSRC provides the best overall classification performance of pixels in the Corn-middle class residing in the top-right corner of the map, while other methods either fail or have a poor performance.

The classification result for the University of Houston dataset is shown in Table 3.3. The "optimal" parameter values for the algorithms (as determined using 50 training samples per class) are as follows. The sparsity levels of cdSRC and KcdSRC, SRC and KSRC

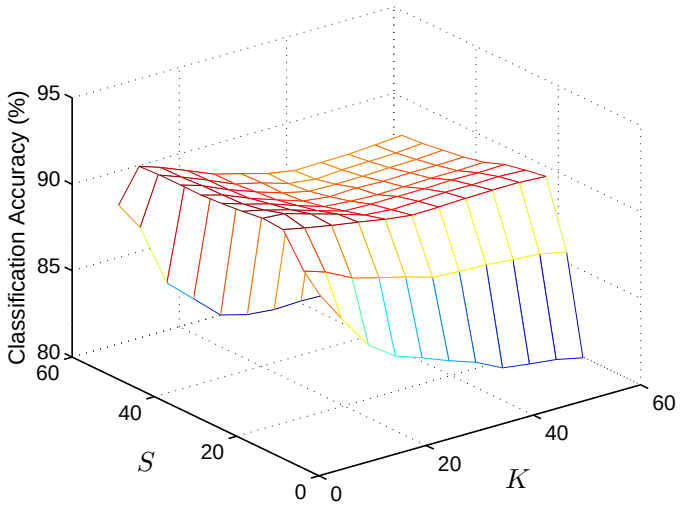


Figure 3.10: Overall classification accuracies versus sparsity level S and number of nearest neighbors K for Indian Pines dataset. 50 training samples per class are employed in this experiment.

are 10, 10, 10, and 50 respectively. The reduced dimensionality values of LFDA/KLFDA in cdSRC/KcdSRC are both 30. The λ values of cdSRC, KcdSRC and NRS are 0.01 and 0.00005 and 0.0005 respectively. The sigma values in the RBF kernel function of KcdSRC, KSRC, KCRC and SVM are 3, 0.1, 0.1 and 0.01 respectively. It is obvious from Table 3.3 that the proposed algorithms yield higher classification accuracies than any other baseline algorithms. As with the University of Houston dataset, we observe that cdSRC (and KcdSRC) consistently outperform baseline approaches with both under-complete and over-complete dictionaries.

The overall classification accuracies versus λ value is plotted in Fig. 3.12. Fig. 3.13 illustrates the overall classification accuracy versus sparsity level S and number of nearest neighbors K.

The classification maps of University of Houston data are shown in Fig. 3.14j, which are generated using the proposed algorithms as well as baselines. The number of training

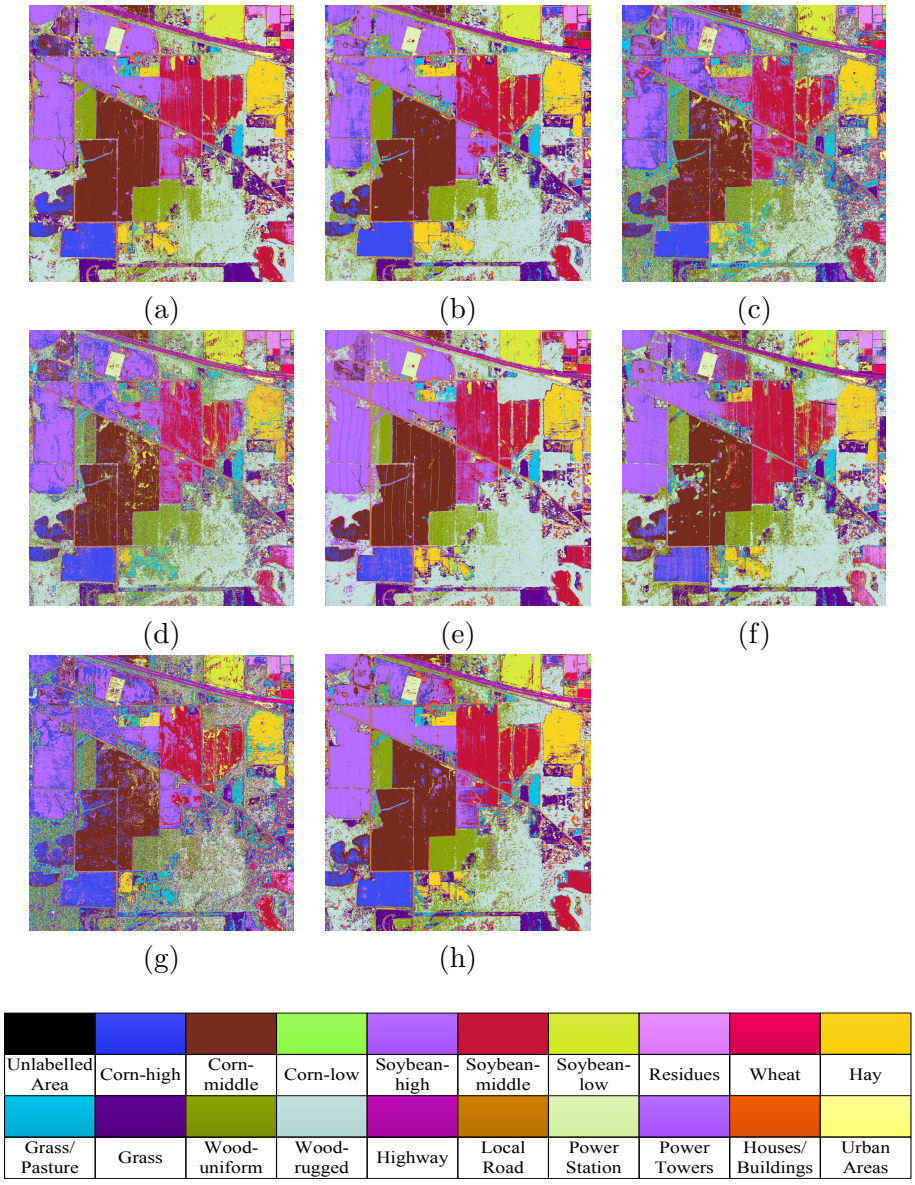


Figure 3.11: Classification maps of Indian Pines dataset generated using (a) cdSRC (b) KcdSRC (c) SRC (d) KSRC (e) NRS (f) CRC (g) KCRC (h) SVM.

Table 3.3: Overall classification accuracies (%) and standard deviation (in bracket) as a function of the number of training samples per class for University of Houston dataset.

	Number of training samples per class					
Algorithm	***5***	***10***	***30***	***50***	***100***	***150***
cdSRC	67.6 (2.8)	73.3 (2.8)	79.1 (1.8)	82.1 (0.8)	82.6 (0.7)	83.0 (0.2)
KcdSRC	67.9 (2.3)	73.0 (2.0)	80.0 (1.1)	82.0 (0.4)	82.9 (0.5)	82.8 (0.3)
SRC	59.4 (1.0)	62.8 (2.0)	68.4 (1.7)	71.0 (0.8)	73.0 (0.7)	73.7 (0.4)
KSRC	62.6 (3.1)	68.2 (1.3)	73.0 (2.4)	76.3 (0.7)	77.4 (0.6)	78.1 (0.3)
NRS	62.0 (7.6)	68.9 (2.0)	74.2 (0.6)	75.3 (1.7)	74.3 (1.4)	71.7 (1.1)
CRC	60.2 (2.8)	64.6 (2.6)	67.4 (1.2)	68.5 (0.6)	69.1 (0.5)	69.2 (0.4)
KCRC	63.7 (1.5)	66.7 (3.6)	74.2 (0.7)	74.7 (0.6)	77.0 (0.5)	77.7 (0.4)
KNN	59.1 (2.7)	62.6 (1.8)	68.1 (1.3)	70.1 (0.4)	71.2 (0.4)	72.4 (0.3)
SVM	65.4 (2.1)	69.7 (1.9)	76.4 (1.4)	78.1 (1.2)	78.6 (0.6)	79.2 (0.4)

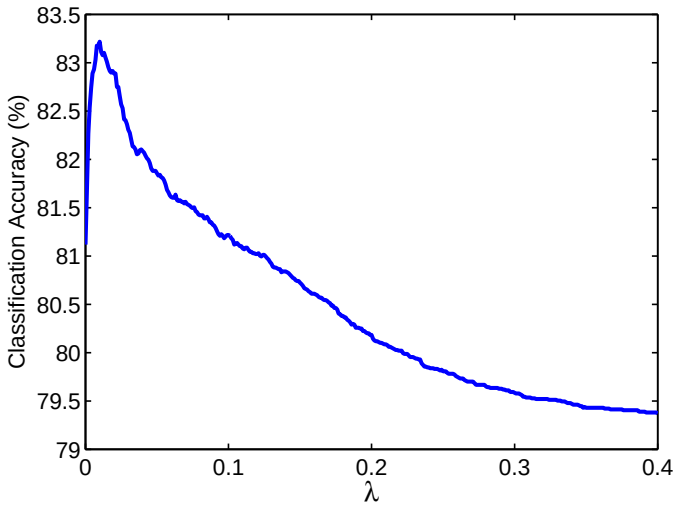


Figure 3.12: Overall classification accuracies (%) versus different values of λ for University of Houston dataset.

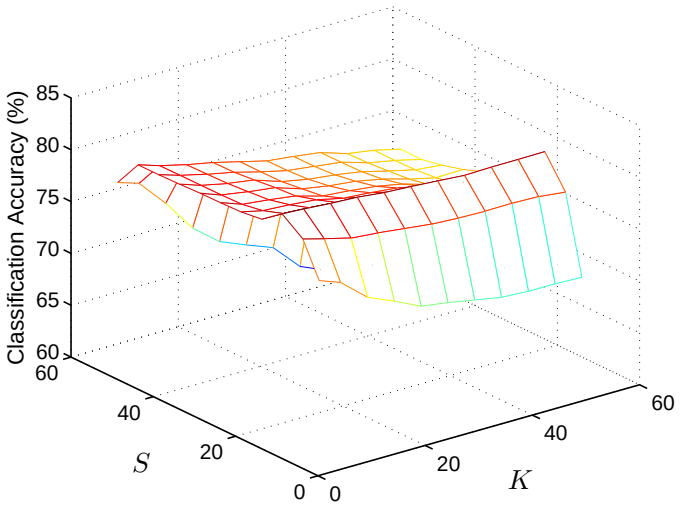


Figure 3.13: Overall classification accuracies versus sparsity level S and number of nearest neighbor K for University of Houston dataset. 50 training samples per class are employed in this experiment.

samples per class used here is 30. From the classification maps in Fig. 3.14, we can see that all methods exhibit poor classification performance under the cloud shadow due to no training sample is selected in that area. The proposed methods have comparable (and for some classes, better) performance compared with other baseline methods in cloud-free areas.

To test the computational complexity of the proposed cdSRC and KcdSRC approaches, we compare it with standard SRC/KSRC. We choose the University of Pavia dataset to perform this test. The total number of test samples used here is 100, and the sparsity level for all of these four algorithms are set to 10 for fair comparison. All the experiments are based on a single desktop PC, 8-core, 2.14GHz, using Matlab R2012a. Based on the test, the CPU times (milliseconds) for cdSRC and KcdSRC are 2.69 and 2.41 respectively, and 0.72 and 0.63 for SRC and KSRC which are approximately 4 times faster than the proposed approaches. This is primarily due to the class dependent structure of cdSRC. However, the

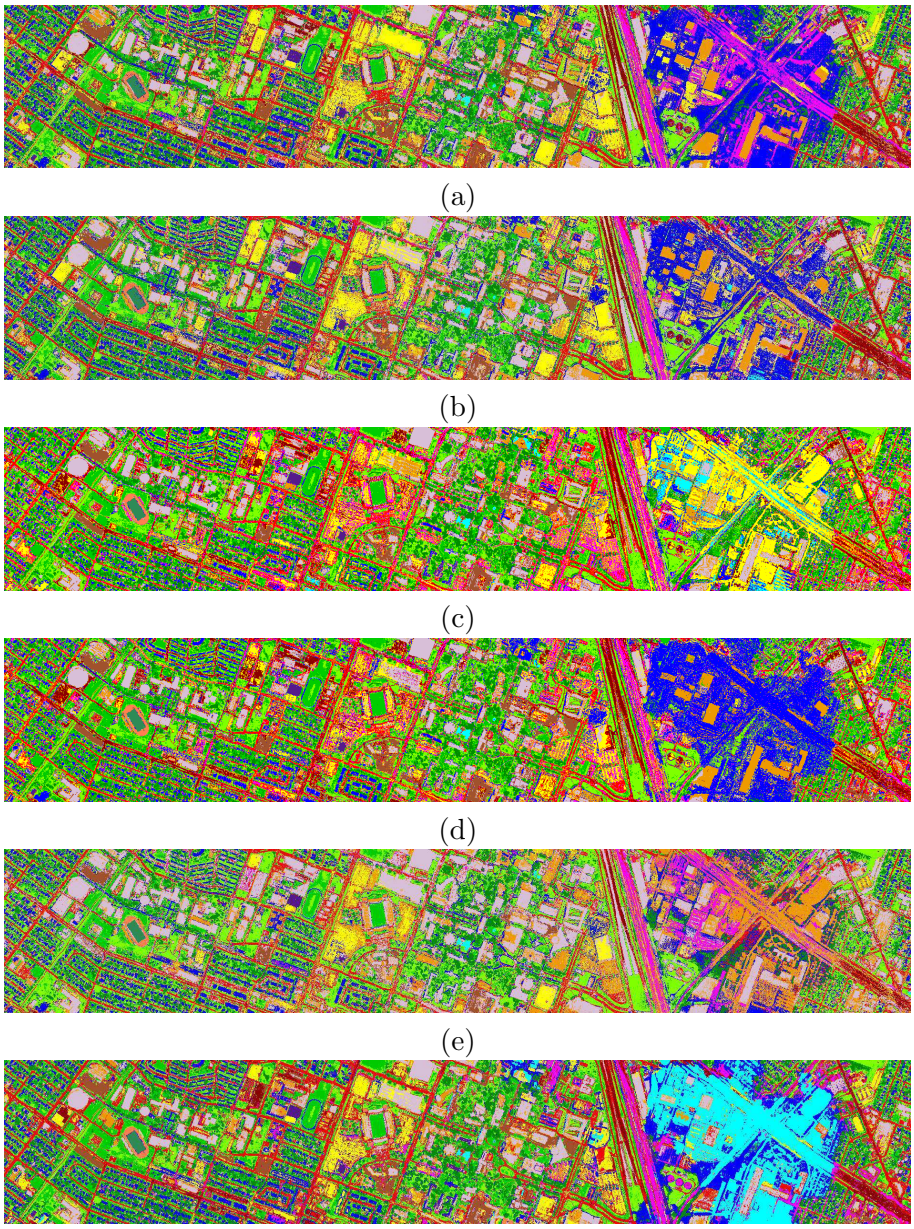


Figure 3.14: Classification maps of University of Houston dataset generated (a) cdSRC (b) KcdSRC (c) SRC (d) KSRC (e) NRS (f) SVM.

proposed approaches can be easily implemented in a distributed manner for efficiency by calculating the residual and distance for each class simultaneously.

3.4 Proposed class-dependent orthogonal least square

3.4.1 Difference between OMP and OLS

Before we describe our proposed method, we first review the difference between OMP and OLS in detail. Although there are several different ways to solve this sparse approximation problem in (3.3), we focus on greedy pursuit based approaches including OMP and OLS in this work. For other approaches such as convex relaxation and Bayesian based framework etc., we refer readers to these works [45, 57, 60]. Both OMP and OLS can be used to approximate the sparsest solution in (3.3). In each iteration, the atom selected by OMP is not designed to minimize the residual norm after projecting the target signal onto the selected elements, while OLS selects the atom that minimizes the residual based on the previously selected atoms. Thus the final residual norm generated by OLS is always smaller than OMP under similar conditions. However, OLS does not always give the sparsest solution. To find an optimal S-term representation of an signal $\boldsymbol{x}$ in (3.3), a simple approach to finding the sparsest solution then is to search over all possible linear combinations of S atoms in $\boldsymbol{A}$. Let us denote this exhaustive searching algorithm as combinatorial orthogonal least square (COLS).

We use an intuitive example to illustrate the differences of OMP, OLS and COLS algorithms. In [46], the authors use a graphical interpretation to show the difference between OMP and OLS in terms of atom selection procedure. In this example, we will further illustrate that the norm of residual generated by OLS is smaller than OMP but they are both

not optimal. We will demonstrate later that the signal reconstruction performance of OLS is close to optimal. Assume the true sparsity level in (3.3) is S. Let $\boldsymbol{z}_1$, $\boldsymbol{z}_2$ and $\boldsymbol{z}_3$ be the axes in a 3-dimensional space, and $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, $\boldsymbol{a}_3$ be the atoms in a dictionary D. Without loss of generality, assume $\boldsymbol{a}_1$ and $\boldsymbol{z}_1$ are overlapped with each other, and $\boldsymbol{a}_2$ and $\boldsymbol{a}_3$ are in the $\boldsymbol{z}_1\boldsymbol{z}_2$-plane and $\boldsymbol{z}_1\boldsymbol{z}_3$-plane respectively. Let $\boldsymbol{x}$ be a target signal, and assume that $\boldsymbol{a}_1$ is the most correlated with $\boldsymbol{x}$ than $\boldsymbol{a}_2$ and $\boldsymbol{a}_3$. Let $\vec{OF} = \vec{AD}$. Let ϕ_1 and ϕ_2 be the angles between $\boldsymbol{a}_2$ and $\vec{OF}$, and $\boldsymbol{a}_3$ and $\vec{OF}$ respectively. Under this scenario, we will analyze the optimal sparse S-term representation using OMP, OLS and COLS, where S equals to 2. 1) OMP first selects the most correlated atom which is $\boldsymbol{a}_1$, and produces the residual $\vec{AD}$ by projecting $\boldsymbol{x}$ onto it. Next, OMP selects an atom that is mostly correlated with $\vec{AD}$. Since $\vec{OF} = \vec{AD}$ and $\phi_1 < \phi_2$, OMP selects $\boldsymbol{a}_2$. Therefore, the final residual norm produced by OMP is $\|\vec{AB}\|_2$, which is obtained by projecting $\boldsymbol{x}$ onto $\boldsymbol{a}_1\boldsymbol{a}_2$-plane. 2) For OLS, the first atom selected is $\boldsymbol{a}_1$, since OMP and OLS are the same in the first iteration. Next, OLS calculates the residual norms of $\|\vec{AC}\|_2$ and $\|\vec{AB}\|_2$ obtained by projecting $\boldsymbol{x}$ onto $\boldsymbol{a}_1\boldsymbol{a}_3$-plane and $\boldsymbol{a}_1\boldsymbol{a}_2$-plane respectively, and selects $\boldsymbol{a}_3$, since $\|\vec{AC}\|_2 < \|\vec{AB}\|_2$. Thus, the final residual norm of OLS is $\|\vec{AC}\|_2$ obtained by projecting $\boldsymbol{x}$ onto $\boldsymbol{z}_1\boldsymbol{z}_3$-plane. 3) COLS calculates all residuals by projecting $\boldsymbol{x}$ onto planes formed by every combination of two atoms. Since $\|\vec{AE}\|_2 < \|\vec{AC}\|_2 < \|\vec{AB}\|_2$, COLS selects $\boldsymbol{a}_2$ and $\boldsymbol{a}_3$. The final residual norm is $\|\vec{AE}\|_2$. For the special case when D is an orthonormal dictionary, all of the above three methods will find an optimal S-term representation [61]. Overall, the performance of these methods with regard to the reconstruction error are COLS $\geq$ OLS $\geq$ OMP.

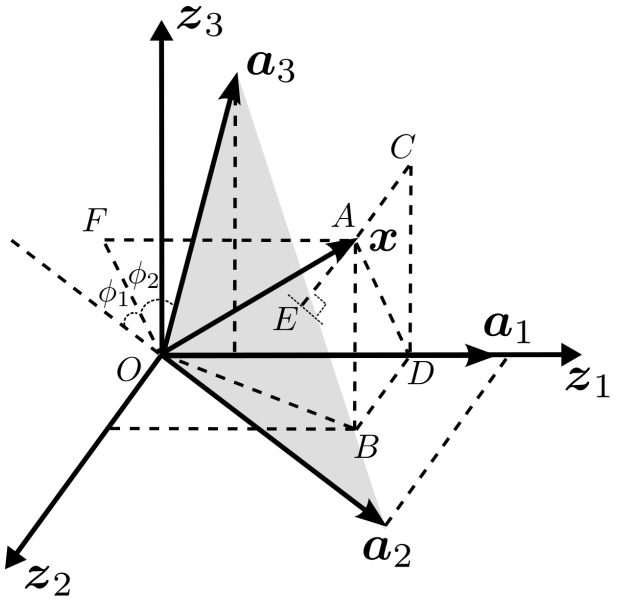


Figure 3.15: Graphically illustrating OMP, OLS and COLS.

3.4.2 Class-dependent orthogonal least square

The recent work in [54] demonstrates that operating SRC in a class-wise manner can significantly improve the classification performance of SRC. As is explained in the previous section, the recovery ability of OLS is always better than OMP in terms of the least square error under the same condition (i.e. the same sparsity level). Therefore, it is expected that the classification performance can be significantly enhanced by replacing OMP with OLS under this framework. We name this algorithm class-dependent OLS (cdOLS). Note that the stopping criterion in cdOLS is based on the sparsity level. This is because the signal estimation error monotonically decreases as the sparsity level increases. Hence, we use the same sparsity level for each class to circumvent this bias. We also extend cdOLS to a "kernel" cdOLS (KcdOLS). The cdOLS and KcdOLS algorithms are described in Algorithm 3 and Algorithm 4 respectively. For a faster implementation of OLS, readers can refer to [46].

Algorithm 3 *cdOLS*

Input: A training dataset $\boldsymbol{A} \in \{\boldsymbol{A}_l\}_{l=1}^{c} \in \mathbb{R}^{d \times n}$, test sample $\boldsymbol{x} \in \mathbb{R}^d$ and sparsity level S.

---

for all $l \in 1, 2, \ldots, c$ **do**

  Set $\Lambda^0 = \emptyset$, $\boldsymbol{r}^0 = \boldsymbol{y}$, and iteration counter $m = 1$.

  while $m \leq S$ **do**

    Update the support set $\Lambda^m = \Lambda^{m-1} \cup \lambda^m$ by solving

$$\lambda^m = \underset{j=1,2,\ldots,n}{\operatorname{argmin}} \|\boldsymbol{x} - (\boldsymbol{A}_l)_{:,\Lambda^{m-1} \cup j} \tilde{\boldsymbol{\beta}}\|_2,$$

    where $\tilde{\boldsymbol{\beta}} = (\boldsymbol{A}_l^{\dagger})_{:,\Lambda^{m-1} \cup j} \boldsymbol{x}$.

    Calculate the residual $\boldsymbol{r}^m$ by solving

$$\boldsymbol{r}^m = \boldsymbol{x} - \boldsymbol{A}_{:,\Lambda^m} \hat{\boldsymbol{\beta}},$$

    where $\hat{\boldsymbol{\beta}} = (\boldsymbol{A}_l^{\dagger})_{:,\Lambda^m} \boldsymbol{x}$.

    $m \leftarrow m + 1$

  end while

  Calculate the l-th class residual norm $\nu_l = \|\boldsymbol{r}^{m-1}\|_2$.

end for

Class label of $\boldsymbol{x}$: $\omega = \underset{l=1,2,\ldots,c}{\operatorname{argmin}} \nu_l$.

Output: A class label ω.

Algorithm 4 *KcdOLS*

Input: A training dataset $\boldsymbol{A} = \{\boldsymbol{A}_l\}_{l=1}^{c} \in \mathbb{R}^{d\times n}$, where $\boldsymbol{A}_l = \{\boldsymbol{a}_{li}\}_{i=1}^{n_l} \in \mathbb{R}^{d\times n_l}$, test sample $\boldsymbol{x} \in \mathbb{R}^d$, kernel function κ, sparsity level S.

---

for all $l \in 1, 2, \ldots, c$ **do**

Calculate l-th class kernel matrix $\boldsymbol{K}_l \in \mathbb{R}^{n_l\times n_l}$ whose (i,j)-th entry is $\kappa(\boldsymbol{a}_{li}, \boldsymbol{a}_{lj})$ and $\boldsymbol{k}_l \in \mathbb{R}^{n_l}$ whose i-th entry is $\kappa(\boldsymbol{x}, \boldsymbol{a}_{li})$. Set index set Λ^1 to be the index corresponding to the largest entry in $\boldsymbol{k}_l$ and iteration counter $m = 2$.

while $m \leq S$ **do**

Update the support set $\Lambda^m = \Lambda^{m-1} \cup \lambda^m$ by solving

$$\lambda^m = \underset{j\in 1,2,\ldots,n}{\operatorname{argmin}} (\kappa(\boldsymbol{x},\boldsymbol{x}) \; - \; 2(\boldsymbol{k}_l^\top)_{\Lambda^{m-1}\cup j}\tilde{\boldsymbol{\beta}} \; + \; \tilde{\boldsymbol{\beta}}^\top(\boldsymbol{K}_l)_{\Lambda^{m-1}\cup j,\Lambda^{m-1}\cup j}\tilde{\boldsymbol{\beta}}),$$

where $\tilde{\boldsymbol{\beta}} = ((\boldsymbol{K}_l)_{\Lambda^{m-1}\cup j,\Lambda^{m-1}\cup j})^{-1}(\boldsymbol{k}_l)_{\Lambda^{m-1}\cup j}$.

$m \leftarrow m + 1$

end while

The l-th class residual norm can be calculated via

$$\nu_l = \sqrt{\kappa(\boldsymbol{y},\boldsymbol{y}) - 2(\hat{\boldsymbol{\beta}})^\top(\boldsymbol{k}_l)_{\Lambda^{m-1}} + (\hat{\boldsymbol{\beta}})^\top(\boldsymbol{K}_l)_{\Lambda^{m-1},\Lambda^{m-1}}\hat{\boldsymbol{\beta}}},$$

where $\hat{\boldsymbol{\beta}} = \left((\boldsymbol{K}_l)_{\Lambda^m,\Lambda^m}\right)^{-1}(\boldsymbol{k}_l)_{\Lambda^m}$.

end for

Class label of $\boldsymbol{x}$: $\omega = \underset{l=1,2,\ldots,c}{\operatorname{argmin}} \nu_l$.

Output: A class label ω.

3.4.3 Experimental results and analysis

The classification results based on the University of Pavia and University of Houston datasets are presented in Table 3.4 and Table 3.5 respectively. To evaluate the classification performance of cdOLS and KcdOLS, several baseline approaches including SRC, kernel SRC (KSRC), class-dependent OMP (cdOMP), kernel cdOMP (KcdOMP), and nonlinear support vector machine (SVM) are compared. For SRC (KSRC), we use OMP (KOMP) as the recovery method for fair comparison, although convex optimization-based approaches generally outperform greedy-based approaches. Additionally, we also implement the COLS in a class-wise manner (cdCOLS) as well as its kernel version KcdCOLS — these COLS based variants can be considered as upper bounds in performance of OLS based methods. The kernel functions used in these kernel-based methods was the radial basis function (RBF). The optimal parameters including sparsity level and kernel parameter in RBF are determined via cross-validation.

The classification results for these two datasets are presented in Table 3.4 and Table 3.5 respectively. As expected, we observe that the higher the reconstruction accuracy, the better the classification result. Since COLS is a combinatorial searching method, it is practically unfeasible, particularly when the dictionary size is large. We add it as a comparative method in this work in order to compare the performance gap between cdOLS and cdCOLS. We note that cdCOLS may be feasible in scenarios where the dictionary size is small, and so is the underlying sparsity level for the representations. The overall performance of cdCOLS and cdOLS are similar with a slightly better performance for cdCOLS (as expected). The average performance of cdOLS is generally better than cdOMP.

To analyze the effect of sparsity level, we evaluate the performance of cdCOLS, cdOLS

Table 3.4: Classification accuracy (%) and standard deviation (in bracket) as a function of training sample size per class for University of Pavia data.

Algorithm / Sample Size	*10*	*30*	*50*
KcdCOLS	79.6 (1.5)	86.8 (0.9)	88.4 (0.8)
cdCOLS	73.9 (2.1)	79.3 (1.4)	81.0 (0.8)
KcdOLS	79.6 (1.5)	85.7 (0.9)	87.6 (0.5)
cdOLS	73.3 (1.8)	77.4 (1.2)	78.9 (1.1)
KcdOMP	79.6 (1.5)	85.8 (1.1)	87.3 (0.6)
cdOMP	73.0 (1.8)	76.3 (1.8)	77.8 (1.4)
KSRC	79.7 (1.4)	84.6 (0.9)	86.4 (0.6)
SRC	70.8 (1.0)	75.1 (1.3)	77.8 (1.4)
SVM	79.1 (1.6)	85.8 (0.6)	87.9 (0.8)

and cdOMP under the different sparsity levels. Fig. 3.16 and Fig. 3.17 show the classification accuracy as a function of sparsity level for University of Pavia and University of Houston data respectively. The number of samples per class in this experiment is set to 30. Hence we test the sparsity level starting from 1 to the highest possible number 30. From these two figures, we notice that the optimal sparsity level for these methods are generally very low. This is due to the fact that the within-class hyperspectral data samples are very correlated with each other, and a low residual norm can be derived using a small number of atoms.

Table 3.5: Classification accuracy (%) and standard deviation (in bracket) as a function of training sample size per class for University of Houston data.

Algorithm / Sample Size	*10*	*30*	*50*
KcdCOLS	85.8 (1.7)	95.2 (0.6)	97.3 (0.2)
cdCOLS	84.6 (1.3)	93.4 (0.6)	96.3 (0.4)
KcdOLS	85.7 (1.7)	94.8 (0.7)	97.2 (0.3)
cdOLS	84.5 (1.5)	92.9 (0.9)	95.9 (0.6)
KcdOMP	82.4 (1.3)	89.6 (0.8)	92.5 (0.5)
cdOMP	79.7 (1.2)	87.1 (0.6)	91.8 (0.5)
KSRC	80.0 (0.9)	87.6 (0.7)	91.8 (0.6)
SRC	78.7 (0.8)	88.5 (0.5)	92.2 (0.6)
SVM	79.1 (1.2)	88.8 (0.7)	92.9 (0.8)

Next, we analyze the class-specific residuals obtained for cdCOLS, cdOLS and cdOMP. In this experiment, we select a test sample from class-1 and calculate the residual of the test sample using the training samples from class-1 for both datasets. This experiment is

repeated 100 times and the average residuals are reported. Fig. 3.18 and Fig. 3.19 show the residual plots for University of Pavia and University of Houston data respectively. As can be seen from the figures, the residual obtained from cdOLS in each iteration is smaller than the residual obtained from cdOMP. Also, the residual obtained from cdOLS is close to the optimal one obtained from cdCOLS in each iteration.

Finally, in order to validate the generalization capabilities of these classifiers, we plot for the University of Pavia dataset in Fig. 3.20. In this experiment, 30 training samples per class are used. As can be seen from these maps, KcdOLS and cdOLS generally gives much more accurate classification maps compared with KcdOMP and cdOMP. For e.g., there is substantially less salt and pepper misclassification error for the brick and bitumen classes for KcdOLS compared with KcdOMP. We use a white rectangle to mark the regions in the maps.

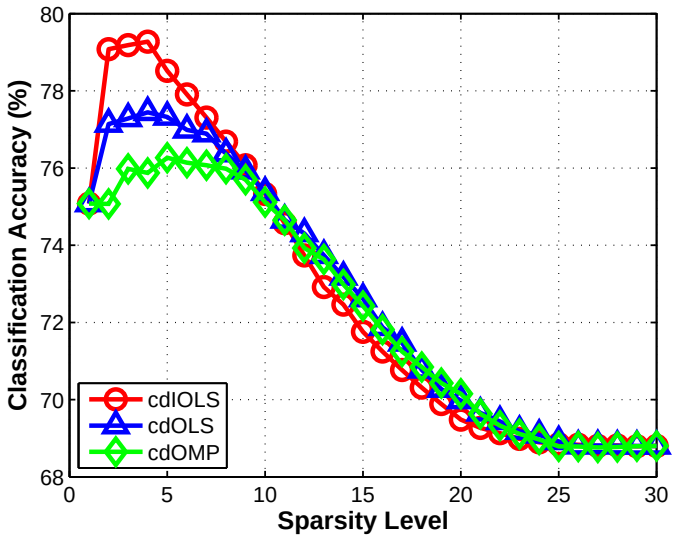


Figure 3.16: Overall classification accuracy (%) versus sparsity level S for the University of Pavia data.

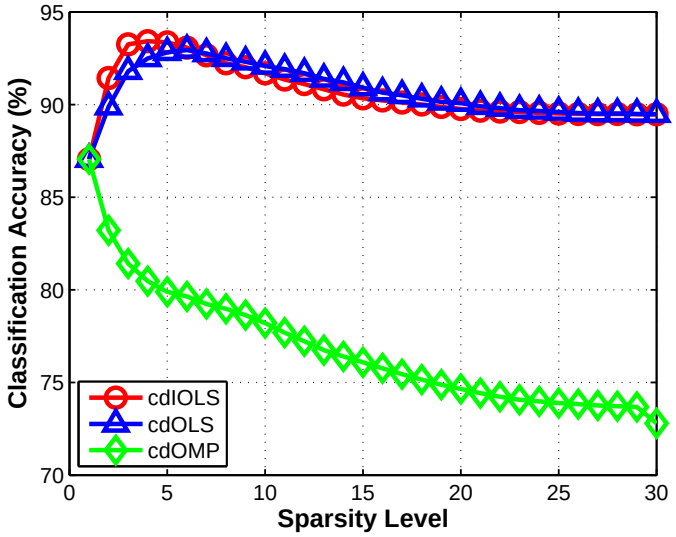


Figure 3.17: Overall classification accuracy (%) versus sparsity level S for the University of Houston data.

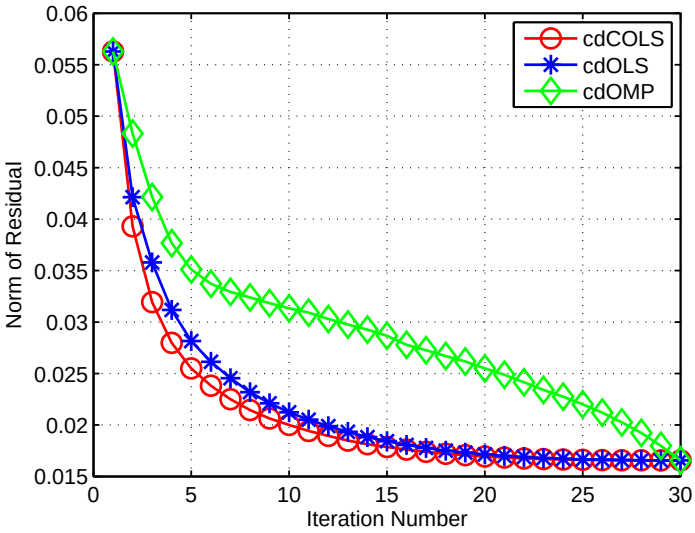


Figure 3.18: Norm of residual versus iteration number for the University of Pavia data.

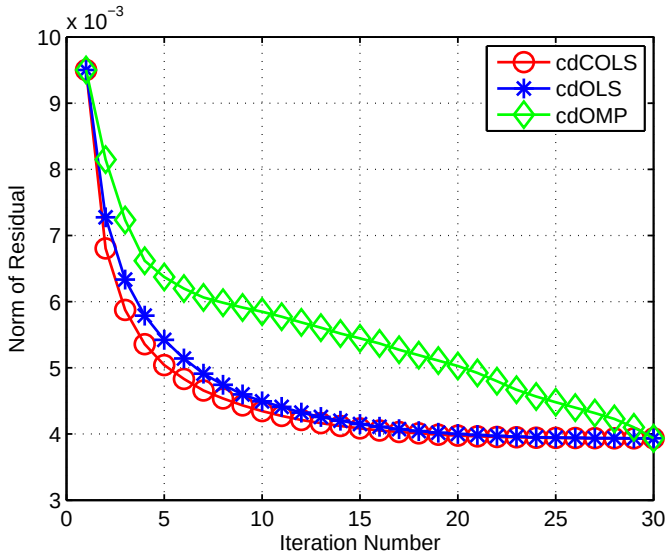


Figure 3.19: Norm of residual versus iteration number for the University of Houston data.

3.5 Proposed simultaneous block orthogonal matching based classification

3.5.1 Simultaneous orthogonal matching pursuit

The simultaneous orthogonal matching pursuit (SOMP) [62] and block orthogonal matching pursuit (BOMP) [63] are all variants of orthogonal matching pursuit (OMP) that explore the *block structure* of test samples and training samples respectively. In this work, we effectively combine these two recovery methods and propose SBOMP to explore the block structure of both training and test samples simultaneously. SBOMP is illustrated in Algorithm 5. Let x_t be a test sample. Assume A_i contains spatial neighborhood samples around x_i (inclusive of x_i), S contains the spatial neighborhood samples of x_t (inclusive of x_t) and K is the sparsity level. SBOMP estimates the coefficient $\hat{C}$ based on the K mostly correlated spatial training samples in A.

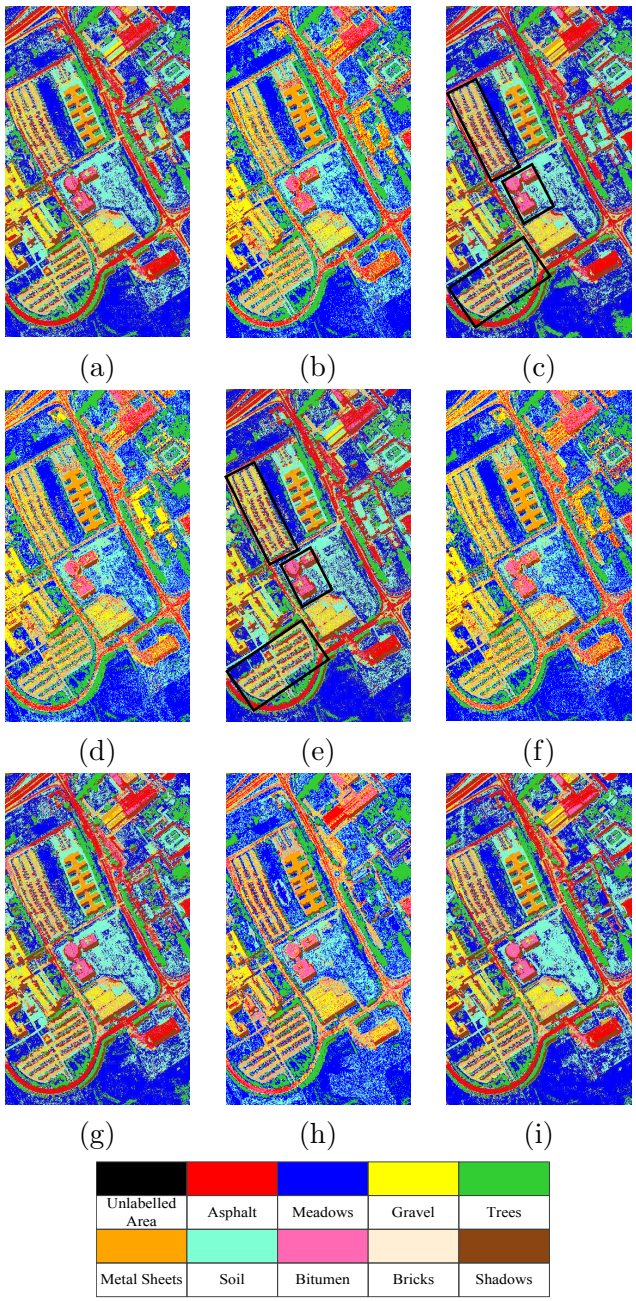


Figure 3.20: Classification maps of University of Pavia dataset generated using (a) KcdCOLS (b) cdCOLS (c) KcdOLS (d) cdOLS (e) KcdOMP (f) cdOMP (g) KSRC (h) SRC (i) SVM.

Algorithm 5 SBOMP

1: **Input:** A training dataset $A = \{A_i\}_{i=1}^{n}$, test data S and row sparsity level K.

2: Initialize $R^0 = S$, $\Lambda^0 = \emptyset$, and the iteration counter $m = 1$.

3: **while** $m \leq K$ **do**

4: Update the support set $\Lambda^m = \Lambda^{m-1} \cup \lambda$ by solving

$$\lambda = \underset{i=1,2,\dots,n}{\operatorname{argmax}} \|A_i^t R^{m-1}\|_{2,1}.$$

5: Derive the coefficient matrix C^m based on

$$C^m = \left(A_{\Lambda^m}^t A_{\Lambda^m}\right)^{-1} A_{\Lambda^m}^t S$$

6: Update the residual matrix R^m

$$R^m = S - A_{\Lambda^m} C^m$$

7: $m \leftarrow m + 1$

8: **end while**

9: **Output:** Coefficient matrix $\hat{C} = C^{m-1}$.

3.5.2 Simultaneous orthogonal matching pursuit based classification

The classification method employed after SLSPP is SBOMP-based Classification (SBOMP-C), as described in Algorithm 6. Since SLSPP can preserve the spatial neighboring samples for both training and test samples in a lower-dimensional subspace, by using SBOMP-C, we can exploit the block structure relationship effectively in the spatial domain between training and test samples.

Algorithm 6 SBOMP-C

1: **Input:** A spectral-spatial training data $A = \{A_i\}_{i=1}^{n}$, test data S and row sparsity level K.

2: Calculate row-sparsity coefficient $\hat{C}$ based on

$$\hat{C} = \text{SBOMP } (A, S, K)$$

3: Calculate residuals for each class

$$r_k(S) = \|S - A\delta_k(\hat{C})\|_2, \quad k = 1, 2, \dots, c$$

4: Determine the class label of S based on

$$\omega = \underset{k=1,2,\dots,c}{\text{argmin}} (r_k(S)).$$

5: **Output:** A class label ω.

3.6 Conclusion

In this work, we have presented a new sparse representation based classifier — cdSRC for HSI classification. In cdSRC, a test sample is represented in a way that exploits the correlation and Euclidean distance information between the test sample and training samples in

a class-wise manner. Through experimental results based on three real-world hyperspectral datasets, it is clear that cdSRC not only dramatically improves the performance of traditional SRC but also outperforms popular traditional classifiers, including ℓ_2-regularized classifiers and SVM. Additional improvements in classification performance can be observed with the kernel variant of the proposed approach (KcdSRC).

We also present a class-dependent OLS-based classification method named cdOLS for the problem of hyperspectral image classification. We also extend cdOLS into its kernel variant. Through two real-world hyperspectral datasets, we demonstrate that our proposed methods outperform cdOMP, KcdOMP as well as SVM. We also demonstrate that the classification performance of the proposed methods are close to that of cdCOLS and KcdCOLS. Our proposed developments are based on the observation that OLS is generally better suited for sparse coefficient recovery. We also present an *combinatorial* OLS based classifier - COLS, that acts as an upper bound on the performance of such classifiers, and can itself be used as well when the training dictionary is small. For scenarios where training dictionaries are not small, the more feasible cdOLS method has very similar performance to cdCOLS (in both the input and kernel induced space).

Finally, we present SBOMP based classification method to exploit the spatial structure between training and test samples. Since it incorporates the spatial information of training and test samples, the classification accuracy is dramatically higher than the ones without using spatial information. We also demonstrate that its classification performance can be further improve by preprocessing the data using the SLSPP which preserves the angular relationship between samples and their spatial neighborhood.

Chapter 4

Real-World Applications

4.1 Introduction

A primary purpose of using remote sensing images, such as hyperspectral images, is to identify or classify target materials present in a scene without physical contact. Modern hyperspectral sensors are able to acquire densely sampled spectral information of objects across the wide range of electromagnetic spectrum ranging from visible to near-infrared. This enormous information can facilitate the identification and classification process of object materials present in a scene. However, for hyperspectral images, the desired objects are occasionally covered by shadows of buildings, trees or clouds (or in general, suffer from illumination variation between training and testing conditions). Hyperspectral measurements under such variation results in significant variability in the spectral profiles of such objects. Such scenarios can complicate or even destroy performance of classification systems if they are not invariant to such behavior.

There has been some prior work focused on addressing the issue of different illumination conditions for remote sensing images. In [64, 65], the authors first segment out the shadow areas and correct their intensity differences based on the information provided by the non-shadow areas. In [66], shadow-insensitive detection and classification based on the atmospherically corrected hyperspectral imagery are presented. A spectral anomaly detection algorithm under the deep shadow areas has been proposed in [67]. In [68], the authors

use the spectral angle measure to estimate abundance. A cloud-shadow atmospheric correction technique is developed in [69] for hyperspectral coastal ocean data. However, such an approach needs specific pixels under the shadows and sunlit areas to be able to obtain good correction results. In [70], the authors combine LiDAR and hyperspectral data to detect vehicles under the shadow areas. The problem of hyperspectral data change detection under the different illumination conditions is addressed in [71]. The methods described above rely on either pre-processing the hyperspectral data to reduce the effect of shadows or employing another data source, such as LiDAR to capture the characteristics of materials under shadow areas. In [72], the authors proposed a morphological shadow index to automatically detect the presence of shadows, which is then used as a spatial constraint for extracting building features from remotely sensed images — it is very specific to the extraction of buildings based on the spatial constraint of shadows. In our work, we focus on robust classification of object pixels that are occluded by shadow and make no assumptions about the spatial properties of objects.

A marsh is a type of wetland whose main characteristics depend on the soil type, salinity of the water, the plants as well as some other environmental factors. Marshes provide habitat for variety of plants and wildlife which are of economic importance. Some other functionalities of marshes include controlling floods and purifying water by filtering out pollutants etc. Therefore conserving marshes are of great importance. Since marshes are composed of a variety of plants, the identification of wetland plant species is an important research topic for rapid marsh delineation and proactive management.

Mangroves are of considerable importance for the salt marsh ecosystem. Most importantly, they can provide wild life habit and a nursery area for many important coastal

creatures as well as provide natural protection of the coastline. Most of commercial seafood spent their lives in mangroves residing in the wetland. Besides these benefits, mangroves are also a profound sources of food. The mangroves leaves can be turned into a rich source of food called detritus for many animals including crabs and shrimps. Mangroves also have the ability to improve the water quality and clarity by cycling pollutants and filtering sediments and debris. Realizing its importance for the marsh ecosystem and tremendous lost during the last decades, mangrove vegetation are now legally prohibited by government from cutting and destruction without a permit. Considering the uniqueness and their multiple ecological functions they provide, maintaining successful establishment of mangroves are vital for the salt marsh ecosystem. Therefore, finding an efficient way to monitor the increase or decrease of the mangrove vegetation in the salt marsh area plays a vital role. Another important species in wetland is the spartina alterniflora which is native to the southern and east coasts of North America. Although spartina alterniflora sometimes can be beneficial to human beings in that it can trap sediment and raise marsh elevations to protect against flooding and provide habitat for aquatic animals, it is a non-native and invasive species in salt marsh systems. It is well-known that wetlands are generally susceptible to invasion, and non-native species can cause changes to the wetlands including populations and communities of plants and animals as well as the nutrients. The spartina alterniflora suppresses the seedlings of mangroves. The most effective way to estimate the increase or decrease of the mangroves and spartina alterniflora are based on the aerial photographs of the coastal areas. In the following paragraphs, we will discuss the remote sensing techniques in the application of wetland species monitoring and classifications.

There has recently been a growing interest in the application of remote sensing approaches for the analysis of marsh species thanks to the recent advances in airborne imaging sensors as well as some remote sensing satellites. Using remote sensing images for mapping and monitoring the wetlands can avoid directly accessing to the wetland, where there are some dangerous wildlife and endemic diseases. Different types of wetlands classification and identification based on the satellite remote sensing data is studied in [73] including which classification techniques were most effective and useful in identifying wetlands and separating them from other land cover classes. In [74], the authors apply spectral mixture analysis technique on AVIRIS hyperspectral data to study the structure of wetlands with emphasis on the complex of spartina species. In [75], several satellite and airborne multispectral and hyperspectral remote sensing images are used to classify different types of salt-marsh vegetation species based on several unsupervised and supervised classifiers including K-means, maximum likelihood and spectral angle mapper. In [76], the authors developed a comprehensive way to build spectral libraries of wetland species to facilitate the application of hyperspectral images for wetlands cover classification and monitoring. In [77], hyperspectral images and light detection and ranging are combined to exploit both the spectral and elevation information of different salt marsh cover classes based on the decision tree. A comprehensive review of multispectral and hyperspectral remote sensing for mapping of wetland vegetation can be found in [78, 79].

4.2 Urban hyperspectral data classification under the shadow

The first validation dataset is acquired using an ITRES-CASI 1500 hyperspectral imager over the University of Houston campus and the neighboring urban area. It has 144 spectral bands over the 380 - 1050nm wavelength range. This image has a spatial dimension of

1905×349 with a spatial resolution of 2.5 m. The image has a prominent cloud in a part of the scene. There are eight classes defined under the shadow and non-shadow areas. The true color image of University of Houston dataset inset with the ground truth is shown in Fig. 4.1. Fig. 4.2 shows the mean signature for these classes under the sun and shadow (in the remainder of this paper, for notational convenience, we refer to pixels under well illuminated regions with as being *under sun*, and regions under cloud shadows as being *under shadows*). Note that this is a subset of the 15 classes available for this dataset (released by us for the 2013 IEEE GRSS Data Fusion Contest [1]) — we focus on classes where such illumination variations exist, to demonstrate performance for the specific case of such illumination variations. The total number of samples under the *well-lit* and *cloud shadow* areas for each class is listed in Table 4.1.

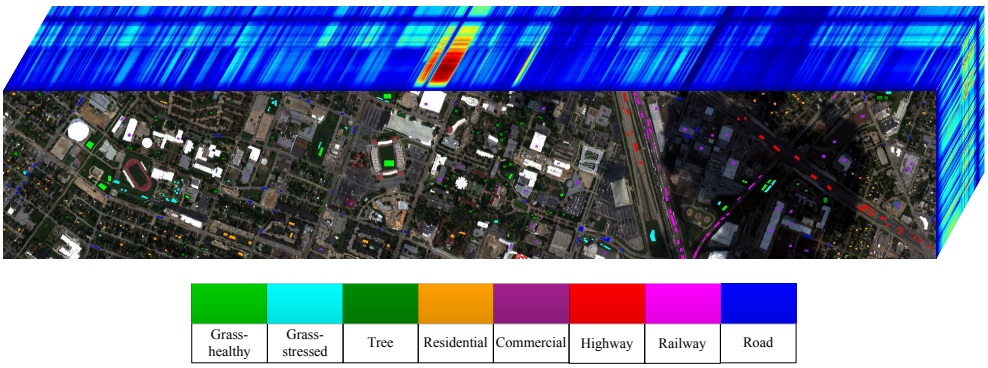
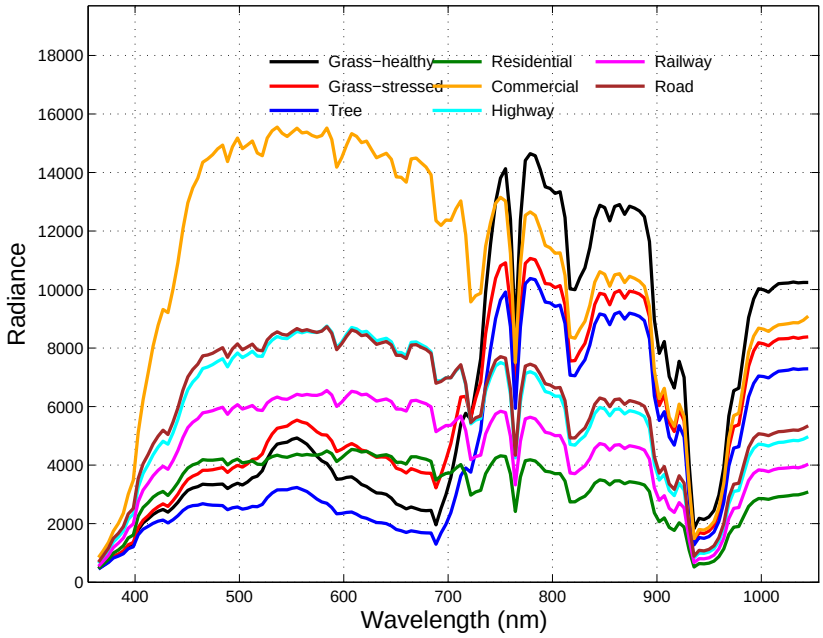


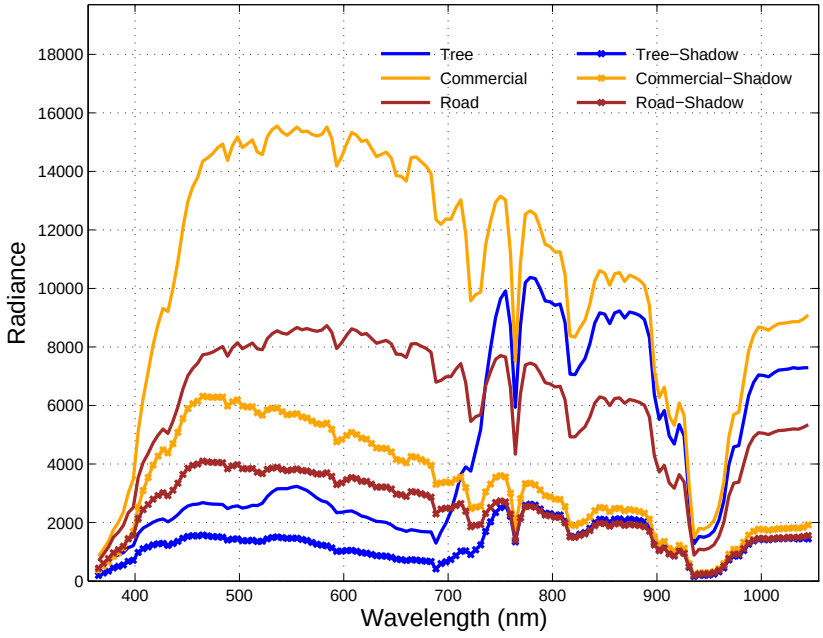
Figure 4.1: True color image of hyperspectral University of Houston data inset with ground truth used in this work.

We first visualize the subspaces found by different dimensionality reduction methods for the data affected by varying illumination conditions. For the purpose of visualization, we pick three wavelength bands (two in visible and one in near-infrared) to show the distribution of three different class samples including tree, building and road under sun and shadow

[1] `http://hyperspectral.ee.uh.edu/?page_id=459`



(a)



(b)

Figure 4.2: Mean signatures of hyperspectral University of Houston dataset under (a) well-lit and (b) well-lit and cloud-shadow areas.

Table 4.1: The number of samplers for eight classes that exist under the well-lit and cloud-shadow areas in the University of Houston data.

Class Name		*Number of Samples*	
		Non-Shadow	*Shadow*
1	*Grass-healthy*	875	178
2	*Grass-stressed*	906	158
3	*Tree*	986	119
4	*Residential*	992	80
5	*Commercial*	597	456
6	*Highway*	710	326
7	*Railway*	914	232
8	*Road*	1059	150

on a sphere in Fig. 4.3. Note that we expect to see similar trends in general for various combinations of spectral wavelengths. We use triangle and circle to represent samples under the sun and shadow areas respectively. From the figure, we can see that samples under the sun and shadow areas are close to each other in terms of angle. The samples in building and road classes are close to each other because of the similarity in spectral response, likely due to similar types of materials being used.

We then analyze the subspaces found by dimensionality reduction methods based on Euclidean distance and angular information respectively. In this experiment, we combine samples under the non-shadow and shadow areas into the same class. As is shown in [34], the rank of ADA and LDA is at most $c - 1$, where c is the number of classes. Thus ADA and LDA both find two-dimensional subspaces for this particular example since we only consider three classes. Fig. 4.4 depicts the subspaces found by ADA, LDA, LADA and LFDA using the University of Houston datasets. From the figure, we can tell that the subspace found by ADA is more discriminative than the one found by LDA. The poor performance of LDA is mainly caused by the huge intensity differences of spectral signatures under the non-shadow and shadow areas which form different clusters in Euclidean-distance

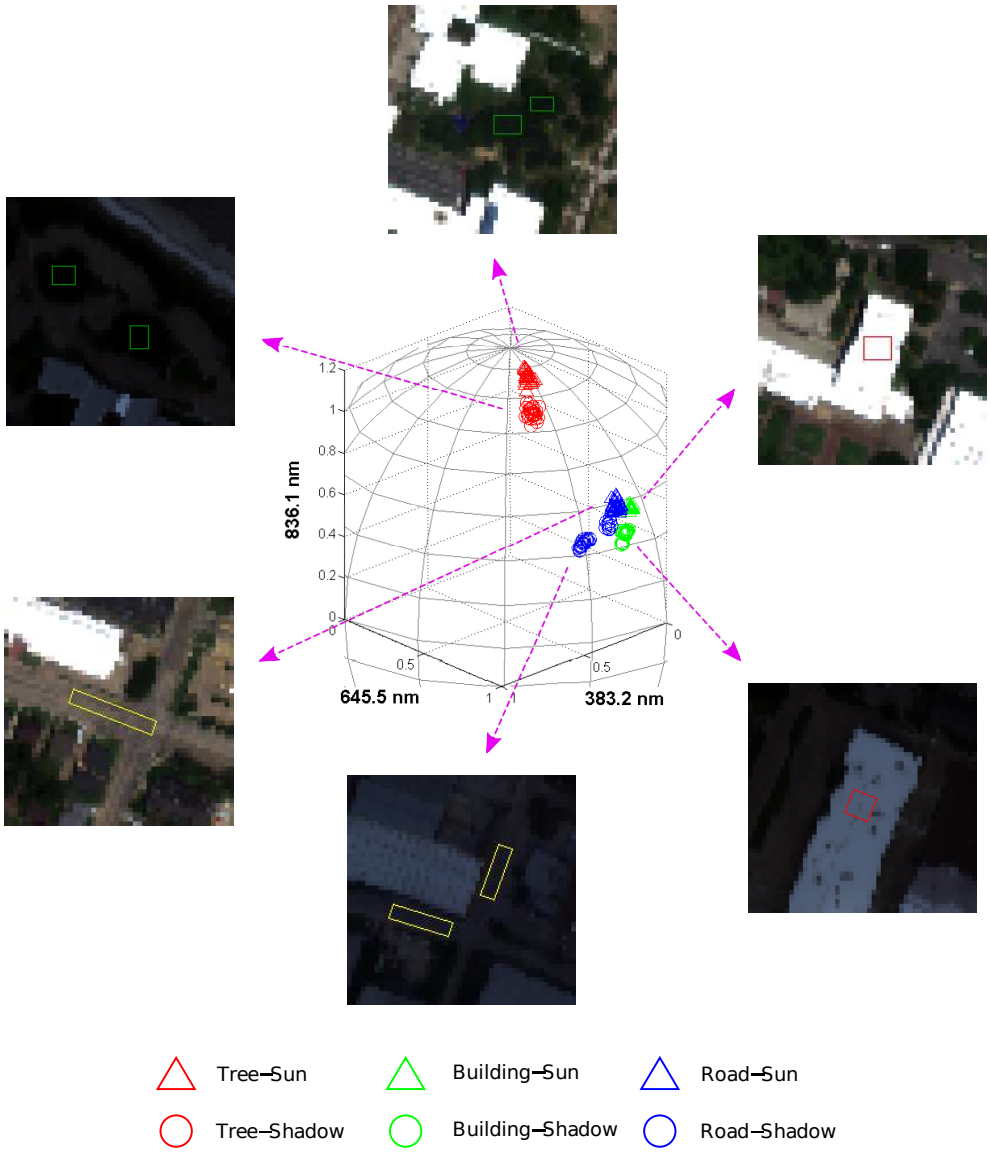


Figure 4.3: Illustrating the normalized clusters on a sphere, corresponding to classes under the well-illuminated and shadow areas.

domain. Due to its global structure, LDA can not preserve these local structures which causes inaccurate estimation of within-class scatter matrices. ADA, on the other hand, is minimally affected by the illumination differences due to its invariance nature to linear scaling. LFDA can preserve these clusters locally but there are still some overlaps between samples from different classes. LADA on the other hand, not only can angularly separate between-class samples but also preserves the local structure of within-class samples which lead to even better separation of between-class samples in terms of angle.

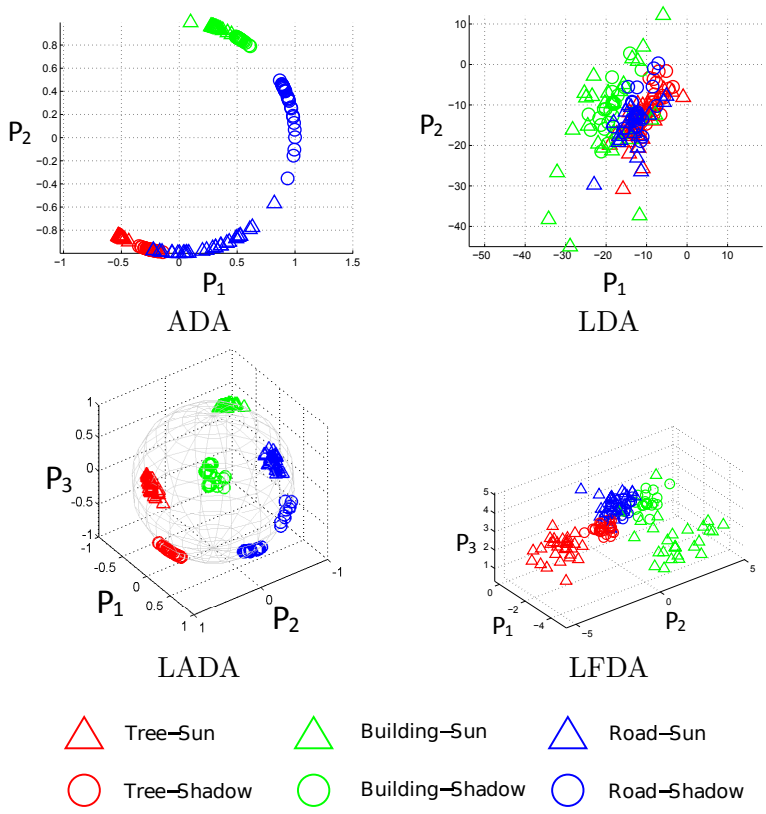


Figure 4.4: Subspaces found by ADA, LDA, LADA and LFDA for three classes including tree, building and road in University of Houston data under the well-lit and cloud-shadow areas.

The class-wise accuracies using NN and SRC as the backend classifiers for various dimensionality reduction methods for the University of Houston are shown in Table 4.2 and

Table 4.3. The class-wise accuracies are calculated using 50 training samples per class. Each experiment is repeated 10 times and the average accuracy is reported. As can be seen from the tables, the angle-based dimensionality reduction methods yield very high classification accuracies for each class and significantly outperform the corresponding Euclidean-based dimensionality reduction methods.

Table 4.2: Classwise accuracies using NN as the backend classifier for various dimensionality reduction methods for the University of Houston data.

Class / Algorithms		*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Grass-healthy*	96.9	95.9	95.4	92.4	95.6	86.3	91.6	80.1
2	*Grass-stressed*	97.5	97.1	96.4	92.6	95.5	91.4	72.7	81.5
3	*Tree*	98.2	98.2	95.2	95.4	95.1	90.8	84.1	84.5
4	*Residential*	94.7	86.5	83.6	77.5	86.4	84.9	77.2	68.8
5	*Commercial*	89.2	90.9	82.6	86.1	75.5	78.5	80.5	55.6
6	*Highway*	95	91.1	85.7	82.7	83.2	71.8	79.9	51.1
7	*Railway*	93	80.9	75.1	65.0	76.1	70.9	77.2	48.8
8	*Road*	91.4	85.2	82	75.1	86.7	78.0	82.2	53.7
	Overall Accuracy	94.5	90.7	87.0	83.4	86.8	81.6	80.7	65.5

Table 4.3: Classwise accuracies using SRC as the backend classifier for various dimensionality reduction methods for the University of Houston data.

Class / Algorithms		*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Grass-healthy*	97	95.9	95.4	92.5	96.6	85.8	94.5	79.9
2	*Grass-stressed*	98	97.1	96.4	92.7	95.5	84.7	91.0	74.5
3	*Tree*	98.3	98.2	95.3	95.6	97.6	90.8	88.1	85.2
4	*Residential*	95.1	86.8	83.6	78.4	87.5	68.8	80.0	53.6
5	*Commercial*	90.1	91.0	82.6	86.9	84.1	74.0	84.0	46.6
6	*Highway*	95	91.1	85.7	83.0	87.7	65.0	82.3	48.5
7	*Railway*	93.1	80.9	75.2	65.1	80.6	63.1	78.9	45.0
8	*Road*	91.5	85.2	82	75.1	86.2	66.6	81.3	41.9
	Overall Accuracy	94.8	90.8	87.0	83.7	89.5	74.9	85.0	59.4

Next, we demonstrate how the number of training samples affects the classification performance for different methods. We test the number of training samples per class from 20 to 80 with a step size of 5. Fig. 4.5 and Fig. 4.6 show the classification accuracies versus number of training samples per class using NN and SRC for the University of Houston

dataset respectively. It is obvious from the figure that the classification performance of angle-based dimensionality reduction methods are approximately insensitive to the number of training samples, and they significantly outperform the corresponding Euclidean distance-based dimensionality reduction methods, especially when the number of training samples is small. This is a very useful property for remote sensing data analysis since acquiring labelled samples is time-consuming and expensive. Euclidean distance-based dimensionality reduction methods generally need sufficient training samples to accurately estimate the parameters due to the within-class high variance of data caused by different illumination conditions.

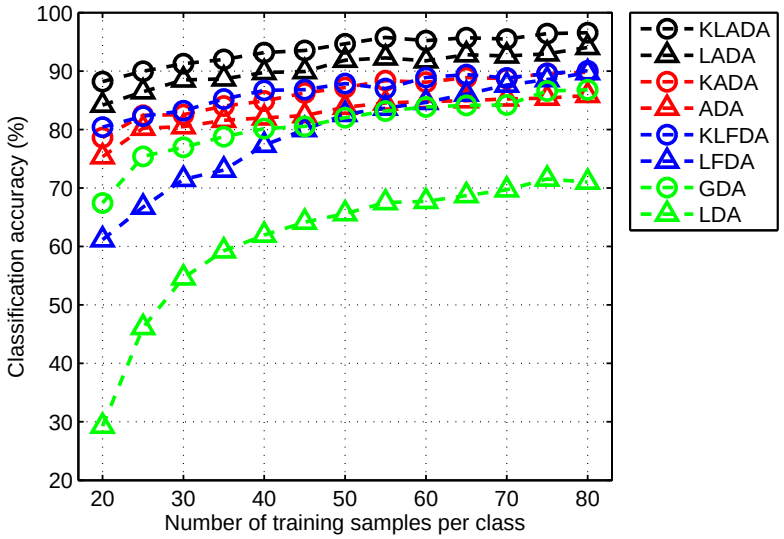


Figure 4.5: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, LFDA, GDA and LDA followed by NN for the University of Houston data.

4.3 Wetland hyperspectral data classification under the shadow

The second hyperspectral data was acquired by us in Galveston, Texas in October, 2014 and includes two scenes captured at ground-level (side-looking views) over wetlands in

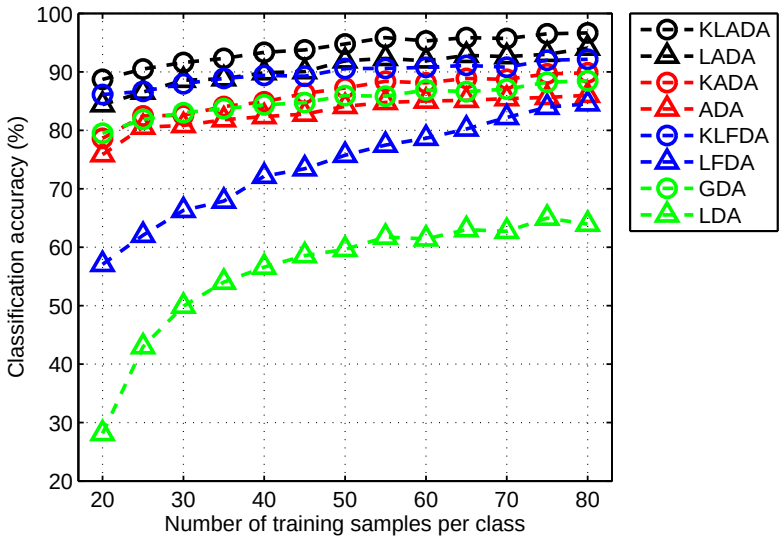
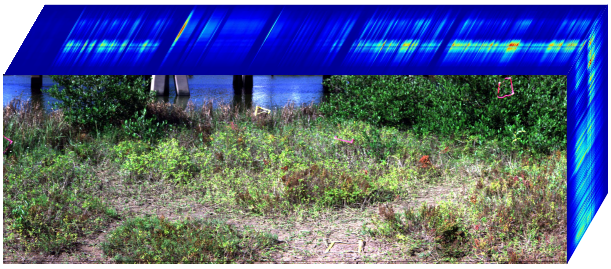


Figure 4.6: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, LFDA, GDA and LDA followed by SRC for the University of Houston data.

Galveston. The two image cubes are referred to as area 1, and area 2, representing different regions of the wetlands that were imaged — In addition to common wetland classes, area 2 has Black Mangrove (*Avicennia germinans*) trees in the scene, a species which is of particular interest in ecological studies of wetlands. The two true color images of Galveston data are shown in Fig. 4.7. We acquired this data using a Headwall Photonics hyperspectral imager which provides measurements in 325 spectral bands with a spatial size of 1004×5130. The hyperspectral data uniformly spanned the visible and near-infrared spectrum from 400 nm - 1000 nm. The objects of interests are primarily vegetation species common in such wetlands. Six different classes were identified in area-1 including soil, *symphyotrichum*, *sedge*, *spartina patens*, *borrichia* and *rayjacksonia*. The second area includes *Avicennia germinans*, *batis*, *schoenoplectus*, *spartina alterniflora*, *soil*, *water* and *bridge*. Since soil and *schoenoplectus* are included in both areas, the total number of classes in the combined

library are eleven. The total number of samples for each class is tabulated in Table 4.4 and the mean signatures are plotted in Fig. 4.8 for the samples under the sun and shadow. We note that very little work has been done with such side-looking hyperspectral images of vegetation, and that such images are very likely to have the same class appear in well-lit and shadow areas (shadows from the canopy of plants).

Wetland data, area - 1

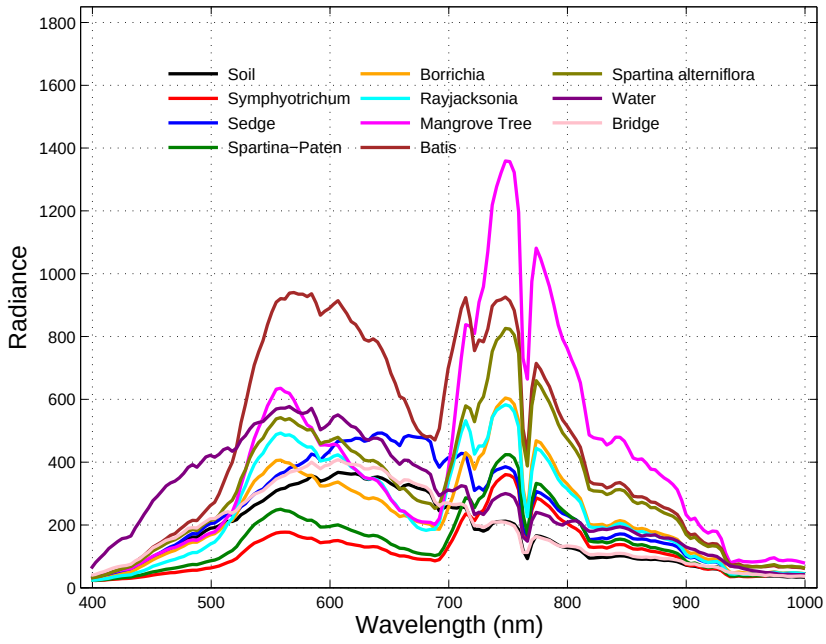


Wetland data, area - 2

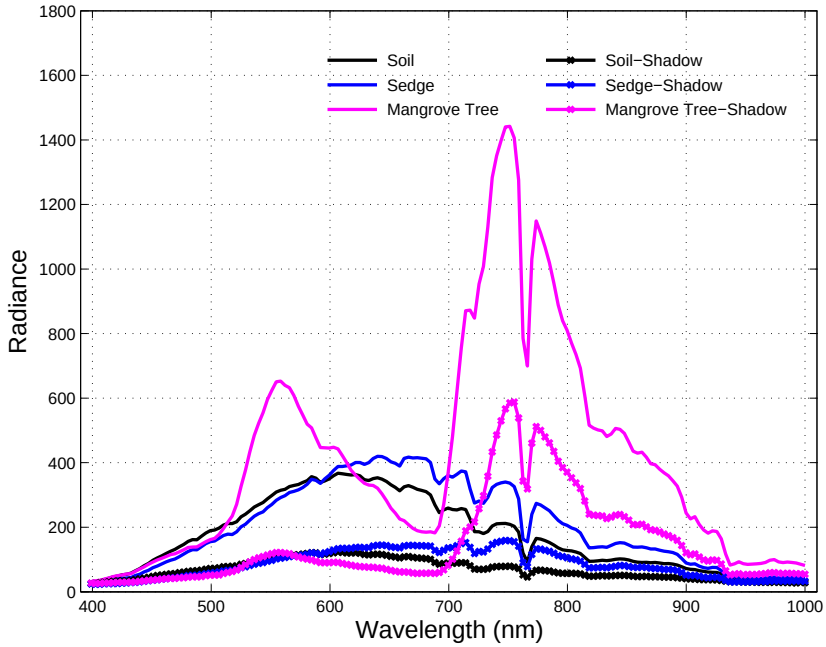
Figure 4.7: True color images of the wetland dataset.

Table 4.4: Six classes under well-lit and shadow in Galveston data and the corresponding number of samples

Class Name		*Number of Samples*	
		Non-Shadow	*Shadow*
1	*Soil*	233	181
2	*Symphyotrichum*	196	247
3	*Sedge*	485	334
4	*Spartina patens*	469	393
5	*Borrichia*	227	213
6	*Rayjacksonia*	209	161
7	*Avicennia germinans*	590	598
8	*Batis*	204	152
9	*Spartina alterniflora*	228	232
10	*Water*	254	240
11	*Bridge*	280	313



(a)



(b)

Figure 4.8: Mean signatures of hyperspectral University of Houston dataset under (a) well-lit and (b) well-lit and cloud shadow areas.

A similar analysis was performed for the wetland data. The main focus of this data is on the different types of vegetation species found in marshes around the Texas coast — the ability to monitor such species is very beneficial for a variety of ecological studies. Fig. 4.9 shows distributions of three different types of species including *schoenoplectusz*, *borrichia*, *rayjacksonia* on a sphere under the well-lit and shadow areas respectively. Since these three classes are different types of grasses, they are all somewhat close to each other even in an angular sense. The subspaces found by angle and Euclidean distance based dimensionality reduction methods are shown in Fig. 4.10. Even for this very challenging data, angle-based dimensionality reduction methods generally find subspaces that are more discriminative than Euclidean-based counterparts.

Table 4.5 and Table 4.6 show the class-specific accuracies using NN and SRC as the backend classifiers for the Galveston dataset respectively. Fig. 4.11 and Fig. 4.12 show the classification accuracies obtained as a function of training sample size per class using NN and SRC as the backend classifiers. It can be seen from these two results that the angle-based dimensionality reduction methods still outperform Euclidean distance-based counterparts for this complex vegetation imagery, particularly for difficult classes, using very few training samples.

We also generated classification maps for the wetland dataset (which is a very unique kind of hyperspectral imagery) to demonstrate the potential for using such imagery for classification. Fig. 4.13 and Fig. 4.14 shows the classification maps of wetland data in the two areas respectively. We use 50 training samples per class for both areas. From these two figures, it is clear that we can identify the complex canopy structure of the major species much better with the angular approach (KLADA), compared to an approach that uses

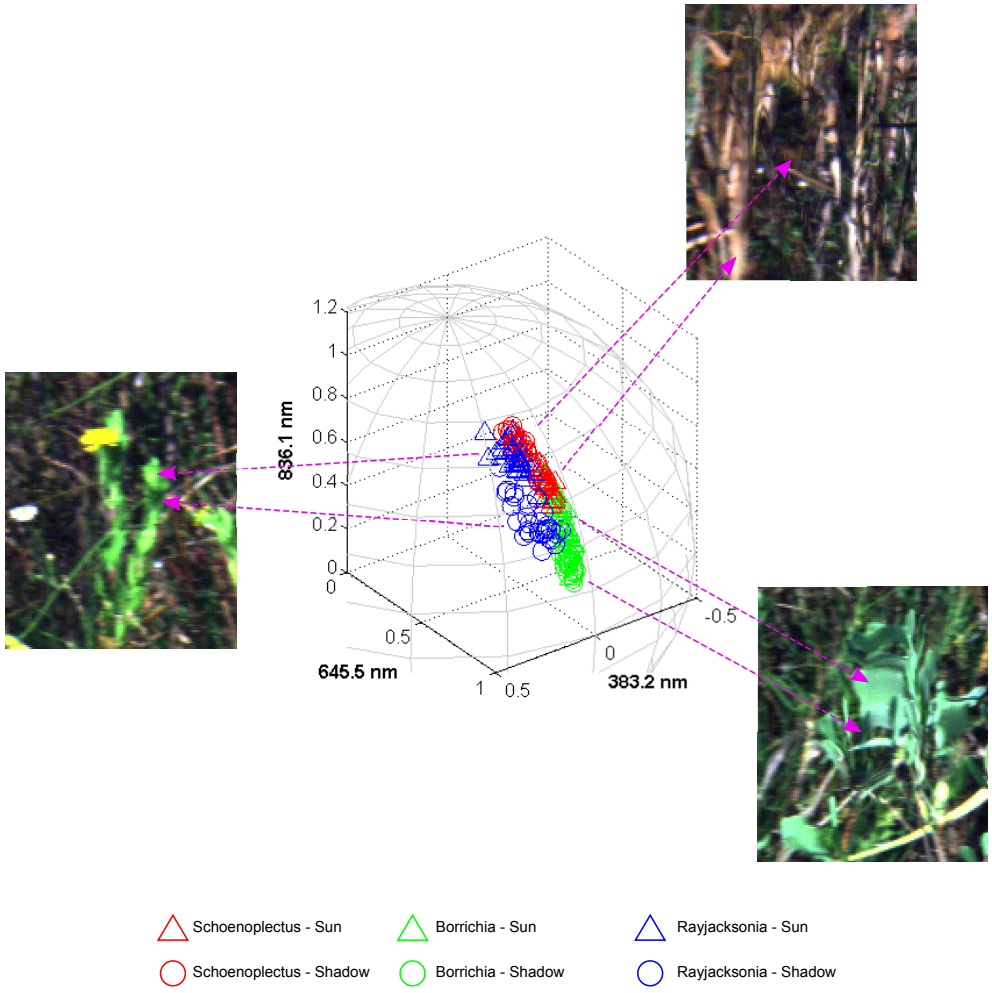


Figure 4.9: Illustrating the normalized clusters on a sphere corresponding to the *schoenoplectus*, *borrichia*, *rayjacksonia* classes under the well-illuminated and shadow areas.

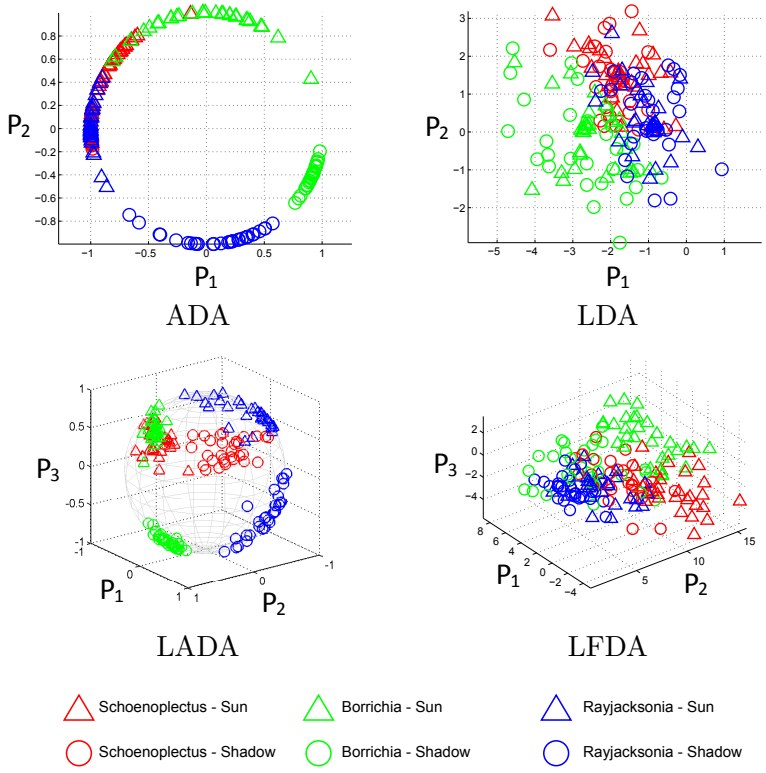


Figure 4.10: Subspaces found by ADA, LDA, LADA and LFDA for three different species classes including *schoenoplectus*, *borrichia*, *rayjacksonia* in the wetland data under the well-lit and shadow areas.

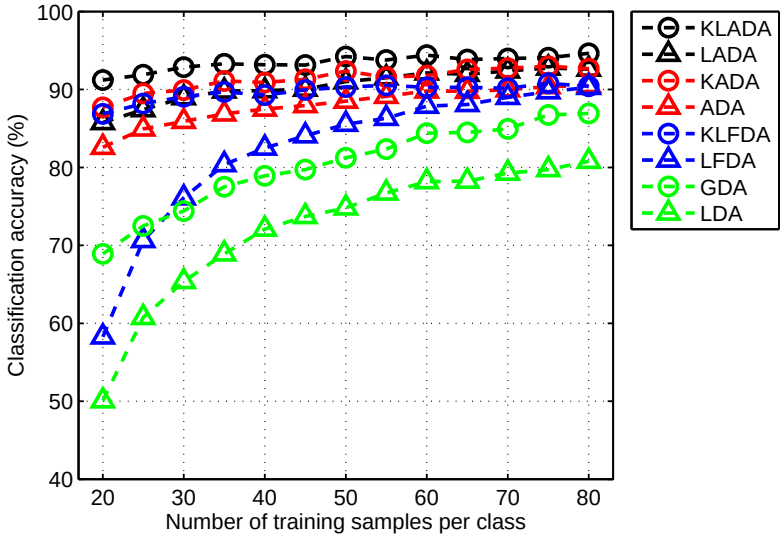


Figure 4.11: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, LFDA, GDA and LDA follwed by NN for the wetland data.

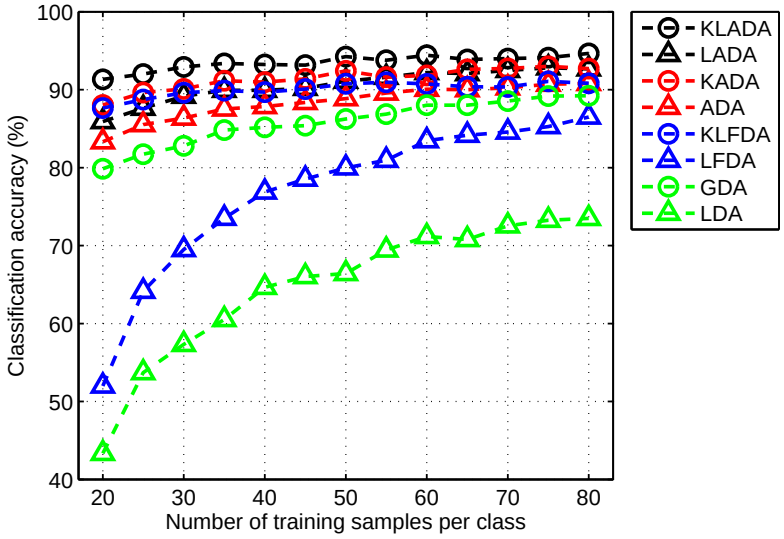


Figure 4.12: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, LFDA, GDA and LDA follwed by SRC for the wetland data.

Euclidean distances (KLFDA).

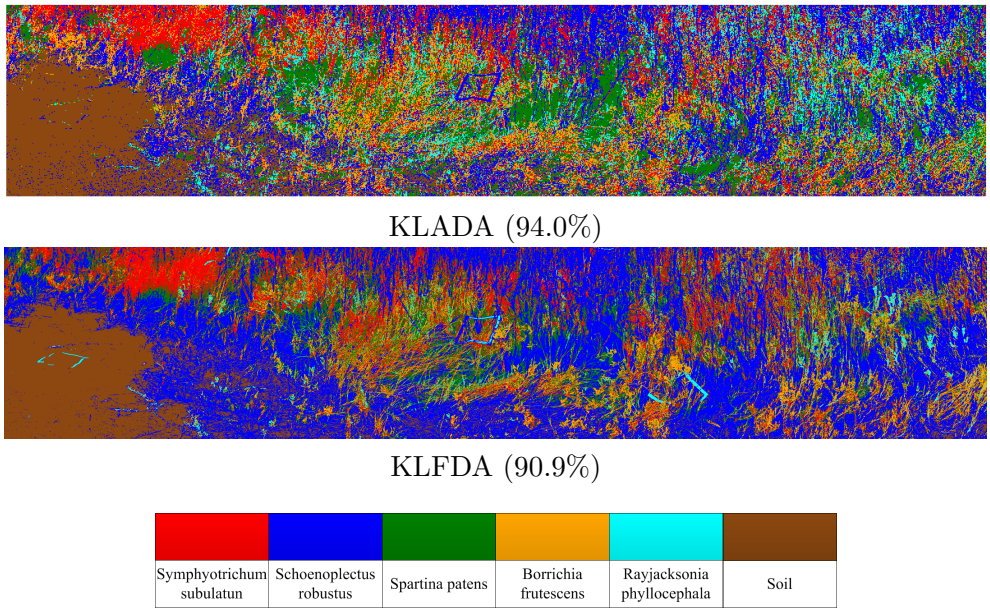


Figure 4.13: Classification maps and accuracies in bracket of wetland data, area-1 generated by KLADA and KLFDA, followed by a NN classifier.

We also validate this method for the problem of wetland mapping using satellite hyperspectral imagery — We used Hyperion imagery, which is widely used due to its spectral range and resolution for ground cover classification [80–82]. We extracted 7 pertinent broader classes (consistent with the labeling potential of Hyperion imagery, with a spatial resolution of $30m$) including saline marsh, intermediate marsh, forest, soil, water, developed areas, and cloud. Marsh classes were extracted using ground truth available through the United States Geological Survey National Wetlands Research Center (USGS-NWRC)[2]. Additional background classes were extracted using the National Land Cover Dataset (NLCD)[3]. Among

[2] `http://pubs.usgs.gov/sir/2014/5110/`

[3] `http://www.mrlc.gov/`

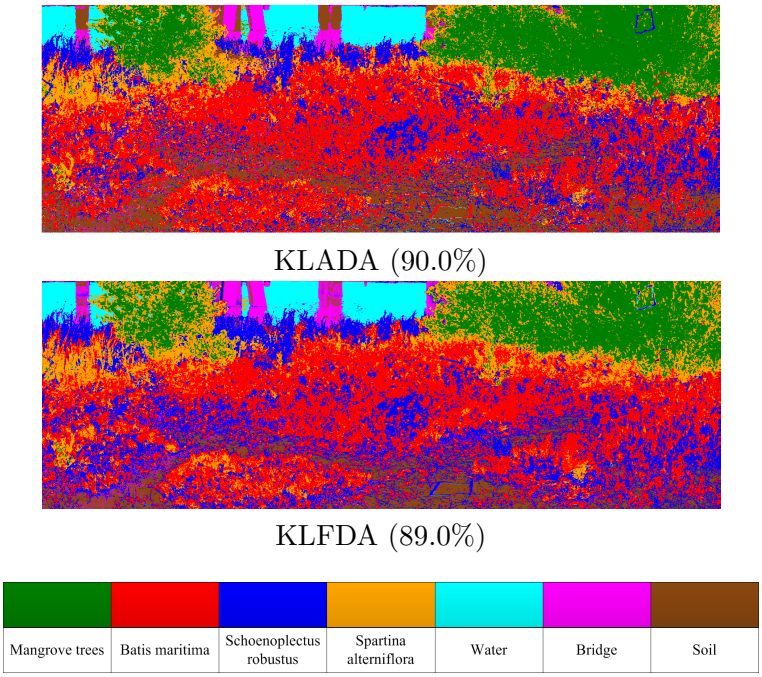


Figure 4.14: Classification maps and accuracies in bracket of wetland data, area-2 generated by KLADA and KLFDA, followed by a NN classifier.

Table 4.5: Classwise accuracies using NN as the backend classifier for various dimensionality reduction methods for the wetland data.

Class / Algorithms		*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Soil*	98.6	97.7	97	96.4	98.0	96.7	94.3	89.4
2	*Symphyotrichum*	90.5	87.5	84.1	85.3	72.7	66.0	78.5	62.1
3	*Sedge*	95.5	96.5	95.9	95.2	94.5	90.0	88.7	83.4
4	*Spartina patens*	77.2	73.0	72.1	67.3	65.9	59.2	68.3	40.9
5	*Borrichia*	94.9	91.6	92.6	88.6	86.4	82.1	88.6	73.0
6	*Rayjacksonia*	97.2	94.3	95.6	88.0	90.0	85.1	85.4	72.8
7	*Mangrove trees*	95.2	97.3	95	95.3	94.1	92.9	63.6	85.9
8	*Batis*	99.6	97.9	92.8	94.4	98.8	97.2	61.2	83.2
9	*Spartina alterniflora*	89.7	66.0	84.8	66.8	92.8	89.3	76.3	73.0
10	*Water*	99.9	99.7	100	99.6	99.7	99.4	84.6	96.7
11	*Bridge*	92	97.7	97.9	98.8	98.8	83.6	94.6	65.4
	Overall Accuracy	93.7	90.8	91.6	88.7	90.2	85.6	80.4	75.1

Table 4.6: Classwise accuracies using SRC as the backend classifier for various dimensionality reduction methods for the wetland data.

	Class / Algorithms	*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Soil*	98.6	97.7	97.2	97.9	97.9	91.7	94.9	78.5
2	*Symphyotrichum*	90.5	87.5	84.1	73.3	73.3	62.3	79.5	61.1
3	*Sedge*	95.6	96.5	96	94.7	94.7	89.8	92.6	83.2
4	*Spartina patens*	77.4	73.0	72.1	67.1	67.1	53.3	70.4	34.6
5	*Borrichia*	94.9	91.7	92.6	86.6	86.6	79.7	87.5	69.2
6	*Rayjacksonia*	97.2	94.3	95.6	91.1	91.1	84.0	86.3	70.5
7	*Mangrove trees*	95.3	97.3	95.4	95.1	95.1	92.6	93.0	83.0
8	*Batis*	99.6	97.9	92.8	99.1	99.1	96.2	88.7	72.6
9	*Spartina alterniflora*	89.9	66.7	85	91.7	91.7	64.7	65.5	43.7
10	*Water*	99.9	99.7	100	99.7	99.7	98.7	99.5	94.7
11	*Bridge*	91.9	97.7	97.9	99.0	99.0	69.2	92.4	51.9
	Overall Accuracy	93.7	90.9	91.7	90.5	90.5	80.2	86.4	67.5

these classes, intermediate marsh, forest, soil and water classes are both under the well illuminated and cloud shadow regions in the Hyperion hyperspectral images, as shown in Fig. 4.15. The image size is 600×330, and the class name with corresponding sample size are tabulated in Table 4.7. The Hyperion hyperspectral imager is aboard NASA's Earth Observing 1 (EO-1) spacecraft, and it has spectral range between 300 nm and 2400 nm with a spatial resolution of 30 m. Radiance values from the Hyperion image are converted to reflectance image using Fast Line-of-sight Atmospheric Analysis of Hypercubes (FLAASH). The specific model input parameter values used in FLAASH are set as follows: atmospheric model is set to tropical, aerosol model is set to maritime and the water absorption feature is set to 1135 nm.

Table 4.8 and Table 4.9 show the class-specific accuracies using NN and SRC as the backend classifiers for the hyperion dataset respectively. The classification results using NN and SRC as a function of different number of training samples for the hyperion data are shown in Fig. 4.16 and Fig. 4.17 respectively. Fig. 4.18 shows the classification maps for the hyperion data using the KLADA and KLFDA to demonstrate the benefit of the

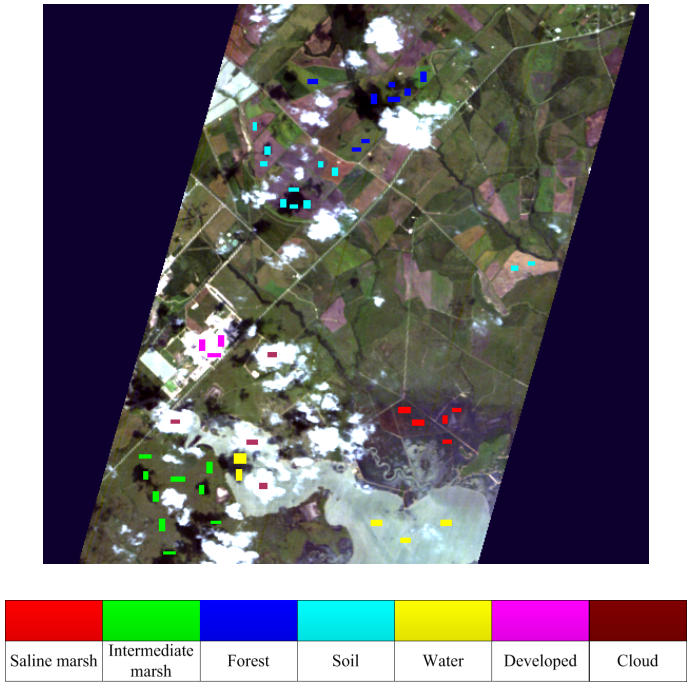


Figure 4.15: True color images of the hyperion hyperspectral data under the cloud.

Table 4.7: Seven classes in Hyperion data and the corresponding number of samples

	Class Name	*Number of Samples*
1	Saline marsh	228
2	Intermediate marsh	402
3	Forest	491
4	Soil	364
5	Water	330
6	Developed	329
7	Cloud	164

angular-based dimensionality reduction methods. 10 training samples per class are used for this experiment. We also highlighted some of the regions which demonstrate the benefit of the KLADA, particularly when classifying important marsh classes under cloud shadows.

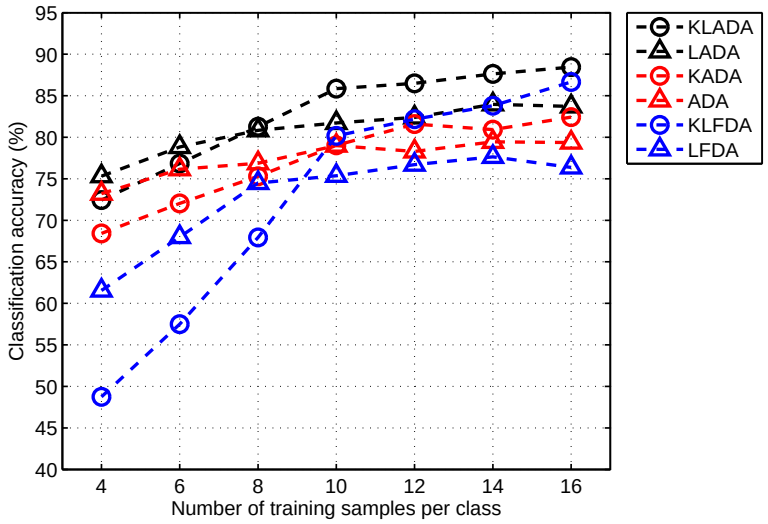


Figure 4.16: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, and LFDA follwed by NN for the first data.

Table 4.8: Classwise accuracies using NN as the backend classifier for various dimensionality reduction methods for the hyperion data.

	Class / Algorithms	*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Saline marsh*	75.6	59.5	68	57.7	66.0	63.3	67.8	15.4
2	*Intermediate marsh*	75.3	74.7	67.8	70.6	68.3	58.8	56.9	26.7
3	*Forest*	71.6	70.9	68.9	67.4	48.0	60.5	59.0	43.9
4	*Soil*	83.9	85.8	78.6	78.7	74.7	61.9	75.5	10.8
5	*Water*	98.3	92.5	93	89.2	89.1	94.9	84.4	73.8
6	*Developed*	86.6	87.6	82.9	84.1	90.9	85.1	26.7	60.3
7	*Cloud*	99.2	99.7	89.3	99.1	94.6	92.6	33.3	64.8
	Overall Accuracy	84.4	81.5	78.4	78.1	75.9	73.9	57.7	42.2

4.4 Conclusion

In this paper, we presented an approach to perform hyperspectral image classification under the challenging scenario of varying illumination conditions in a scene. Specifically, we

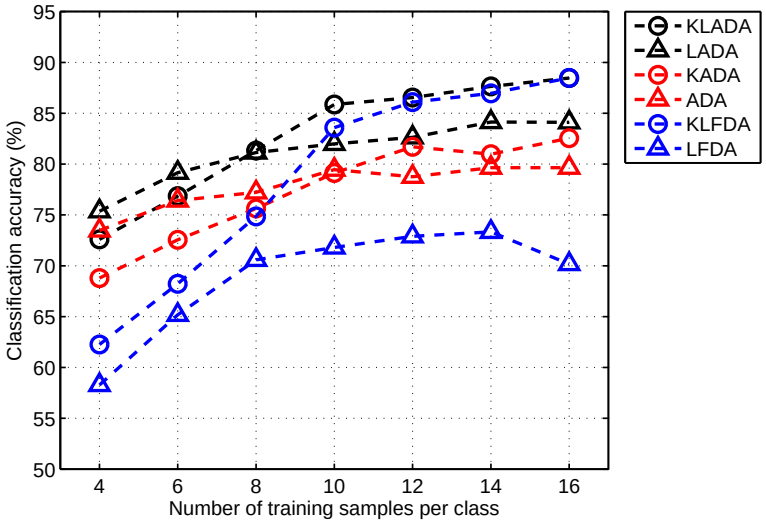


Figure 4.17: Classification accuracies versus number of training samples per class using KLADA, LADA, KADA, ADA, KLFDA, and LFDA follwed by SRC for the Hyperion data.

Table 4.9: Classwise accuracies using SRC as the backend classifier for various dimensionality reduction methods for the hyperion data.

	Class / Algorithms	*KLADA*	*LADA*	*KADA*	*ADA*	*KLFDA*	*LFDA*	*GDA*	*LDA*
1	*Saline marsh*	70.9	56.5	68.1	52.4	69.4	30.4	78.0	33.3
2	*Intermediate marsh*	70.8	74.7	67.7	66.0	73.4	43.5	70.2	44.0
3	*Forest*	68.2	70.9	68.8	66.1	56.6	52.5	64.8	37.8
4	*Soil*	79.4	84.2	79.2	74.7	79.0	25.1	76.8	20.9
5	*Water*	96.3	88.6	93	84.5	94.1	81.4	94.1	76.2
6	*Developed*	85.4	87.6	83.1	78.9	85.5	86.3	81.6	40.2
7	*Cloud*	99.3	99.7	89.5	95.2	98.0	98.6	94.1	52.9
	Overall Accuracy	81.5	80.3	78.5	74.0	79.4	59.7	79.9	43.6

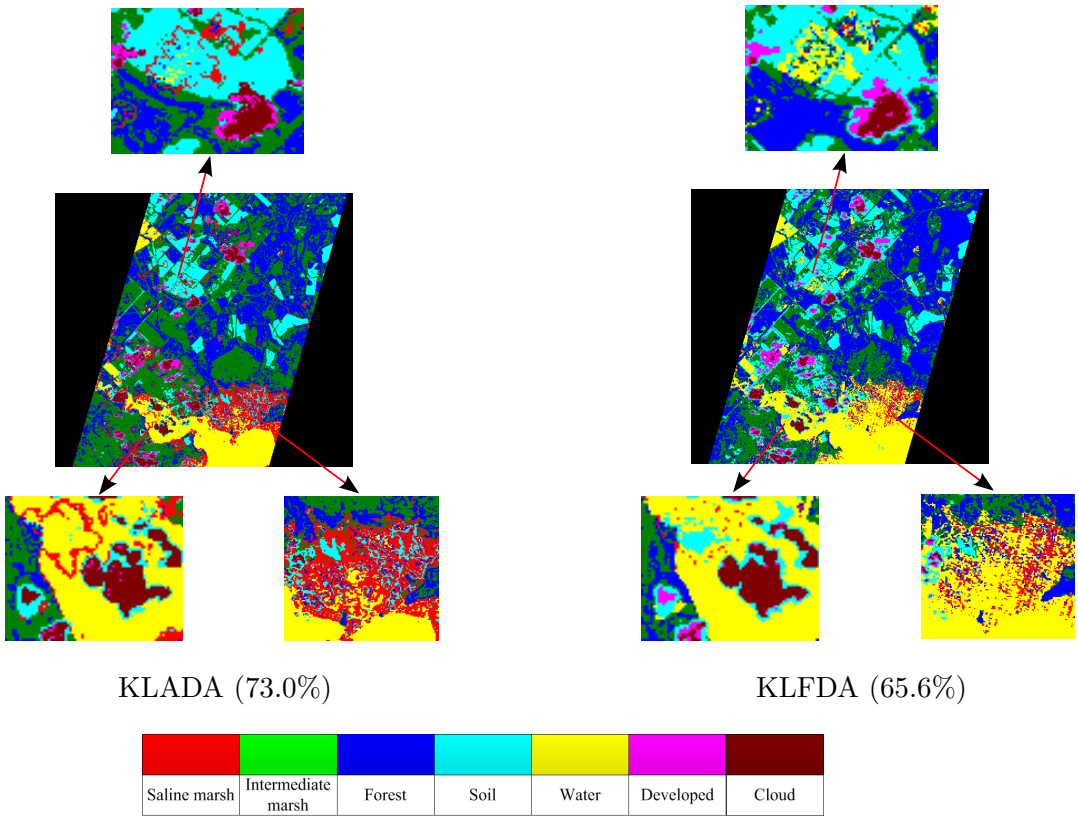


Figure 4.18: Classification maps and accuracies in bracket of hyperion data generated by KLADA and KLFDA, followed by a NN classifier.

demonstrate that the recently developed spectral angle based discriminant analysis methods are particularly suited to hyperspectral image analysis applications where there are illumination variations within the classes. Experimental results on two hyperspectral datasets show that the proposed methods greatly outperform traditional methods, using very limited training data. Within the context of remote sensing images (aerial and satellite images in particular), we note that although the proposed methods would still not be able to classify pixels corresponding to cloud cover, they neverthless have significant potential to improve ground cover classification in regions where there are cloud shadows. It is expected that the classification performance can be further improved by adding spatial (contextual) information when analyzing such challenging hyperspectral imagery with significant illumination variability. In future work, we will extend the dimensionality reduction methods proposed in [34] by incorporating the spatially adjacent information around each sample when learning our optimal subspace.

Chapter 5

Summary and Conclusion

In this dissertation, we proposed several dimensionality reduction and classification methods for hyperspectral image analysis as well as their applications to real-world remote sensing data classification problems.

The main contributions of this dissertation are summarized as follows:

a. ADA, an angle-based dimensionality reduction method, is proposed to project hyperspectral data into a lower-dimensional subspace to separate different class samples in an angular sense. For scenarios where the hyperspectral data exhibit multi-modality or nonlinear separation, we propose local and kernel extensions of ADA — LADA and KLADA, respectively, to tackle the multi-modality and nonlinearity of hyperspectral data.

b. We extend ADA to its unsupervised variant (LSPP) to harness the unlabeled samples as well as preserve the angular relationship between the training samples in the projected subspace. In order to explore the spatial contextual information of HSI, which is known to be useful for hyperspectral data classification, the spatial variant of LSPP (SLSPP) is also proposed, developed and validated.

c. Class dependent Sparse Representation based Classification (cdSRC) is proposed to address the limitations of SRC and improve its classification performance. To deal with classification problems when different class samples cannot be linearly separable,

its kernel variant KcdSRC is also developed.

d. To overcome the limitations of cdOMP for the sparse coefficient recovery, we proposed cdOLS to improve the accuracy of the recovered sparse coefficients which in turn increase the backend classification performance of sparse representation based classifiers.

e. In order to utilize the spatial information of training and test samples, we also propose SBOMP-C. The classification performance of SBOMP-C can be further improved if we preprocess the data with the SLSPP projection we previously developed, which can preserve the spatial information of samples in terms of angle in the lower dimensional subspace.

f. Finally, we use the proposed method to solve real-world remote sensing classification problems including classifying different objects under varying illumination conditions and classifying coastal wetland vegetation species which are of great importance for ecological studies.

References

[1] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *Artif. Neural Networks*. Springer, 1997, pp. 583–588.

[2] G. Baudat and F. Anouar, "Generalized discriminant analysis using a kernel approach," *Neural Comput.*, vol. 12, no. 10, pp. 2385–2404, 2000.

[3] S. Prasad and L. M. Bruce, "Limitations of principal components analysis for hyperspectral target recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 4, pp. 625–629, 2008.

[4] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[5] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." in *NIPS*, vol. 14, 2001, pp. 585–591.

[7] X. He and P. Niyogi, "Locality preserving projections," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, Vancouver, Canada, 2003, pp. 234–241.

[8] M. Sugiyama, "Dimensionaltity reduction of multimodal labeled data by local Fisher discriminant analysis," *J. Mach. Learn. Research*, vol. 8, no. 5, pp. 1027–1061, May 2007.

[9] C. M. Bachmann, T. L. Ainsworth, and R. A. Fusina, "Exploiting manifold geometry in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 441–454, 2005.

[10] D. Lunga, S. Prasad, M. Crawford, and O. Ersoy, "Manifold-learning-based feature extraction for classification of hyperspectral data," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 55–66, January 2014.

[11] D. Lunga and O. Ersoy, "Spherical stochastic neighbor embedding of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 857–871, 2013.

[12] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving dimensionality reduction and classification for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1185–1198, April 2012.

[13] L. Ma, M. M. Crawford, and J. Tian, "Local manifold learning-based k-nearest-neighbor for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 11, pp. 4099–4109, October 2010.

[14] F. van der Meer, "The effectiveness of spectral similarity measures for the analysis of hyperspectral imagery," *Int. J. Appl. Earth Observ. Geoinform.*, vol. 8, no. 1, pp. 3–17, 2006.

[15] Y. Sohn and N. S. Rebello, "Supervised and unsupervised spectral angle classifiers," *Photogramm. Eng. Remote Sens.*, vol. 68, no. 12, pp. 1271–1282, 2002.

[16] Y. Du, C.-I. Chang, H. Ren, C.-C. Chang, J. O. Jensen, and F. M. DAmico, "New

hyperspectral discrimination measure for spectral characterization," *Opt. Eng.*, vol. 43, no. 8, pp. 1777–1786, 2004.

[17] Y. Sohn, E. Moran, and F. Gurri, "Deforestation in north-central yucatan(1985-1995)-mapping secondary succession of forest and agricultural land use in sotuta using the cosine of the angle concept," *Photogramm. Eng. Remote Sens.*, vol. 65, pp. 947–958, 1999.

[18] D. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, no. 12, pp. 2639–2664, 2004.

[19] T. Kim, J. Kittler, and R. Cipolla, "Discriminative learning and recognition of image set classes using canonical correlations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1005–1018, 2007.

[20] Y. Ma, S. Lao, E. Takikawa, and M. Kawade, "Discriminant analysis in correlation similarity measure space," in *Proc. Int. Conf. Mach. Learn.*, 2007, pp. 577–584.

[21] L. Samaniego, A. Bárdossy, and K. Schulz, "Supervised classification of remotely sensed imagery using a modified k-nn technique," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2112–2125, May 2008.

[22] S. D. Zenzo, R. Bernstein, S. Degloria, and H. Kolsky, "Gaussian maximum likelihood and contextual classification algorithms for multicrop classification," *IEEE Trans. Geosci. Remote Sens.*, vol. GE-25, no. 6, pp. 805–814, January 1987.

[23] S. Tadjudin and D. A. Landgrebe, "Robust parameter estimation for mixture model," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 1, pp. 439–445, August 2000.

[24] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, August 2004.

[25] R. Archibald and G. Fann, "Feature selection and classification of hyperspectral images with support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 674–677, October 2007.

[26] J. A. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, February 2009.

[27] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. Int. Conf. Comput. Vis.*, Barcelona, Spain, 2011, pp. 471–478.

[28] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, October 2011.

[29] Y. Chen, N. M. Nasrabadiand, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," in *Proc. Int. Conf. Image Process.*, Baltimore, MD, 2011, pp. 1233–1236.

[30] W. Li, E. W. Tramel, S. Prasad, and J. E. Fowler, "Nearest regularized subspace for

hyperspectral classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 477–489, March 2013.

[31] U. Srinivas, Y. Chen, V. Monga, N. M. Nasrabadi, and T. D. Tran, "Exploiting sparsity in hyperspectral image classification via graphical models," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 3, pp. 505–509, May 2013.

[32] J. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[33] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: a unifying view on subspace-based learning," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 376–383.

[34] M. Cui and S. Prasad, "Angular discriminant analysis for hyperspectral image classification," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1003–1015, April 2015.

[35] S. Prasad and M. Cui, "Sparse representations for classification of high dimensional multi-sensor geospatial data," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, 2013, pp. 811–815.

[36] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace ratio vs. ratio trace for dimensionality reduction," in *Proc. Int. Conf. Comput. Vis. Pattern Reco.*, 2007, pp. 1–8.

[37] F. Nie, S. Xiang, Y. Jia, C. Zhang, and S. Yan, "Trace ratio criterion for feature selection." in *Proc. AAAI Conf. Artif. Intell.*, vol. 2, 2008, pp. 671–676.

[38] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT press, 2002.

[39] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 586–597, 2007.

[40] J. Yang and D. Chu, "Sparse representation classifier steered discriminative projection," in *Proc. Int. Conf. Pattern Recog.*, 2010, pp. 694–697.

[41] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, "Spectral–spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2973–2987, 2009.

[42] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 3, pp. 809–823, 2012.

[43] M. Cui, S. Prasad, W. Li, and L. M. Bruce, "Locality preserving genetic algorithms for spatial-spectral hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 3, pp. 1688–1697, 2013.

[44] P. Gamba, "A collection of data for urban area characterization," in *Proc. Int. Geosci. Remote Sens. Symp.*, Anchorage, Alaska, September 2004, pp. 69–72.

[45] S. J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1 regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, 2007.

[46] T. Blumensath and M. E. Davies, "On the difference between orthogonal matching pursuit and orthogonal least squares," *Tech. Rep*, March 2007.

[47] L. Rebollo-Neira and D. Lowe, "Optimized orthogonal matching pursuit approach," *IEEE Signal Proces. Lett.*, vol. 9, no. 4, pp. 137–140, 2002.

[48] C. Soussen, R. Gribonval, J. Idier, and C. Herzet, "Joint k-step analysis of orthogonal matching pursuit and orthogonal least squares," *IEEE Trans. Inf. Theory*, vol. 59, no. 5, pp. 3158–3174, January 2013.

[49] S. Chen, "Local regularization assisted orthogonal least squares regression," *Neural Comput.*, vol. 69, no. 4, pp. 559–585, 2006.

[50] S. Chen, X. Hong, B. L. Luk, and C. J. Harris, "Orthogonal-least-squares regression: A unified approach for data modelling," *Neural Comput.*, vol. 72, no. 10, pp. 2670–2681, 2009.

[51] D.-S. Huang and W.-B. Zhao, "Determining the centers of radial basis probabilistic neural networks by recursive orthogonal least square algorithms," *App. Math. Comput.*, vol. 162, no. 1, pp. 461–473, 2005.

[52] G. Huang, S. Song, and C. Wu, "Orthogonal least squares algorithm for training cascade neural networks," *Pattern Reco.*, vol. 59, no. 11, pp. 2629–2637, 2012.

[53] L. Zhang, K. Li, E.-W. Bai, and G. W. Irwin, "Two-stage orthogonal least squares methods for neural network construction," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 26, no. 8, pp. 1608–1621, 2014.

[54] M. Cui and S. Prasad, "Class–dependent sparse representation classifier for robust hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2683–2695, September 2015.

[55] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[56] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, January 2006.

[57] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, February 1998.

[58] P. Vincent and Y. Bengio, "Kernel matching pursuit," *Mach. Learn.*, vol. 48, no. 1-3, pp. 165–187, 2002.

[59] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving discriminant analysis in kernel-induced feature spaces for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 5, pp. 894–898, 2011.

[60] D. P. Wipf and B. D. Rao, "Sparse bayesian learning for basis selection," *IEEE Trans. Signal Proces.*, vol. 52, no. 8, pp. 2153–2164, 2004.

[61] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[62] J. A. Tropp, A. C. Gilbert, and M. J. Strauss, "Algorithms for simultaneous sparse

approximation. part i: Greedy pursuit," *IEEE Trans. Signal Proces.*, vol. 86, no. 3, pp. 572–588, 2006.

[63] Y. C. Eldar, P. Kuppinger, and H. Bölcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Proces.*, vol. 58, no. 6, pp. 3042–3054, 2010.

[64] H.-G. Sohn and K.-H. Yun, "Shadow-effect correction in aerial color imagery," *Photogramm. Eng. Remote Sens.*, vol. 74, no. 5, pp. 611–618, 2008.

[65] Y. Li, P. Gong, and T. Sasagawa, "Integrated shadow removal based on photogrammetry and image analysis," *Int. J. Remote Sens.*, vol. 26, no. 18, pp. 3911–3929, 2005.

[66] S. M. Adler-Golden, R. Y. Levine, M. W. Matthew, S. C. Richtsmeier, L. S. Bernstein, J. H. Gruninger, G. W. Felde, M. L. Hoke, G. P. Anderson *et al.*, "Shadow-insensitive material detection/classification with atmospherically corrected hyperspectral imagery," in *SPIE*. International Society for Optics and Photonics, 2001, pp. 460–469.

[67] A. V. Kanaev and J. Murray-Krezan, "Spectral anomaly detection in deep shadows," *Appl. Opt.*, vol. 49, no. 9, pp. 1614–1622, Mar 2010. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-49-9-1614

[68] Z. Ben Rabah, I. R. Farah, G. Mercier, and B. Solaiman, "A new method to change illumination effect reduction based on spectral angle constraint for hyperspectral image unmixing," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 6, pp. 1110–1114, 2011.

[69] R. Amin, D. Lewis, R. W. Gould, W. Hou, A. Lawson, M. Ondrusek, and R. Arnone,

"Assessing the application of cloud–shadow atmospheric correction algorithm on hico," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2646–2653, 2014.

[70] M. Shimoni, G. Tolt, C. Perneel, and J. Ahlberg, "Detection of vehicles in shadow areas using combined hyperspectral and lidar data," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2011, pp. 4427–4430.

[71] J. Meola, M. T. Eismann, R. L. Moses, and J. N. Ash, "Application of model-based change detection to airborne vnir/swir hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 10, pp. 3693–3706, 2012.

[72] X. Huang and L. Zhang, "Morphological building/shadow index for building extraction from high-resolution imagery over urban areas," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 1, pp. 161–172, 2012.

[73] S. L. Ozesmi and M. E. Bauer, "Satellite remote sensing of wetlands," *Wetlands ecology and management*, vol. 10, no. 5, pp. 381–402, 2002.

[74] P. Rosso, S. Ustin, and A. Hastings, "Mapping marshland vegetation of san francisco bay, california, using hyperspectral data," *Int. J. Remote Sens.*, vol. 26, no. 23, pp. 5169–5191, 2005.

[75] E. Belluco, M. Camuffo, S. Ferrari, L. Modenese, S. Silvestri, A. Marani, and M. Marani, "Mapping salt-marsh vegetation by multispectral and hyperspectral remote sensing," *Remote Sens. Environ.*, vol. 105, no. 1, pp. 54–67, 2006.

[76] R. Zomer, A. Trabucco, and S. Ustin, "Building spectral libraries for wetlands land

cover classification and hyperspectral remote sensing," *J. Environ. Manage.*, vol. 90, no. 7, pp. 2170–2177, 2009.

[77] C. Hladik, J. Schalles, and M. Alber, "Salt marsh elevation and habitat mapping using hyperspectral and lidar data," *Remote Sens. Environ.*, vol. 139, pp. 318–330, 2013.

[78] E. Adam, O. Mutanga, and D. Rugege, "Multispectral and hyperspectral remote sensing for identification and mapping of wetland vegetation: a review," *Wetlands Ecology and Management*, vol. 18, no. 3, pp. 281–296, 2010.

[79] V. Klemas, "Remote sensing of coastal wetland biomass: an overview," *J. Coastal Research*, vol. 29, no. 5, pp. 1016–1028, 2013.

[80] J. Senthilnath, S. Omkar, V. Mani, N. Karnwal, and P. Shreyas, "Crop stage classification of hyperspectral data using unsupervised techniques," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 861–866, 2013.

[81] M. Gianinetto and G. Lechi, "The development of superspectral approaches for the improvement of land cover classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 11, pp. 2670–2679, 2004.

[82] W. Di and M. M. Crawford, "Active learning via multi-view and local proximity co-regularization for hyperspectral image classification," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 3, pp. 618–628, 2011.

Appendix A

Appendix

A.1 Proof of Proposition 1

$\hat{S}^{(\mathrm{w})}$ can be reformulated as

$$\begin{aligned}
\hat{S}^{(\mathrm{w})} &= \sum_{l=1}^{c} \sum_{i:y_i=l} \| \boldsymbol{T}^t \boldsymbol{x}_i - \boldsymbol{T}^t \boldsymbol{\mu}_l \|^2 \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} \mathrm{tr}\left((\boldsymbol{T}^t \boldsymbol{x}_i - \boldsymbol{T}^t \boldsymbol{\mu}_l)(\boldsymbol{T}^t \boldsymbol{x}_i - \boldsymbol{T}^t \boldsymbol{\mu}_l)^t \right) \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} \mathrm{tr}\left(\boldsymbol{T}^t \boldsymbol{x}_i \boldsymbol{x}_i^t \boldsymbol{T} + \boldsymbol{T}^t \boldsymbol{\mu}_l \boldsymbol{\mu}_l^t \boldsymbol{T} - 2\, \boldsymbol{T}^t \boldsymbol{x}_i \boldsymbol{\mu}_l^t \boldsymbol{T} \right) \\
&= \sum_{l=1}^{c} \sum_{i:y_i=l} (\| \boldsymbol{T}^t \boldsymbol{x}_i \|^2 + \| \boldsymbol{T}^t \boldsymbol{\mu}_l \|^2) - 2\, \mathrm{tr}(\boldsymbol{T}^t \bar{\boldsymbol{O}}^{(\mathrm{w})} \boldsymbol{T}),
\end{aligned} \tag{A.1}$$

where $\bar{\boldsymbol{O}}^{(\mathrm{w})} = \sum_{l=1}^{c} \sum_{i:y_i=l} \|\boldsymbol{x}_i\| \|\boldsymbol{\mu}_l\| \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{\mu}}_l^t$.

Similar to $\hat{S}^{(\mathrm{w})}$, $\hat{S}^{(\mathrm{b})}$ can be reformulated as

$$\hat{S}^{(\mathrm{b})} = \sum_{l=1}^{c} n_l (\| \boldsymbol{T}^t \boldsymbol{\mu}_l \|^2 + \| \boldsymbol{T}^t \boldsymbol{\mu} \|^2) - 2\, \mathrm{tr}(\boldsymbol{T}^t \bar{\boldsymbol{O}}^{(\mathrm{b})} \boldsymbol{T}), \tag{A.2}$$

where $\bar{\boldsymbol{O}}^{(\mathrm{b})} = \sum_{l=1}^{c} n_l \|\boldsymbol{\mu}_l\| \|\boldsymbol{\mu}\| \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{\mu}}^t$.

A.2 Proof of Proposition 2

Let z_i denote the cluster label of $\boldsymbol{x}_i$. $\boldsymbol{O}^{(\mathrm{lw})}$ and $\boldsymbol{O}^{(\mathrm{lb})}$ defined in (2.32) and (2.33) can

be reformulated as

$$\begin{aligned}
\boldsymbol{O}^{(\mathrm{lw})} &= \sum_{i,j=1}^{n} \tilde{W}_{ij}^{(\mathrm{lw})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \\
&= \sum_{l=1}^{c} \frac{1}{n_l} \sum_{i,j:y_i,y_j=l} \tilde{A}_{ij} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t + 0 \sum_{i,j:y_i \neq y_j} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \\
&= \sum_{l=1}^{c} \frac{1}{n_l} \Big(\sum_{\substack{i,j:y_i,y_j=l; \\ z_i=z_j}} \tilde{A}_{ij} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t + \sum_{\substack{i,j:y_i,y_j=l; \\ z_i \neq z_j}} \tilde{A}_{ij} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \Big) \text{ and} \qquad \text{(A.3)} \\
\boldsymbol{O}^{(\mathrm{lb})} &= \sum_{i,j=1}^{n} \tilde{W}_{ij}^{(\mathrm{lb})} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \\
&= \sum_{l=1}^{c} \Big(\frac{1}{n} - \frac{1}{n_l} \Big) \Big(\sum_{\substack{i,j:y_i,y_j=l; \\ z_i=z_j}} \tilde{A}_{ij} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t + \sum_{\substack{i,j:y_i,y_j=l; \\ z_i \neq z_j}} \tilde{A}_{ij} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t \Big) \\
&\qquad + \frac{1}{n} \sum_{i,j:y_i \neq y_j} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_j^t. \qquad \text{(A.4)}
\end{aligned}$$

In $\boldsymbol{O}^{(\mathrm{w})}$ and $\boldsymbol{O}^{(\mathrm{b})}$, $\tilde{A}_{ij} = 1$ for all pairs of within-class samples. Thus based on (A.3) and (A.4), $\boldsymbol{O}^{(\mathrm{w})}$ and $\boldsymbol{O}^{(\mathrm{b})}$ can be defined as

$$\begin{aligned}
\boldsymbol{O}^{(\mathrm{w})} &= \sum_{l=1}^{c} \frac{1}{n_l} \Big(\sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t + \sum_{k \neq q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t \Big) \text{ and} \qquad \text{(A.5)} \\
\boldsymbol{O}^{(\mathrm{b})} &= \sum_{l=1}^{c} \Big(\frac{1}{n_l} - \frac{1}{n} \Big) \Big(\sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t + \sum_{k \neq q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t \Big) \\
&\qquad + \frac{1}{n} \sum_{l \neq m} n_l n_m \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{\mu}}_m^t. \qquad \text{(A.6)}
\end{aligned}$$

Define $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$ as neighbors if $\|\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}}_j\| \leq \epsilon$, where ϵ is the radius of a hypersphere around a sample $\tilde{\boldsymbol{x}}_i$ that defines the neighborhood of $\tilde{\boldsymbol{x}}_i$. For simplicity, let $\tilde{A}_{ij} = 1$ if within-class sample pair $\tilde{\boldsymbol{x}}_i$ and $\tilde{\boldsymbol{x}}_j$ are neighbors and $\tilde{A}_{ij} = 0$ otherwise. It indicates within-class samples from different clusters are not neighbors of each other on a unit hypersphere,

yielding $\tilde{A}_{ij} = 0$ for $z_i \neq z_j$, then $\boldsymbol{O}^{(\mathrm{lw})}$ and $\boldsymbol{O}^{(\mathrm{lb})}$ can be simplified to

$$\boldsymbol{O}^{(\mathrm{lw})} = \sum_{l=1}^{c} \frac{1}{n_l} \sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t \text{ and} \tag{A.7}$$

$$\boldsymbol{O}^{(\mathrm{lb})} = \sum_{l=1}^{c} (\frac{1}{n_l} - \frac{1}{n}) \sum_{k=q} n_{lk} n_{lq} \tilde{\boldsymbol{\mu}}_{lk} \tilde{\boldsymbol{\mu}}_{lq}^t + \frac{1}{n} \sum_{l \neq m} n_l n_m \tilde{\boldsymbol{\mu}}_l \tilde{\boldsymbol{\mu}}_m^t. \tag{A.8}$$

Note that we can relax the strict definition of neighborhood (based on the choice of $\tilde{A}_{ij}$ above) by utilizing smooth functions to generate the affinity matrix (e.g., the heat kernel). Proposition 2 would still hold in an approximate sense for such a choice.