INFORMATION FUSION FOR MULTI-SOURCE DATA CLASSIFICATION

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

Yuhang Zhang

December 2015

INFORMATION FUSION FOR MULTI-SOURCE DATA CLASSIFICATION

Yuhang Zhang

Approved:

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

Co-Chair of the Committee
Dr. Jose L. Contreras-Vidal, Professor,
Dept. of Electrical and Computer Engineering

Committee Members:

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

Dr. Demetrio Labate, Professor,
Dept. of Mathematics

Dr. Melba M. Crawford, Professor,
Dept. of Civil Engineering, Purdue University

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

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

Acknowledgements

First, I would express my sincere gratitude to my advisor Dr. Saurabh Prasad and co-advisor Dr. Jose Luis Contreras-Vidal, who are invaluable for constantly motivating me to explore my capability of doing research, inspiring me to provide innovative ideas. With their guidance, I not only learned how to efficiently accomplish my PhD research, but also how to become a diligent, smart and fruitful researcher. Besides my advisors, I want to thank, Dr. Melba Crawford for her guidance on our projects and collaborative papers, Dr. Badri Roysam and Dr. Demetrio Labate for their guidance on the specific research areas. Thank you all my committee members for taking time to serve in my dissertation defense and sharing their valuable advices to improve this dissertation.

I would thank my dear colleagues from both Hyperspectral Image Analysis Laboratory and Non-invasive Brain Machine Interface Laboratory. I enjoy the research atmosphere that we help, encourage and learn from each other. I appreciate all of your valuable comments and suggestions on my research. Thank you to all of my friends. Three years ago, I first set my foot on the land of the United States. Without their help, I could not quickly adapt to the life here. Without their company, I would not enjoyed my life at Houston.

Last but not least, I would like to thank my parents, who always give me unconditional support and love. Thank you to my dear fiance, who is not only a partner but also a friend, a mentor in life. Whenever I encountered difficulties and feel frustrated, their encouragements were always the best thing to help me get through them. Without their love, I could not have today's achievement.

INFORMATION FUSION FOR MULTI-SOURCE DATA CLASSIFICATION

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

Yuhang Zhang

December 2015

Abstract

Multi-source data, either from different sensors or disparate features extracted from the same sensor, are often valuable for data analysis due to their potential for providing complementary information. Effective fusion of information from such multi-source data is critical to enhanced and robust interpretation about the underlying classification problem. Nevertheless, multi-source data also bring unique challenges for data processing, e.g., high-dimensional features, lack of compact representation, and insufficient quantity of labeled data. To make the most use of multi-source data and to address the above challenges, in this research, we develop and validate data fusion algorithms on multiple datasets in two active research areas — remote sensing and brain machine interface (BMI).

We develop a mixture-of-kernels approach for data fusion, and demonstrate its efficacy at fusion of multi-source data in the kernel space. In the proposed approach, each source of data is represented by a dedicated kernel — one can then learn a classifier (or an "optimal" feature subspace) by optimizing the kernel parameters for maximum discriminative potential. A direct related benefit is that this learning framework provides a natural and automated mechanism to learn weight distributions in the weighted mixture of kernels, that are strongly indicative of strengths and weaknesses of various sources in the underlying multi-source data analysis problem. We illustrate the benefit of this property and apply it to infer the relative importance of different sources of information in a BMI application. Additionally, to save the labor of labeling a large quantity of samples in real world remote sensing applications, an ensemble based multiple kernel active learning framework is proposed to effectively select important unlabeled samples from multi-source data for classification. We also propose a multi-source feature extraction method based on a composite kernel mapping, to project the multi-source data to a lower dimensional subspace for effective feature fusion.

Finally, to effectively represent multi-source data in a compact and robust manner, we propose a joint sparse representation model with adaptive locality weights for classification. By adapting the penalty on individual atoms in the dictionary, we show that one can achieve better signal representation and reduce estimation errors. Further, we also develop a kernel variant of the proposed fusion framework, which is conceptually consistent and aligned with the mixture-of-kernels approach developed previously.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Multi-source Data Fusion for Classification

Multi-source data, either from different sensors or disparate features extracted from the same sensor, are valuable for data analysis due to their potential for providing complementary features. To effectively combine data from multiple sources to improve the interpretation performances of individual data sources, data fusion has been studied in a variety of fields, e.g., signal detection, object recognition, tracking, change detection and classification, for different applications, e.g., computer vision, remote sensing, medical analysis and defence security.

In general, data fusion techniques can be categorized into three levels [1, 2] — pixel/data level, feature level and decision level. A more detailed classification of data fusion techniques was provided by Dasarathy [3] by expending the three level hierarchy into five categories based on the input and output modes — (1) data in-data out (DI-DO), (2) data in-feature out (DI-FO), (3) feature in-feature out (FI-FO), (4) feature in-decision out (FI-DO), (5) decision in-decision out (DI-DO).

The data/pixel level fusion, also known as low level fusion, is the most basic data fusion method. The raw data are directly provided from multiple sources as input to the data fusion process, and the output provides a single resolution data, which are expected to be more informative than each individual input.

In feature fusion, a medium level fusion, features extracted from raw data are fused to obtain new features that could be employed for further processing. Note that in image processing,

such fusion requires a precise (pixel-level) registration of the available images. Methods applied to extract features usually depend on the characteristics of the individual data streams, and therefore may have very different properties if the data sets used are heterogeneous. Feature fusion strategies vary greatly, depending upon properties of the given data and the ultimate goal.

Decision fusion is a high level fusion approach, which takes symbolic representations as input and combines them to obtain a more accurate decision output. When the results from different algorithms are expressed as confidence measures (or scores) rather than decisions, it is called soft fusion; otherwise, when only label information is used, it is referred to as hard fusion. Typical decision fusion approaches include voting-based methods, statistical or probability-based methods and fuzzy logic-based methods.

A problem of particular interest in multi-source data fusion applications is classification, where the ultimate question is how to take advantage of related information from different sources representing the same physical quantity to achieve robust classification. A variety of fusion approaches have been proposed in the literature for classification, and most methods fall into the categories of feature or decision level fusion, particularly as DI-DO and FI-FO [4].

The focus of this research is to design data fusion algorithms (particularly feature fusion and decision fusion) for robust multi-source data classification. The underlying assumption is that following effective fusion, multi-source data would have a better representation in the feature space and their classification/interpretation performance is improved compared to any one individual data source. The research are mainly conducted for two specific applications, i.e., remote sensing (RS) image classification and brain machine interface (BMI).

1.2 Applications

1.2.1 Remote Sensing

Applications of multi-source remote sensing data for earth observation and analysis have been an active research topic in recent years, due to the rapid development of relevant remote sensing technologies, e.g., very high resolution (VHR) optical, multispectral and hyperspectral sensors, Synthetic Aperture Radar (SAR), and Light Detection and Ranging (LiDAR) systems. Remote sensing data fusion aims to integrate the information acquired with different spatial and spectral resolutions from sensors mounted on satellites, aircraft and ground platforms to produce fused data that contain more information than each of the sources individually.

Hyperspectral imagery has been increasingly applied for material classification. The hyperspectral sensors are designed to capture digital images in hundreds of narrow spectral bands ranging from the visible to the infrared spectrum [5]. The spatially and spectrally sampled information in hyperspectral imagery can be considered as a data cube, wherein the spatial coordinates provide image information, and the depth represents intensity as a function of spectral bands (or wavelength). Consequently, any hyperspectral image has a three-dimensional (3D) data structure. Each plane (spectral band) of the cube is a grayscale image that represents the spatial distribution of the scene's reflectance in the corresponding spectral wavelength. Along the wavelength dimension, each image pixel provides a spectrum characterizing the materials within the pixel. Different materials typically reflect electromagnetic energy differently at specific wavelengths, which makes discrimination of materials possible based on the spectral characteristics.

Nevertheless, using spectral information from hyperspectral images may not provide comprehensive information of ground objects for certain applications. Recent studies have shown

that the spatial information (e.g., textural, contextual and morphological) can be utilized for a variety of image analysis tasks [6, 7]. Additionally, LiDAR, an active optical remote sensing modality is now becoming increasingly popular for discriminating different ground classes, particularly where topographic variations are important, and has been used for multiple applications, e.g., landscape level analysis of salt marsh plant habitats [8] and large area ecosystem characterization [9]. The fusion of multi-source geospatial data, such as hyperspectral images with spatial features or LiDAR data, could potentially provide more information than using either sensor by itself [10, 11].

Many techniques have been developed to process and fuse features from different sources. Markov random fields (MRF) and its variants have been widely used to model contextual constraints and combine multi-source information for remote sensing applications [12–15]. In [10], the authors investigated the joint use of hyperspectral and LiDAR data for the classification of complex forest areas, based on support vector machine (SVM) and Gaussian maximum likelihood with leave-one-out-covariance algorithm (GML-LOOC) classifiers. In [16], the authors proposed to fuse spectral, spatial and elevation data by applying morphological attribute profiles for urban area classification. In [17], multi-source information was exploited through the use of composite kernels in SVM. The composite kernel SVM has been demonstrated to be an effective strategy to combine different sources of data [18].

1.2.2 Brain Machine Interface

Brain Machine Interface (BMI) or Brain Computer Interface (BCI) systems have attracted extensive attention in the past decade because of their potential in improving human life, especially for those who are affected by motor disabilities, e.g., stroke, paraplegia, and quadriplegia. A BMI system is designed to communicate between a subject and the external device without involving any peripheral and muscular activity [19–22]. Previously, BMI (BCI) systems were

mainly employed to control external devices such as computer cursors [23] and robotic prostheses/orthoses [24] using invasive methods. In recent studies, BMIs have been used to control lower-body or upper-body exoskeletons for stroke and paraplegic recovery and rehabilitation via non-invasive approaches [25, 26]. To control a device via BMI, different brain activity patterns produced by a user need to be accurately identified by a neural interface system and translated into appropriate commands. To achieve this goal, the design of high performance decoding system have become a research focus in recent years.

Electroencephalography (EEG) which records brain signals along the scalp generated by the concerted action of millions of cortical cells is an attractive method for developing non-invasive clinical BMI systems. From a machine learning point of view, decoding of EEG signal is a challenging task for several reasons. First, the EEG signal is non-stationary and commonly contaminated by artifacts. Artifacts can come from physiological sources, such as eye blinks, muscle activities, and mechanical sources, such as motion of electrodes or cables during use. Although advanced signal processing techniques, such as Independent Component Analysis (ICA), have been employed to remove contaminants, substantial noise still co-exists with the "pure" signal. EEG signal is also non-stationary since it may vary rapidly over time and more critically over sessions. In addition, the quality of the data is often affected by the extent of concentration of the subject, medication or even the mood of the subject during data recording, which may result in changing of patterns in the data [27]. Secondly, EEG data has low spatial resolution. It is still unclear which areas of the brain contribute most to a particular response. Thus, intensive interpretations are required for decoders. Thirdly, the feature space resulting from EEG data can be potentially high dimensional, while at the same time, the number of training examples is limited as collecting labelled data is time consuming and a cognitively demanding process for the subjects. This brings about difficulties in designing a robust and

effective classifier, because traditional statistical classifiers are usually sensitive to the quantity of the labeled data, and easily affected by the "curse of dimensionality."

As the sensor technology develops, there is a trend to collect different sources of data simultaneously from the subject with multiple sensors. In [28], a novel protocol was presented for non-invasive collection of brain activity (EEG), muscle activity (electromyography (EMG)), and whole-body kinematic data (head, torso, and limb trajectories) during both treadmill and over ground walking tasks. In [29], scalp EEG and Kalman decoders were used to infer both kinematics and the surface EMG patterns of stroke patients wearing a robotic exoskeleton, which opens a window of opportunity to combine source signals as well as control variables, e.g., prediction of movement kinematics could be used to control the robotic exoskeleton while prediction of sEMG patterns could be used to drive functional electrical stimulation (FES) system in parallel.

Data fusion can also be developed for effective fusion of multichannel data. It has been shown that several multichannel fusion models are able to exploit the different but complementary brain activity information for robust decoding [30, 31]. A parametric weighted decision fusion model and two parametric weighted data fusion models were introduced for the classification of averaged multichannel evoked potentials in [30]. In [31], the authors proposed and compared two multichannel fusion schemes, e.g. multichannel feature fusion and decision fusion to utilize the information extracted from simultaneously recorded multiple EEG channels. In [32], a framework based on decision fusion was proposed for multimodal neural prosthetic devices, in which the Kalman filter and ANNs were examined in the context of decoding 2-dimensional endpoint trajectories of a neural prosthetic arm. Testing results show that both fusion algorithms successfully fused the individual decoder estimates to produce more accurate predictions, which suggests an interesting direction for decoding using multimodal neural data

in the future.

1.3 Research Motivations and Challenges

The aim of this research is to effectively incorporate disparate features from multiple sources to provide complementary information for accurate classification. Although various fusion algorithms have been proposed and successfully applied for different types of data, there are some challenges in relation to remote sensing and BMI applications, which can be summarized as follows.

1. High Dimensionality

Multi-source data can provide a wealth of complementary information; however, the high-dimensional data often brings unique challenges for data analysis. Hyperspectral sensors typically oversample the spectral signal to ensure that any narrow features are adequately represented, which results in hundreds of spectral bands in hyperspectral data. High dimensionality is also a concern for EEG signal processing. Feature spaces can be high-dimensional considering the number of electrodes used in the experiment as well as the number of features extracted from each channel. For example, if we apply a 200 ms sliding window to extract the EEG delta-band amplitude modulation information from 64 channels for limb movement decoding, the resulting features are in a 1280 dimensional space (assuming a sampling frequency of 100 Hz).

The high-dimensional data can bring about difficulties in designing a robust and effective classifier, because traditional statistical classifiers are usually sensitive to the quantity of the labeled data, and easily affected by the "curse of dimensionality." On the other hand, high-dimensional space is mostly empty, in which data can be projected to a lower dimensional subspace without losing significant information in terms of separability among the different statistical classes. Robust feature extraction or dimensionality reduction methods are hence

necessary to obtain useful information in a much lower dimensional space that allows for the separation of classes.

2. Effective Representation

Another challenge in the signal processing point of view is how to represent the signal in a compact way. As most natural signals are inherently sparse in a certain basis or dictionary, they can be compactly represented by only a few coefficients that carry the most important information. In other words, the intrinsic signals in the same class usually lie in a low-dimensional subspace and the semantic information is often encoded in a sparse representation with respect to some proper basis.

Many biological findings support sparse representation in the brain, as sparsity of the neural response has been observed in neurons [33, 34] — a sparse set of neurons only encode specific concepts rather than responding to every input. In [35], it was found that EEG signals had a sparse spatial and temporal structure which may be utilized to improve signal classification for BMIs across multiple subjects and modalities.

In recent years, the compressed sensing and sparse representation theories have emerged as powerful tools to reconstruct and represent signals by decomposing the sample over a usually overcomplete dictionary generated by or learned from representative samples. Further, sparse representation based classification (SRC), which combines the discrimination power with the reconstruction property and notions of sparsity, has been demonstrated as an effective and robust method for many pattern recognition applications [35–40].

3. Limited labeled samples

Given sufficient labeled ground reference data, supervised learning methods are effective for analysis of remote sensing data, for problems including classification, spectral unmixing and anomaly detection. Unfortunately, the performance of supervised learning models is heavily

dependent on the availability of representative labeled data for training, which in real-word applications are usually expensive and time-consuming to obtain. However, manual selection of training data from imagery, a common practice, is subjective (particularly if accomplished via visual interpretation) and tends to introduce redundancy into the supervised classifier because data are selected in spatially contiguous patches, and thus slow the training process. Therefore, it is important to collect training data that are most informative and useful for the underlying classification task.

Active learning (AL) was introduced for such tasks in the machine learning community [41], and has been demonstrated to be useful for classification of remote sensing data [42], [43]. Unlike traditional passive learning, where labeled data are used to train the classifier and unlabeled samples are subsequently classified, in AL, users can interact with the classifier, both providing capability to select the most informative samples and allowing adaptation in dynamic environments. In the AL framework, classifiers are initially trained on a very limited set of training samples, but additional informative and representative samples are identified from the abundant unlabeled data, labeled, and then inducted into this set, thereby growing the training dataset in a systematic way. The goal of AL is to minimize the cost related to the sample labeling process while maximizing the discrimination capabilities.

1.4 Dissertation Overview

To make the most use of multi-source data for classification and to address the challenges stated above, this research proposes advanced fusion methods from two different aspects — mixture-of-kernels based approaches and sparse representation based approaches. To achieve enhanced performance for multi-source data fusion, it is important to leverage individual sources according to their information contribution and assign each source an optimal weight for fusion. This research investigates different methods to optimize the weights in the proposed

algorithms.

The dissertation first provides background for this research and gives reviews of basic classification and multi-source data learning methods in Chapter 2. In Chapter 3, we present methods to fuse different sources of data in the kernel space through mixture of kernels, where each kernel is dedicated to a particular type of source/feature. Within this chapter, a composite-kernel-based feature extraction method is first proposed as a feature fusion method. In the composite kernel based approach, the weights are optimized by grid search and cross-validation. Further, the multiple kernel learning (MKL), a more sophisticated method to optimize the kernel weights, is applied for scalp region importance learning in BMI tasks. In this framework, user's internal states, such as the gait patterns, can be decoded from the EEG signals and the relative importance of different scalp brain areas can be simultaneously learned by an MKL optimization. In addition, to address the challenge of a limited quantity of labeled samples in real applications, this chapter presents a multi-source active learning framework based on MKL, which can be used to iteratively select important unlabeled samples to enlarge the training set for better classification. In Chapter 4, a data driven joint sparse representation model with adaptive weights for different sources is proposed. By adapting penalty on different atoms, one can not only achieve better signal representation but also reduce the estimation errors. The proposed fusion framework is then extended to the kernel space, which leads to a conceptually similar framework as the multi-kernel based approach. The proposed works are validated on multiple datasets in two active research areas — remote sensing and brain machine interface, for different applications.

Figure 1.1 shows the structure of the dissertation and the proposed methods that are developed and presented in each chapter.

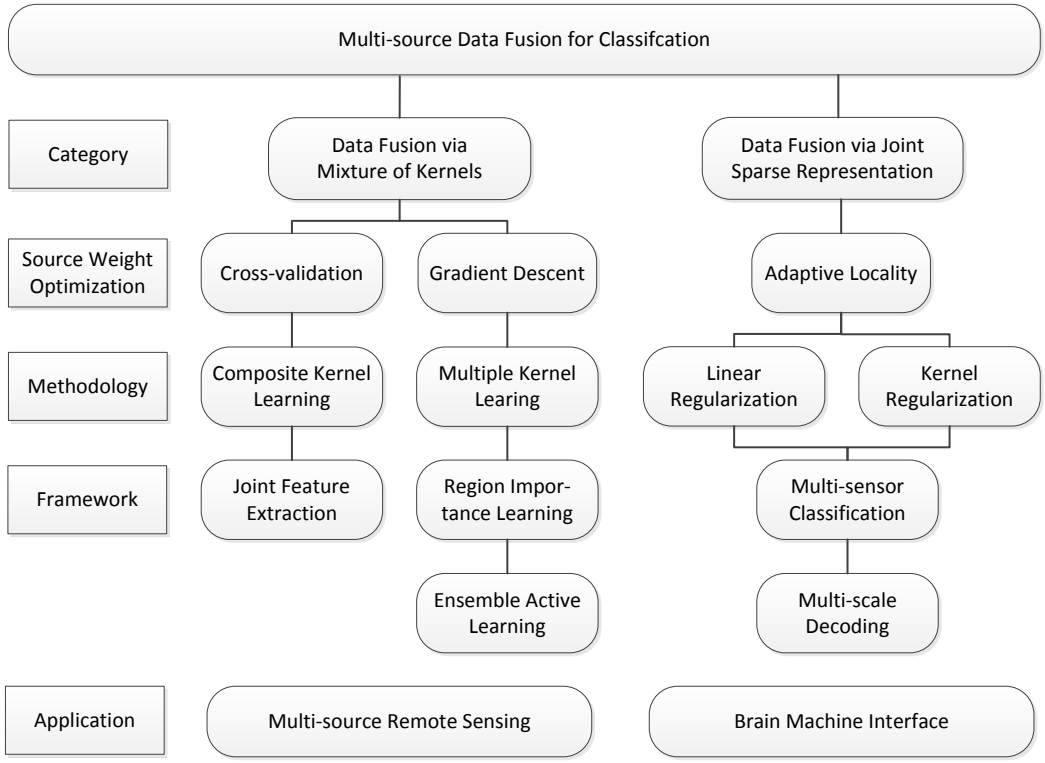


Figure 1.1: Dissertation structure

Chapter 2

Background and Related Work

2.1 Introduction

Feature extraction and classification are two key aspects of most machine learning systems. Feature extraction seeks to find features that facilitate optimal discrimination between classes. This chapter first provides the background for the multi-source data employed in this research, and descriptions of several types of features extracted from each data source. Then the classification algorithms related to this research are reviewed. Based on traditional algorithms for single source data classification, the multi-source data learning and active learning methods are discussed to provide a sound background for the research.

2.2 Data Acquisition and Feature Extraction

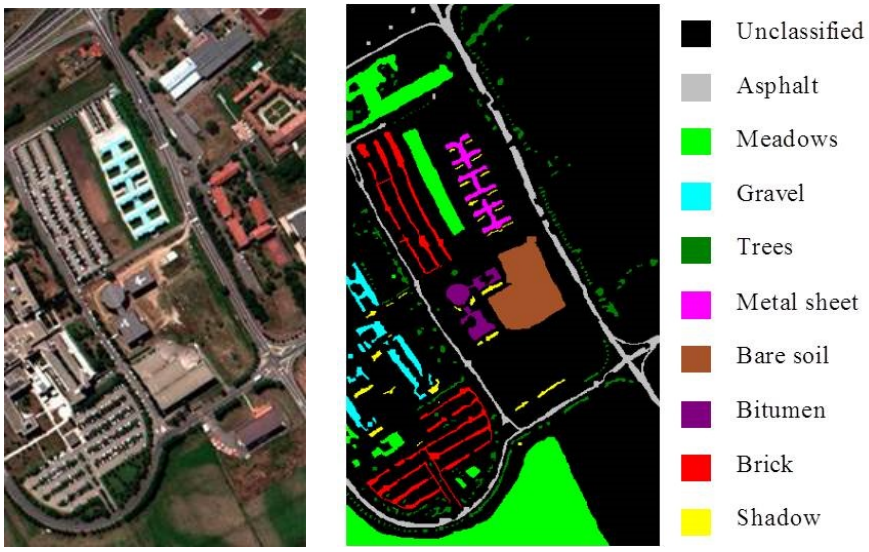
2.2.1 Remote Sensing Datasets

For the remote sensing application, the proposed algorithms are validated on two sets of multi-source data.

1. University of Pavia Dataset

The first hyperspectral dataset was acquired using the Reflective Optics System Imaging Spectrometer (ROSIS) sensor over an urban area surrounding the University of Pavia, Italy. The image has a spatial size of 610×340 pixels with the spatial resolution of 1.3 m per pixel and 103 spectral bands. There are 9 classes of interest, and the number of labeled samples for each class is shown in Table 2.1. The composite image of the hyperspectral data and its

groundtruth are shown in Figure 2.1. The mean spectral signatures are shown in Fig. 2.2, and the *critical* classes are determined as Asphalt, Bitumen, Meadows, Bare Soil, Gravel and Bricks.



(a) (b)

Figure 2.1: University of Pavia dataset. (a) Composite image of the hyperspectral data; (b) Groundtruth map.

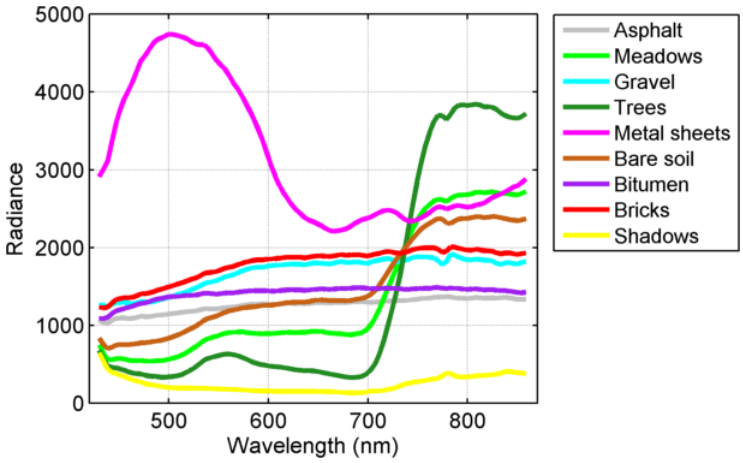


Figure 2.2: Mean spectral signatures of University of Pavia dataset.

Table 2.1: Groundtruth classes for the University of Pavia scene and their respective number of samples

Class Index	Name	# samples	Class Index	Name	# samples
1	Asphalt	6631	6	Bare Soil	5029
2	Meadows	18649	7	Bitumen	1330
3	Gravel	2099	8	Bricks	3682
4	Trees	3064	9	Shadows	947
5	Metal sheets	1345			

2. University of Houston (UH) Dataset

The second dataset is comprised of a hyperspectral image and discrete return LiDAR data. The dataset were acquired over the University of Houston campus and the neighboring urban area. The hyperspectral data were acquired with the ITRES-CASI 1500 sensor, on June 23, 2012 between 17:37:10 and 17:39:50 UTC. The average altitude of the sensor was 5500 ft, which resulted in 2.5 m spatial resolution data. The hyperspectral imagery consists of 144 spectral bands ranging from 380 nm to 1050 nm, and was processed (radiometric correction, attitude processing, GPS processing, geo-correction etc.) to yield the final geo-corrected image cube representing at-sensor spectral radiance, SRU = μW/(cm^2 sr nm). A true color composite of the hyperspectral signatures is shown in Figure 2.4 (a).

The LiDAR data were acquired using an Optech Gemini sensor on June 22, 2012 between 14:37:55 and 15:38:10 UTC. The 167 kHz laser operates at 1064 nm and records up to four returns. The average height of the sensor at the time of acquisition was 2000 ft above ground level, which resulted in an average point density of 35.38 points/m^2 on the ground. The digital surface model (DSM) and pseudo-waveform data were generated from the original LiDAR point cloud as described in Section 2.2.2 and are shown in Figure 2.4 (b) and (c), respectively. The pseudo-waveform data are represented in 80 bins, which correspond to the LiDAR aggregated discrete returns within pre-defined voxels at predetermined elevations above/below the ground. The quantization unit of elevation spacing is 1 m, and the first band corresponds to 9

m below ground elevation, so the 80-th band corresponds to 70 m above ground elevation. For this urban area, most objects are located in the range of $0-20$ m, so some bands are "empty". From Figure 2.4 (c), we can see most of the objects are in the green color range, which means that they are close to the ground. The mid-elevation objects are represented as blue, and taller objects are in red. All images are comprised of a 349×1905 grid at 2.5 m spatial resolution. The ground reference map is shown in Figure 2.4 (d). The total number of ground reference samples is 15,029, covering 15 classes of interest. The number of labeled samples for each class is shown in Table 2.2. The classes include several vegetation classes, different types of roads, as well as some urban classes. We identify Road, Highway, Railway, Parking Lot 1 and Parking Lot 2, as *critical* classes, because of their similar spectral signatures as shown in Fig. 2.3. In particular, parking lots are categorized based on whether they have cars, i.e. Parking Lot 1 is empty, and Parking Lot 2 is filled with cars.

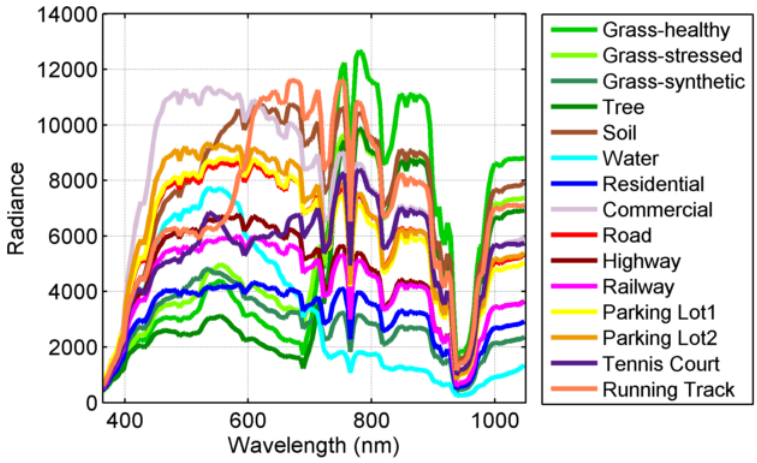


Figure 2.3: Mean spectral signatures of UH dataset.

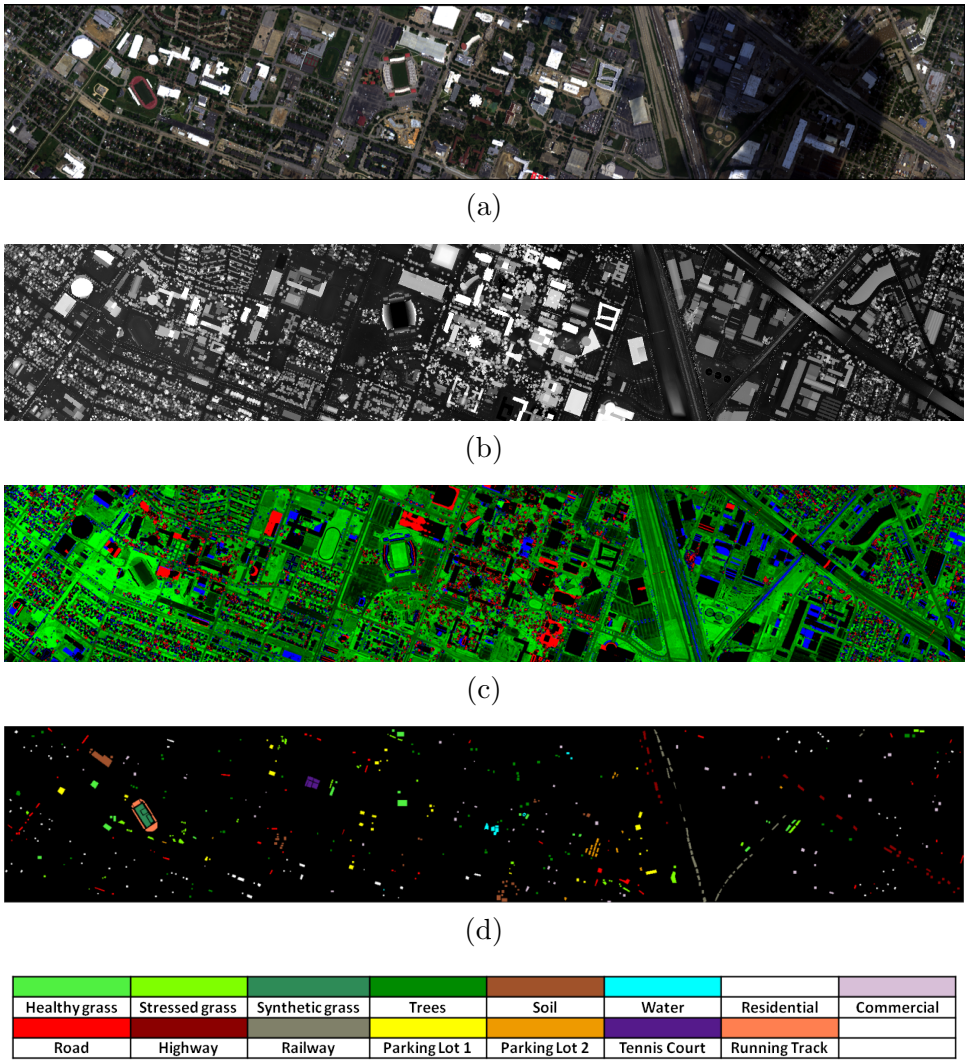


Figure 2.4: UH dataset. (a) True-color composite of the hyperspectral data; (b) LiDAR DSM data; (c) False-color composite of the pseudo-waveform data; (d) Groundtruth map.

Table 2.2: Groundtruth classes for the UH scene and their respective number of samples

Class Index	Name	# samples	Class Index	Name	# samples
1	Healthy grass	1251	9	Road	1252
2	Stressed grass	1254	10	Highway	1227
3	Synthetic grass	697	11	Railway	1235
4	Trees	1244	12	Parking Lot 1	1233
5	Soil	1242	13	Parking Lot 2	469
6	Water	325	14	Tennis Court	428
7	Residential	1268	15	Running Track	660
8	Commercial	1244			

2.2.2 Remote Sensing Multi-source Feature Extraction

We use the terminology multi-source to loosely refer to data obtained via different sensors, or different feature-types derived from the same sensor. For multi-sensor fusion, both hyperspectral and LiDAR data are used as the testbed. DSM and pseudo-waveforms were generated as feature streams for LiDAR data. For multi-feature fusion scenario, the raw spectral data and spatial features derived from the hyperspectral image are utilized. Two potentially diverse spatial features are considered in this research, which are object-based textural features and extended morphological attribute profiles (EMAPs). A brief description of each type of feature set is presented in the following sections.

2.2.2.1 Features from LiDAR Data

Among several data products that can be extracted from discrete return LiDAR data, DSM and pseudo-waveforms are considered in this research. A DSM is one of the most popular and simple data products that can be generated from the discrete return LiDAR data. In addition to the DSM, we also generate pseudo-waveforms over the same grid structure as used in the DSM and the hyperspectral data by stacking voxels with 1 m vertical dimension on the grid and accumulating points within every voxel. We refer readers to [44] for a detailed description of the pseudo-waveform generation process. Although the pseudo-waveform generation approach is adopted in this study since only discrete return LiDAR data are available over the study area, LiDAR data from advanced full waveform LiDAR systems could be used instead by applying waveform decomposition [45].

2.2.2.2 Object-based Texture Features

Spatial features are usually extracted by considering a window-based approach, which, however, suffers from the "border-effect" — an issue where the neighborhood includes pixels from multiple objects/thematic classes. This problem can be mitigated by following an object-based

approach, in which the neighborhood system is defined in an adaptive way. The approach entails three key steps — (1) The original data are subdivided into spatially homogeneous regions using the *HSeg algorithm* [46]. HSeg is a segmentation approach that combines region growing, which produces spatially connected regions, with clustering, which results in groupings based on spectral similarity from spatially disjoint regions. Two main factors influence the merging process: the dissimilarity criterion (DC) and the weighting factor S_{wght}, which ranges from 0 to 1 and sets the relative importance of spatially adjacent regions with respect to those that are nonadjacent. The output is a hierarchy of segmentation maps at different levels of detail; (2) Following this, an unsupervised strategy of pruning is applied to remove subtrees of the hierarchy that are homogeneous with respect to a given homogeneity criterion. In this way, the final segmentation does not represent one of the actual levels of the hierarchy, but incorporates regions potentially selected from different levels. This is accomplished by characterizing each region of the hierarchy in terms of second order statistics (standard deviation is considered in our specific case). The homogeneity criterion is calculated adaptively for each pixel by computing the standard deviation in a window with size $W = [w_x, w_y]$; (3) Finally texture features (mean and standard deviation are considered in our work) are extracted from the object-based detected regions.

2.2.2.3 Extended Multi-attribute Profiles

Morphological attribute filters have been applied to extract morphological features in many recent remote sensing applications [47], [48]. Profiles are computed by removing the connected components that do not fulfill a specified criterion. The value of an arbitrary attribute *attr* measured on a component is compared to a given reference value λ, e.g., $T(Com) = attr(Com) > \lambda$. If the criterion is satisfied, then the regions are kept intact; otherwise, they are set to the gray level of a darker or brighter surrounding region. Such attributes can be geometric (e.g. area,

shape, length of the perimeter, image moments), or textural (e.g. range, standard deviation, entropy), etc. Attribute profiles (APs) are an extension of the widely used morphological profiles (MPs). Analogous to the definition of the MPs, APs consist of n morphological attribute thickening (ϕ^T) and n attribute thinning (γ^T) operators as given by

$$\mathbf{AP}(f) = \{\phi_n^T(f), ..., \phi_1^T(f), f, \gamma_1^T(f), ..., \gamma_n^T(f)\} \ , \tag{2.1}$$

where f is the input image. Each AP can be computed on one of the features from a multivariate image (e.g., the first c principal components of a hyperspectral image), and different APs can be combined as an extended attribute profile (EAP). Also, according to the attributes considered, different morphological information can be extracted from the image, and merged into a single data structure denoted as EMAP

$$\mathbf{EMAP} = \left\{\mathbf{EAP_1}, \mathbf{EAP_2'}, ..., \mathbf{EAP_n'}\right\} \ , \tag{2.2}$$

where each **EAP** corresponds to a specific attribute and $\mathbf{EAP'}$ consists of all thickening and thinning operators, excluding the multiple presence of the input image f or the c principle components which have already been included in $\mathbf{EAP_1}$.

2.2.2.4 Parameter Setting

We retain original features (hyperspectral signatures and pseudo-waveforms) in our set of features. In order to extract textural features, the HSeg algorithm was applied to LiDAR pseudo-waveform and hyperspectral data by adopting four-neighborhood connectivity. For both cases, we considered a Spectral Angle Mapper (SAM) based dissimilarity criterion (DC) and fixed the parameter S_{wght} to 0.1. The strategy of segmentation hierarchy pruning was applied by setting the parameter $W = [w_x, w_y]$ equal to [3, 3]. The number of segments obtained after pruning LiDAR pseudo-waveform and hyperspectral data were 67,735 and 55,802, respectively.

EMAPs were extracted from LiDAR DSM data and hyperspectral signatures. While EMAP features can be computed directly from the single band LiDAR DSM data, for hyperspectral signatures they were extracted from the first four principal components which contain 99% of the total variance of the original data. Three APs were computed for both LiDAR DSM and hyperspectral data considering different attributes related to the geometry of a region. The AP associated with area is a surrogate for the scale of the structures in the scene, which is related to the size of the regions. The length of the diagonal of the bounding box is a different measure of the size and geometrical properties of region. The moment of inertia attribute, which models the elongation of the regions, is a measure of the noncompactness of the objects. As in [48], the values of λ used in each EAP are: 1) area of the regions, $\lambda_a = [100, 500, 1000, 5000]$; 2) length of the diagonal of the box bounding the region, $\lambda_d = [10, 25, 50, 100]$; and 3) moment of inertia, $\lambda_d = [0.2, 0.3, 0.4, 0.5]$. Note that the range of *optimal* parameters are expected to be data dependent. In this work, we experimentally determined these values to be appropriate. Thus, for the LiDAR DSM (hyperspectral) data, each EAP is 9 (36)-dimensional, i.e., it is composed of one (four) APs with nine levels computed on each component. The final EMAP is obtained by stacking the three EAPs into a single data structure and by considering the original LiDAR DSM (principal components) just one time.

2.2.3 EEG Data Acquisition

The restoration and rehabilitation of gait are of great interest to the field of BMIs, i.e. devices that utilize neural activity to control virtual or physical exoskeletons or prostheses. Since gait deficits are commonly associated with spinal cord injury, limb loss, and neurodegenerative diseases, there is a need to investigate innovative therapies to restore gait in such patients. In this research, the specific aim is to classify gait patterns (gait motions or gait phases in a gait cycle) into different classes using non-invasive EEG signals. The gait data employed in this

study were acquired for two applications.

1. BMI system for lower-limb exoskeleton control

This experimental protocols were approved by Institutional Review Board of the University of Houston. After giving the informed consent, subjects were asked to follow and complete a path marked on the ground while a robotic exoskeleton (REX, REX Bionics Ltd.) is controlled by an operator remotely. The robot motions in this study included walking forward, turning right, turning left and stop.

A 64 Channel electrode cap (actiCap system, Brain Products GmbH, Germany) was placed on the subject's head according to the international $10-20$ system having FCz as reference and AFz as ground. A wireless interface (MOVE system, Brain Products GmbH, Germany) was used to transmit data (sampled at 100Hz) to the host PC. Data were then filtered in the $(0.1-2$ Hz) range using a 2nd order Butterworth filter and standardized (z-score) in a preprocessing step. Figure 2.5 illustrates a standard closed-loop BMI system for lower-body exoskeleton control, in which the feature extraction and classification are important components for the entire decoding system.

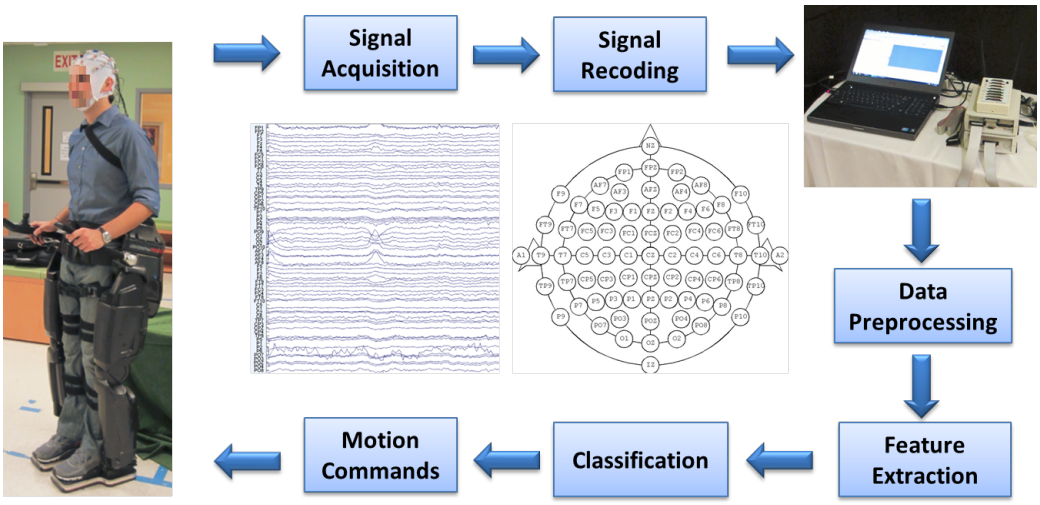


Figure 2.5: Illustrating a closed-loop BMI system being used to control a lower-limb exoskeleton.

2. BMI system for virtual reality application

Five healthy subjects with no history of neurological disease or gait pathology participated in this study for four sessions after each of them submitted a consent form. The experimental protocol was approved by the Institutional Review Board at the University of Houston, USA. At the beginning of each trial, the subject was instructed to stand still for 2 minutes on a treadmill while minimizing eye blinks. The treadmill was then slowly sped up to 1 mph by an experimenter and the subject kept this walking speed for 10 minutes.

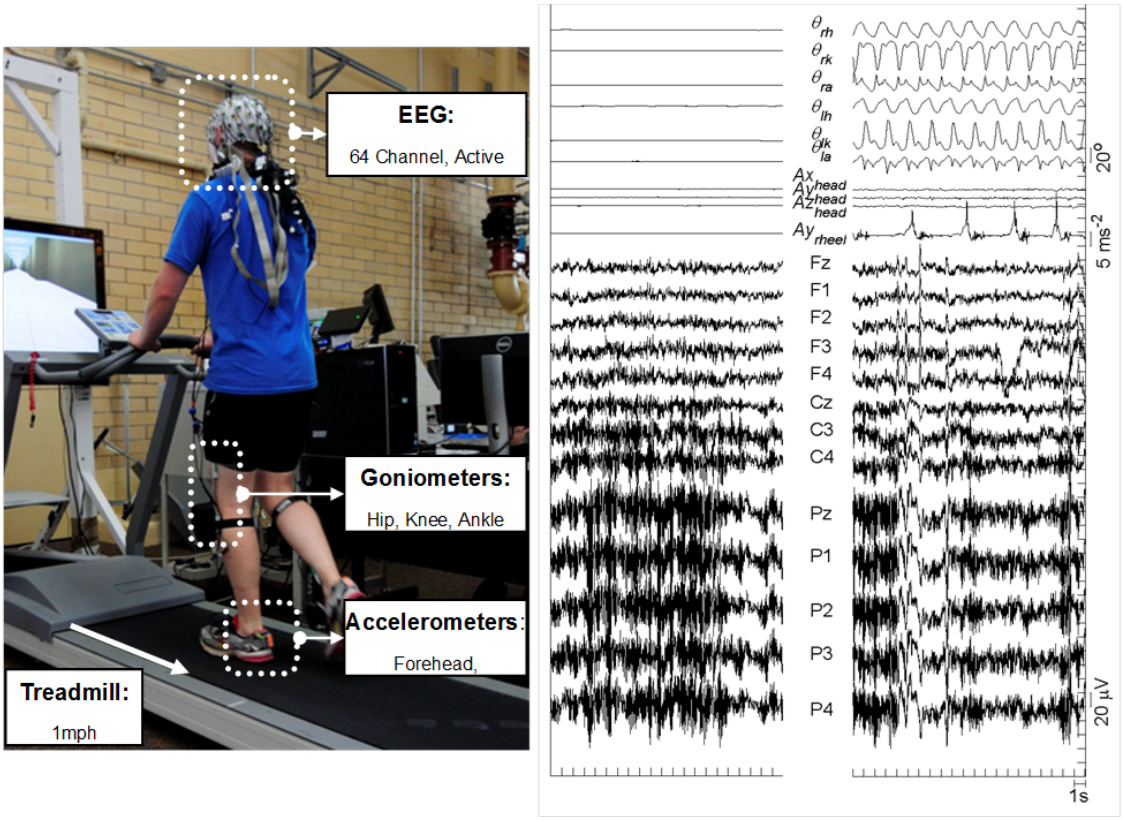


Figure 2.6: Experimental setup. Each subject was instructed to walk on a treadmill at 1 mile per hour (1 mph). EEG, lower limb joint angles, and accelerations of head, left and right heel were recorded.

Multichannel EEG (64 channels) was recorded by combining two 32-channel amplifiers (actiCap system, Brain Products GmbH, Germany). The electrodes were placed and labeled in accordance with the extended 10-20 international system. EEG data were referenced to FCz

channel and sampled at 100 Hz. Lower limb joint angles (hip, knee, and ankle) were recorded by goniometer sensors (SG150 & SG110/A Gonio, Biometrics Ltd, UK) at 100 Hz. Kinematic data (accelerations) were sampled at 128 Hz by using three wireless OPAL sensors (OPAL, APDM Inc., Portland, OR) placed at the forehead, left and right heel of the subject. Kinematic data of the heel would be used to segment all the data into gait cycles. Recording of EEG data, goniometer data, and OPAL data were synchronized using our custom C++ program. A raster plot (Figure 2.6) illustrates all the recorded data during standing and walking phase.

2.2.4 EEG Feature Extraction

The EEG is an electrical waveform that varies in time, and it contains frequency components that can be measured and analyzed. EEG features commonly used for decoding can be extracted in the time domain, the frequency domain, or the time-frequency domain.

2.2.4.1 Time-domain Feature

Signal amplitude modulations are the simplest and most commonly measured time-domain features for EEG data. To capture the amplitudes of a signal, a window of fixed length is often chosen and shifted in time — data captured within the window is then embedded into a feature space of the same dimensionality as the length of the window [25].

2.2.4.2 Frequency-Domain Feature

EEG signals can be analyzed in the frequency domain by giving a description of the signal energy as a function of frequency. Spectral estimation is a typical way to describe the frequency distribution of the power contained in a signal. The classical non-parametric approach is to estimate the power at carefully chosen frequency bands in a Fourier transform (e.g. DFT or FFT) generated spectra. The Fourier spectral features are computed with the Welch's method using windowed Fourier transforms of EEG signal segments.

2.2.4.3 Time-Frequency Feature

The EEG contains frequency components that can be measured and analyzed, and these frequency components have interesting and valuable properties. There are many ways to understand brainwaves. Clinicians view them for diagnostic purposes, seeking to identify patterns that associate with specific pathologies or conditions. Psychologists also study them in association with mental states, mental processing, and to test concepts of how the brain processes information. The basic EEG rhythms for clinical practice are summarized briefly in Table 2.3, with regard to their typical distribution on the scalp, mental states and spectral power changes during walking. Brain states may exist, and be correlated with the presence or absence of various frequencies, in time and space, rather than just one frequency. A direct approach to analyze signal at different frequencies is to decompose the signal in the time-frequency domain.

Table 2.3: EEG frequency bands in clinical practice

Band	Frequency	Location	Mental states	Spectral power during walking
Delta	0.1Hz - 3Hz	frontal regions in adults parietal in children	deep, dreamless sleep unconscious	no noticeable changes
Theta	4Hz - 7Hz	varies	deep relaxation, meditation, problem solving	no noticeable changes
Alpha	8Hz - 12Hz	occipital/parietal regions	relaxed, calm, meditation, creative visualisation	suppressed compared to standing increased during heel strike
Beta	12Hz - 30Hz	typlically frontal regions	awake, normal alert, consciousness	suppressed compared to standing increased during heel strike
Gamma	$>$ 32Hz	somatosensory cortex	thinking, integrated thought	increased compared to standing

A direct way to extend the Fourier transform from the frequency domain to the time-frequency domain is the short-time Fourier transform (STFT). However, there exists a limitation with STFT — the size of the window is fixed considering the fact that high-frequency signals typically require shorter time windows than low-frequency signals. A flexible approach in which the window size can vary across different frequencies is more desirable.

Discrete wavelet transform (DWT) based feature extraction has been found to be useful in EEG signal classification studies [49–51]. The basic idea of DWT is to represent a signal as a linear combination of a particular set of basis functions which can be obtained by shifting and dilating a mother wavelet. The computed wavelet coefficients provide a compact representation that shows the energy distribution of the signal in time and frequency domain.

In this study, we investigate a variant of DWT to extract time-frequency domain features. Redundant discrete wavelet transform (RDWT), also known as stationary wavelet transform, is designed to overcome the lack of translation-invariance of DWT [52]. Unlike DWT which has a compact representation, the implementation of RDWT removes the downsampling operator from the critically sampled DWT and instead upsamples the filter coefficients.

To implement the RDWT, let h and g be the scaling and wavelet filters in an orthonormal DWT, respectively, which at scale $j+1$ are defined recursively as

$$\begin{aligned} h_{j+1}[n] &= h_j[n] \uparrow 2, \\ g_{j+1}[n] &= g_j[n] \uparrow 2, \end{aligned} \tag{2.3}$$

where $h_0[n] = h[n]$, $g_0[n] = g[n]$ and $\uparrow$ denotes the upsampling operator.

The approximation and detail coefficients in the multiscale decomposition of a signal x can be derived via the recursive filter bank operations as

$$\begin{aligned} c_{j+1}[n] &= h_j[-n] * c_j[n], \\ d_{j+1}[n] &= g_j[-n] * c_j[n], \end{aligned} \tag{2.4}$$

where $c_0 = x$ and j is the scale from 0 to $J-1$, J is the ending scale.

An example of the implementation of RDWT to get multiscale representation of a signal x is shown in Figure 2.7. The example only shows when the signal is decomposed into three levels. After the decomposition, the length of the wavelet coefficients for each scale is the same as the original signal. In SRC, the wavelet coefficients in the same scale are used as basis to build the sub-dictionary, and the final dictionary is a combination of all sub-dictionaries.

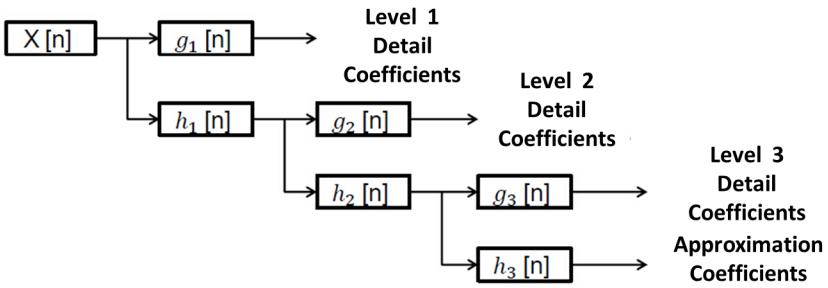


Figure 2.7: Block diagram of RDWT implementation.

2.3 Related Classification Algorithms

Generally, the most commonly used classification algorithms include Linear Discriminant Analysis (LDA), Nearest Neighbor (NN) classifiers, Bayesian classifiers, Hidden Markov Models (HMMs), Logistic Regression (LR), Sparse Representation Classifier (SRC), Artificial Neural Networks (ANNs), and kernel-based classifiers. A large proportion of classification algorithms are based on statistical models that operate by predicting the class for a new sample using statistical knowledge of a given set of training samples. These approaches represented by Bayesian classifiers are often easy to implement, however, are sensitive to the high dimensionality. In this section, we review two basic classifiers that are generally robust to high dimensional data and are directly related to this research.

2.3.1 Support Vector Machine

Support vector machine [53], one of the most popular kernel-based classifier, is originally designed as a linear classifier which discriminates classes by constructing a linear hyperplane similar to LDA. The decision function can be expressed as

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0. \tag{2.5}$$

The underlying principle of linear SVM is to simultaneously minimize the empirical classification error and maximize the geometric margin of the linear separation surface. The optimization problem for SVM classification is formulated as

$$\begin{aligned} &\min_{\mathbf{w},\xi_i,b} J(\mathbf{w},\xi_i,b) = \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \\ &\text{s.t.} \begin{cases} y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right) \geq 1 - \xi_i \\ \xi_i \geq 0, \forall i = 1, 2, \cdots, N \end{cases}, \end{aligned} \tag{2.6}$$

where C is a constant which controls the balance between the margin and empirical loss, ξ_i are slack variables which measure the degree of misclassification, and $\|\mathbf{w}\|^2$ is is inversely related to the margin to the hyperplane.

Although SVM achieves good performance in many linear applications, the reason for its popularity is that it can be easily extended as a nonlinear classifier for classes with nonlinear decision surfaces using the "kernel trick" [54]. Given an input data set $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ in the original data space $\mathbb{R}^d$, define a nonlinear mapping $\Phi(\cdot)$ from input space to a higher dimensional RKHS $\mathcal{H}$ as

$$\Phi : \mathbb{R}^d \to \mathcal{H}, \mathbf{x} \to \Phi(\mathbf{x}). \tag{2.7}$$

Then the so called "kernel trick" is used to involve a nonlinear kernel function in the input space. The mapping is defined as

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j)\rangle, \tag{2.8}$$

where $\langle \cdot, \cdot \rangle$ is the inner product of two vectors.

The key advantage of a SVM with a nonlinear kernel is that it can map the data which is not linearly separable in the input space to a high dimensional feature space, where the data are linearly separable, making the hyperplane decision surface of traditional SVM a reasonable choice.

2.3.2 Sparse Representation-based Classification

Sparse representation-based classification (SRC) is a recently developed classification method. In the sparse representation theory, it is assumed that the training samples from the same class lie on a low-dimensional subspace, and a new test sample will approximately lie in the linear subspace spanned by the training samples in the associated class. Mathematically, a test sample $\mathbf{x} \in \mathbb{R}^d$ can be represented as a sparse linear combination of all training samples as

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n}, \tag{2.9}$$

where $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_C] \in \mathbb{R}^{d \times n}$ is a dictionary representing C classes, $\mathbf{s} \in \mathbb{R}^n$ is the sparse coefficient vector, and $\mathbf{n} \in \mathbb{R}^d$ represents the noise vector.

In order to obtain the sparse solution $\mathbf{s}$, it is natural to model a ℓ_0-norm minimization algorithm, however, it is an NP-hard problem because of the non-differentiability and non-convex nature of the ℓ_0-norm. An alternative approach is to solve a ℓ_1-regularized convex programming problem also known as LASSO, which can be expressed as

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}} \|\mathbf{x} - \mathbf{A}\mathbf{s}\|_2^2 + \lambda \|\mathbf{s}\|_1, \tag{2.10}$$

where λ is a positive regularization parameter.

After the sparse coefficient $\hat{\mathbf{s}}$ is obtained, the class label of $\mathbf{x}$ is determined by the minimum residual between $\mathbf{x}$ and its approximation from each class-wise sub-dictionary as

$$\text{class}(\mathbf{x}) = \underset{c=1,2,\ldots,C}{\arg\min} \|\mathbf{x} - \mathbf{A}_c \hat{\mathbf{s}}_c\|_2, \tag{2.11}$$

where $\hat{\mathbf{s}}_c$ is the subset of the sparse coefficient vector $\hat{\mathbf{s}}$ associated with class c, and $\mathbf{A}_c$ is the corresponding sub-dictionary.

2.4 Multi-source Data Learning

Multi-source data learning is sometimes referred to as multi-view/multi-task learning in the related fields such as computer vision, in which views are obtained from multiple sources or multiple feature subsets. A simple and conventional way to apply machine learning algorithms on multi-source data is to concatenate multiple sources/features into a "single source" and adapt to the learning setting. Although this approach is straightforward, the concatenation usually results in a very high dimensional input space and may cause overfitting in the case of a small size of training set. Multi-view learning algorithms can be classified into three categories as: (1) co-training, (2) multiple kernel learning, and (3) subspace learning [55].

Co-training is one of the earliest methods for multi-view learning, which alternately trains on distinct views to maximize the mutual agreement. Different variants of co-training include generalized expectation-maximization [56], a combination with active learning [57], employment of Bayesian graphical model [58], and application of co-regularization [59]. Generally, co-training algorithms should satisfy three assumptions: (1) conditional independence — given the class, views should be conditionally independent; (2) sufficiency — each view should be sufficient for classification on its own; (c) compatibility — the prediction from different views should have a high probability for a given class.

Multiple kernel learning (MKL) is another approach to learn multi-source data for classification, which can be viewed as a generalization of SVM. MKL is based on a combination

(typically linear) of different base kernels, where each kernel is dedicated to a particular type of feature (e.g., a unique source). The goal of MKL is to simultaneously learn a kernel and associated prediction in a supervised learning setting. MKL was first formulated as a semi-definite programming problem by Lanckriet et al. [60], and was then developed by Bach et al. as an SMO algorithm to solve the medium-scale problem [61]. Further, Sonnenburg et al. developed an efficient semi-infinite linear program (SILP) and made MKL applicable to large scale problems [62]. In 2008, Rakotomamonjy et al. [63] proposed an efficient algorithm, named simpleMKL, by exploring an adaptive ℓ_2-norm regularization formulation. In addition, some research [64], [65] found the consistency and established a connection between MKL and group-LASSO for dealing with the group structure of data.

The aim of subspace learning on multi-source data is to obtain a latent subspace shared by multiple features. This approach is directly related to dimensionality reduction (e.g. via manifold learning), which aims to explore the lower dimensional intrinsic space of data. For single source data, the simplest and most widely used technique to exploit the subspace is principal component analysis (PCA). For multi-source data learning, the corresponding method is the canonical correlation analysis (CCA) [66]. CCA aims to maximize the correlation between different sources in the subspace and outputs the optimal projection for each source. Similar to PCA, CCA exploits the subspace in an unsupervised way without utilizing the label information. In order to make the use of class information, a generalized Fisher's discriminant analysis [67] was proposed to explore the latent subspace spanned by multi-source data. Through subspace learning, data from different sources result in a latent low dimensional subspace, and this avoids the overfitting problem, alleviating the "Hughes phenomenon."

The above multi-source data learning algorithms can be considered as feature fusion approaches. As stated in the first chapter, decision fusion is also widely applied for multi-source

data analysis. Ensemble learning is a representative decision fusion based learning framework for multi-source data classification, which constructs a set of classifiers and makes decisions based on their individual predictions in some way.

Bagging and boosting are two classical ensemble learning algorithms. The bagging algorithm constructs a ensemble of classifiers from different datasets and makes predictions through uniform averaging or weighted voting over predicted labels [68]. Theoretical results show that the expected error of bagging has the same bias component as a single bootstrap replicate, while the variance component is reduced. Boosting is structurally similar to bagging, except that it adaptively trains a new model to compensate for the errors made by earlier models through the learning process [69].

In this research, we investigate the feature and decision fusion methods from various aspects and propose several algorithms extending the basic methods in the above categories.

2.5 Active Learning

Active learning (AL) is a machine learning approach that allows users to interact with the classifier, both providing capability to select the most informative samples and allowing adaptation in dynamic environments. The goal of AL is to minimize the cost related to the sample labeling process while maximizing the discrimination capabilities.

In recent years, many AL strategies have been proposed for remote sensing data classification. One group is specific to margin-based classification approaches, such as SVM classifiers. In this context, margin sampling (MS) represents a simple but powerful strategy [70], [71], where the importance of samples is based on the distance to the hyperplane, which indicates the level of uncertainty and its importance towards learning the decision boundary. Samples whose distance to the hyperplane is small are likely to be support vectors, and thus more important for learning the classifier. In [72], instead of using the distance to the hyperplane as

selection measure, the original classification problem is reformulated into a new binary problem where the goal is to discriminate between significant and nonsignificant samples. While exploiting the kernel space induced by spectral features has been demonstrated as a successful framework for MS, an enhanced kernel space can be constructed by including multiple features or sources within a multiple kernel learning (MKL) framework, as demonstrated in our preliminary work reported in [73]. Although MS is effective in many AL problems, limitations include — (1) It can only be applied to margin-based methods, in which decisions are made on the distance to a separating hyperplane; and (2) It is not suitable when multiple classifiers are involved in the learning process, since no inter-classifier-information among the samples is considered.

Another popular family of AL strategies quantifies the uncertainty of samples by considering a committee of learners [74]. Each member of the committee builds its own learning model, and consequently labels the samples in the candidate pool. The algorithm then selects the samples which have the maximum disagreement for the different classification models in the committee. Among the strategies that can be utilized to construct committees, a recently proposed approach for hyperspectral imagery utilizes feature subsets as a proxy for multiple views (each view being a member in this ensemble/committee) [75].

Other methods are based on posterior probabilities. In [76], using a maximum-likelihood classifier, the samples whose inclusion in the training set maximizes changes in the posterior distribution are selected. In [77], samples are selected as a function of entropy of the corresponding class label. Another strategy is represented by the breaking-ties criterion [78], in which the difference between the two largest posterior probabilities is considered. More recently, researchers have incorporated spatial information [79], [80], and have incorporated the AL framework in practical operational scenarios [81], [82]. Moreover, while the number of

samples is an appealing way to formulate the labeling cost, the time spent for labeling or the distance traveled in the field can be more appropriate for certain applications [83].

Although several AL strategies have been proposed in the literature, they have been applied mostly for single-sensor remote sensing data. Little research has been conducted in multi-sensor scenarios. In this research, one of our goals is to develop a robust multi-source AL framework that can be applied to select important samples for labeling across multiple inputs.

Chapter 3

Multi-source Data Fusion via Optimization of Mixture of kernels

3.1 Introduction

In most kernel-based learning methods, performance is greatly affected by the choice of kernel function and related kernel hyper-parameters. The standard SVM only utilizes a single kernel function with fixed parameters, which necessitates model selection for good classification performance. Besides, using a fixed kernel may introduce bias, since different sources of data may have different representations of the phenomena of interest, and hence the similarity should not be measured via the same kernel function. For such situations, mixture-of-kernels methods have been recently rising as an efficient approach to learn multi-source data for classification. This type of approaches represented by the multiple kernel learning (composite kernel learning) is based on a combination (typically linear) of different base kernels, where each kernel is dedicated to a particular type of feature (e.g., a unique source).

Based on the concept of composite kernel, we propose a joint feature extraction method to obtain useful information from multi-source data in a much lower dimensional space. In this framework, features from different sources are first fused via a weighted composite kernel mapping, and then projected to a lower dimensional subspace in which a kernel local Fisher discriminant analysis (KLFDA) is used to extract the most discriminative information. We

hypothesize that after such a projection, multi-source data would have better class separability between classes, and an efficient linear classification model, such as multinomial logistic regression (MLR), would be suitable for classification.

In the composite kernel based approach, since it usually involves two sources, the weights are typically optimized by grid search and cross-validation. Even though this approach is simple and easy to implement, the exhaustive search over the entire parameter space may not be optimal when the number of sources are more than two. We need a more sophisticated method to optimize the weights. Multiple kernel learning (MKL) have been shown to outperform traditional single-kernel machines in different applications [18, 62, 84–88]. The advantage of using MKL over SVM is that MKL can simultaneously learn the classifier and the optimal weights for base kernels. To investigate this property, we conduct a research making use of the optimized weights to simultaneously decode different patterns from the EEG signals and learn the relative importance of different scalp brain areas.

In addition, to solve the problem of insufficient labeled samples for classification in real application, this chapter also presents a robust multiple kernel AL framework that can be applied in a multi-source environment to select important samples for labeling. In particular, we propose a novel ensemble multiple kernel active learning (EnsembleMKL-AL) system based on the maximum disagreement query strategy that incorporates different types of features and fuses them for robust classification.

The remainder of this chapter is organized as follows. Section 3.2 introduces the CKLFDA as a feature extraction method and validates its performance using two sets of multi-source data for geospatial classification. In section 3.3, the MKL algorithm is applied to simultaneously decode the pattern of user's internal states from the EEG signals and learn the relative importance of different scalp brain areas. In Section 3.4, the EnsembleMKL-AL framework based on

the maximum disagreement query strategy is presented and compared with SimpleMKL-AL to investigate the benefit of using such framework for multi-source data classification. At the end of this chapter, Section 3.5 summarizes the results and contribution.

3.2 Composite Kernel Local Fisher Discriminant Analysis

3.2.1 Proposed Framework

In this section, we present a composite kernel based feature extraction method that is efficient to jointly extract features from multi-source data for classification. The work is based on the KLFDA algorithm, with a composite kernel replacing the single kernel in KLFDA. We hypothesize that composite kernels would facilitate an "optimized" feature space for multi-source data analysis. The proposed CKLFDA algorithm for multi-source data fusion is described in detail below.

Given a training data set $D = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$, where n is the number of samples, $\mathbf{x}_i \in \mathbb{R}^p$ is the input vector with p features and $y_i \in \{1, 2, ..., q\}$ is the class index of sample i. Let n_l be the number of training samples in the lth class, and $\sum_{l=1}^{q} n_l = n$. The affinity between $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as

$$A_{i,j} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\gamma_i \gamma_j}\right), \tag{3.1}$$

where $\gamma_i = \left\|\mathbf{x}_i - \mathbf{x}_i^{(k_{nn})}\right\|$ is the local scaling of samples in the neighborhood of $\mathbf{x}_i$, and $\mathbf{x}_i^{(k_{nn})}$ is the k_{nn}-nearest neighbor of $\mathbf{x}_i$. $A_{i,j} \in [0, 1]$ measures the distance among samples.

With the data projected in RKHS, the local between-class scatter matrix S_{lb} and within-class scatter matrix S_{lw} in KLFDA are defined as

$$\mathrm{S}_{lb} = \frac{1}{2}\sum_{i,j=1}^{n} W_{i,j}^{(lb)}(\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))(\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))^T \text{ and} \tag{3.2}$$

$$\mathrm{S}_{lw} = \frac{1}{2}\sum_{i,j=1}^{n} W_{i,j}^{(lw)}(\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))(\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))^T, \tag{3.3}$$

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

$$W_{i,j}{}^{(lb)} = \begin{cases} A_{i,j}(1/n - 1/n_l), & \text{if } y_i = y_j = l \\ 1/n, & \text{if } y_i \neq y_j \end{cases} \quad \text{and} \tag{3.4}$$

$$W_{i,j}{}^{(lw)} = \begin{cases} A_{i,j}/n_l, & \text{if } y_i = y_j = l \\ 0, & \text{if } y_i \neq y_j. \end{cases} \tag{3.5}$$

The transformation matrix T_{KLFDA} is then given by maximizing the local Fisher's ratio $(T^T S_{lw} T)^{-1} T^T S_{lb} T$ as

$$T_{KLFDA} = \arg\max_T \mathrm{tr}[(T^T S_{lw} T)^{-1} T^T S_{lb} T], \tag{3.6}$$

where any solution T can be expressed as linear combinations of $\Phi(\mathbf{x}_j)$ with coefficients α_j

$$T = \sum_{j=1}^{n} \alpha_j \Phi(\mathbf{x}_j). \tag{3.7}$$

Since T is in the RKHS and cannot be directly computed, *the kernel trick* [54] can be applied as

$$T^T \Phi(\mathbf{x}_i) = \sum_{j=1}^{n} \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) = \alpha^T \mathbf{K}. \tag{3.8}$$

Following this, Eq. (8) can be rewritten using coefficient α as

$$\alpha_{opt} = \arg\max_\alpha \mathrm{tr}[(\alpha^T S_{lw} \alpha)^{-1} \alpha^T S_{lb} \alpha]. \tag{3.9}$$

Finally, samples in the lower-dimensional feature space can be represented as

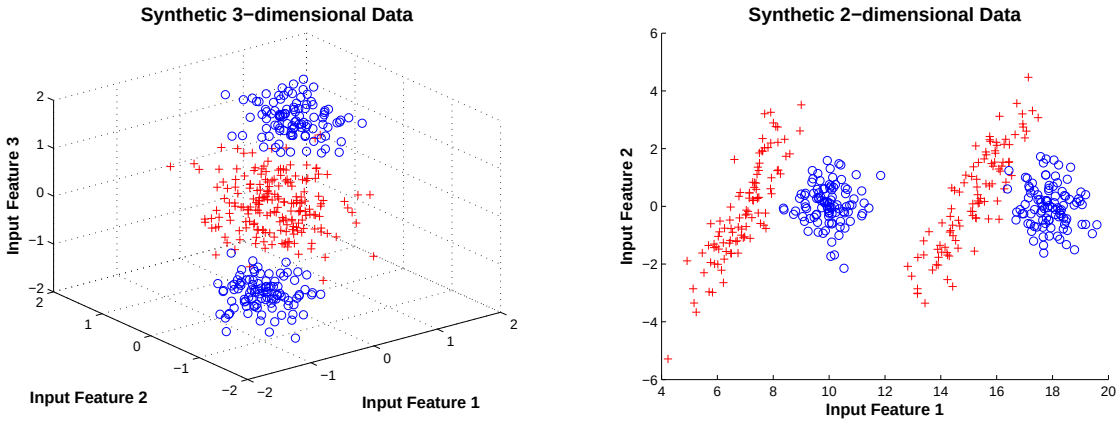
$$\mathbf{z} = \alpha_{opt} \mathbf{K}, \quad where \quad \mathbf{z} \in \mathbb{R}^r, r \ll p. \tag{3.10}$$

With this notion of composite kernels, we extend KLFDA to CKLFDA. For multi-source data, a weighted summation kernel is employed to balance different data sources (for example the hyperspectral and LiDAR data), as

$$K(\mathbf{x}_i, \mathbf{x}_j) = dK_{hy}(\mathbf{x}_i^h, \mathbf{x}_j^h) + (1 - d) K_{li}(\mathbf{x}_i^l, \mathbf{x}_j^l), \tag{3.11}$$

where K_{hy} and K_{li} are hyperspectral and LiDAR basis kernels, respectively; d is the weight of hyperspectral kernel — varying d provides different composite kernels. The CKLFDA algorithm can be derived by replacing the single kernel in KLFDA by a composite kernel. We hypothesize that such a composite-kernel extension of KLFDA (which has been shown previously to outperform KDA) can yield feature spaces that best represent multi-sensor datasets in a reduced dimensional subspace.

In order to highlight the benefit of CKLFDA for multi-source feature extraction and its separability between different classes after data projection, we present an example of dimensionality reduction using synthetic *multi-source* two-class multimodal data. The two-class synthetic multimodal data are shown in Figure 3.2.1, where synthetic data from each source forms a different input space (source 1 in this example is in a 3-dimensional space, while source 2 is in a 2-dimensional space) with different distributions. Following this, different feature extraction methods, including LDA, LFDA, KLFDA, and CKLFDA, are applied to project multi-source data onto a 1-dimensional subspace, and the histograms of the data in the projected subspaces are shown in Figure 3.2.



(a) (b)

Figure 3.1: Synthetic 3-dimensional and 2-dimensional multimodal data

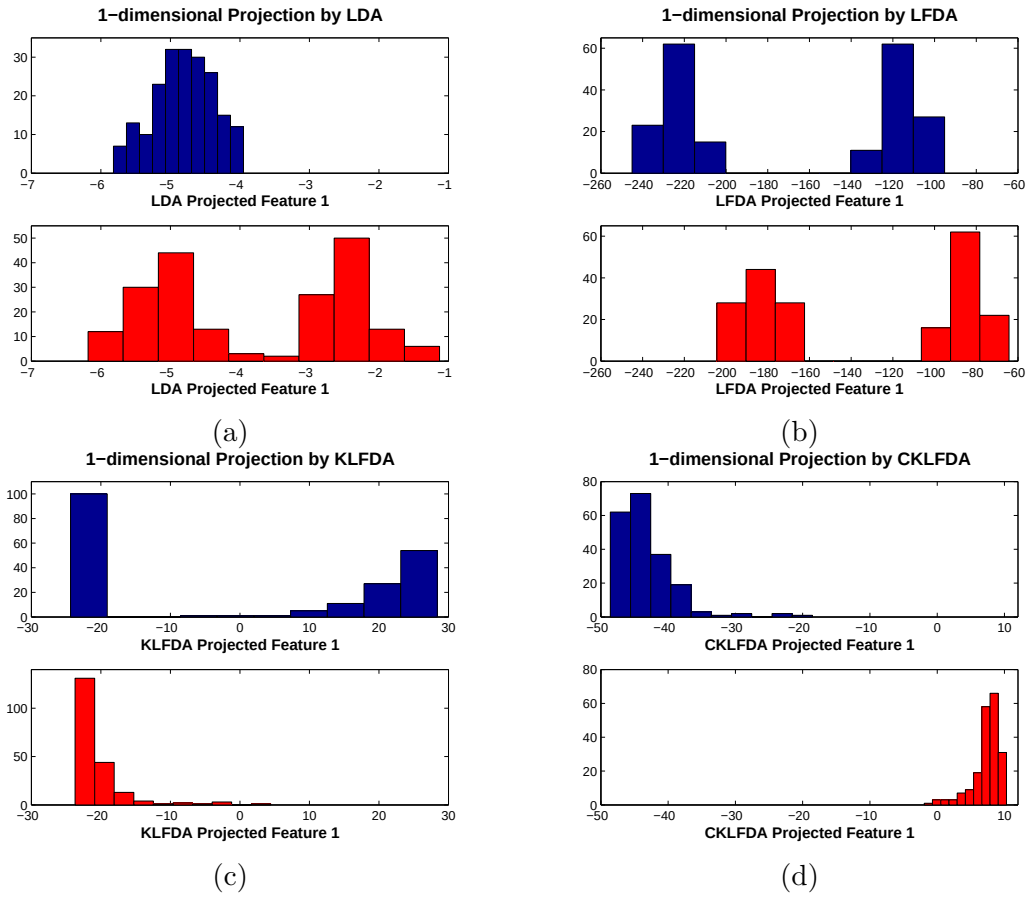


Figure 3.2: Histograms of the synthetic data when projected onto a 1-dimensional subspace using (a) LDA; (b) LFDA; (c) KLFDA; (d) CKLFDA.

It is observed that the distributions of two-class data completely overlap with each other in the LDA subspace, while for LFDA and KLFDA, data is less overlapped, because LFDA and KLFDA can preserve the multimodal structure of the data in the projected subspace. Further, by building a composite kernel for different sources, two-class data are well separated in the subspace, which demonstrates that data have better separability after CKLFDA projection, and are indeed linearly separable as would be expected owing to the nonlinear kernel projection.

Motivated by the above observations, we contend that in the resulting feature space, a simple linear classifier would suffice for classification. Previous work with MKLFDA utilized a nearest neighbor (NN) classifier [89], which is highly local and does not utilize intrinsic statistical information about the data. Our results above demonstrate that a traditional MLR

(or even a quadratic Gaussian maximum likelihood classifier) would be appropriate after a CKLFDA projection, even in multi-source remote sensing settings.

The logistic regression (LR) model is typically used for prediction of occurrence of an event by fitting data to a logistic curve. The standard LR is a linear, supervised classifier used for binary classification. However, in many real world applications, we are dealing with multi-class problems. The MLR is an extension of binomial logistic regression and has been found to work well for multi-class classification. For a given training data set $D = \{\mathbf{x}_i, y_i\}_{i=1}^{n}$, the probability that a given training sample $\mathbf{x}_i$ belongs to class m is given by the MLR model as

$$p(y_i = m|\mathbf{w}_i) = \frac{\exp(\mathbf{w}_i^m)}{\sum_{m=1}^{q} \exp(\mathbf{w}_i^m)}, \tag{3.12}$$

where $\mathbf{w}_i^m = \mathbf{v}^m \mathbf{x}_i$. The parameters (regressors) of the logistic regression model $\mathbf{v} = [\mathbf{v}^1, \mathbf{v}^2, ..., \mathbf{v}^m]$ can be obtained by calculating a maximum a posteriori (MAP) estimate as

$$\mathbf{v}_{MAP} = \arg\max_{\mathbf{v}} [l(\mathbf{v}) + \log p(\mathbf{v})], \tag{3.13}$$

where $p(\mathbf{v})$ is a Laplacian prior on $\mathbf{v}$ which is independent from the observation $\mathbf{x}$. In order to control the sparsity of $\mathbf{v}$, a regularization parameter λ is defined, and $\mathbf{v}$ is modeled as a random vector with Laplacian density $p(\mathbf{v}) \propto \exp(-\lambda \|\mathbf{v}\|_1)$. $p(\mathbf{v})$ forces many components of to be zero, and thus controls the complexity of the MLR classifier.

In (3.13), $l(\mathbf{v})$ is the log-likelihood function given by

$$\begin{aligned} l(\mathbf{v}) &= \log \prod_{i=1}^{n} p(y_i|\mathbf{x}_i, \mathbf{v}) \\ &= \sum_{i=1}^{n} \left((\mathbf{v}^{y_i})^T \mathbf{x}_i - \log \sum_{m=1}^{q} \exp\left((\mathbf{v}^m)^T \mathbf{x}_i \right) \right). \end{aligned} \tag{3.14}$$

The optimization problem in (3.13) can be solved by sparse MLR (SMLR) method proposed in [90].

3.2.2 Experimental Settings and Results

3.2.2.1 Experiment Setting

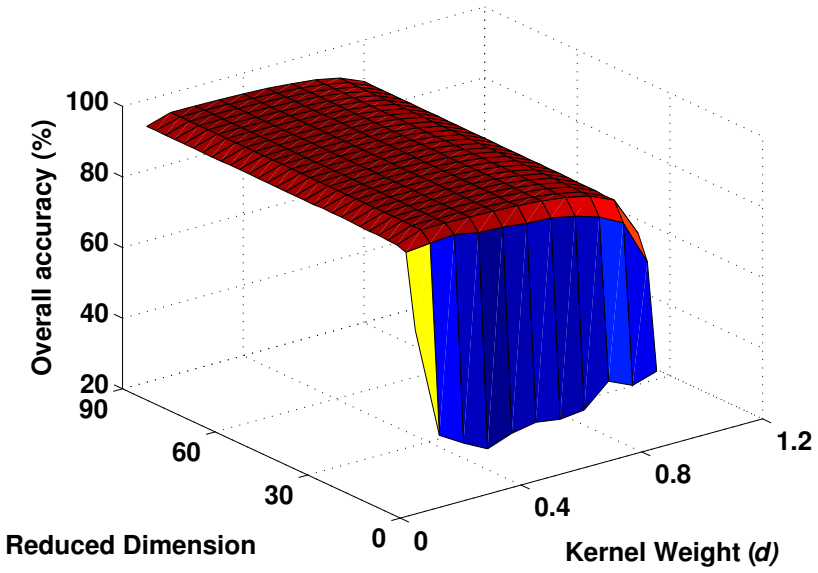
The efficacy of the proposed method is demonstrated via experiments using two different sets of multi-source geospatial data. For feature fusion, the raw spectral data and EMAPs derived from the hyperspectral image are used as a testbed for multi-source image analysis. The second multi-source testbed used for validation involves sensor fusion, in which the hyperspectral and LiDAR data are utilized. In the experiments, composite kernels for training and testing were computed using hyperspectral and LiDAR (or spectral and spatial) *basis kernels* with different weights. We randomly chose 30, 50, 80 samples from each class to build the training kernels and then used them to obtain the CKLFDA transformation matrix. In this case, the testing kernels were projected to the feature space via the same CKLFDA transformation matrix.

Kernel alignment is a reasonable approach to measure the similarity between a candidate kernel and the "ideal" kernel built by class labels [91], and can be used to choose appropriate kernel parameters. Given a kernel matrix $\mathbf{K}_\sigma$, and a vector of labels $\mathbf{y} = [y_1, y_2, ..., y_n]$, the alignment between two kernels can be defined as

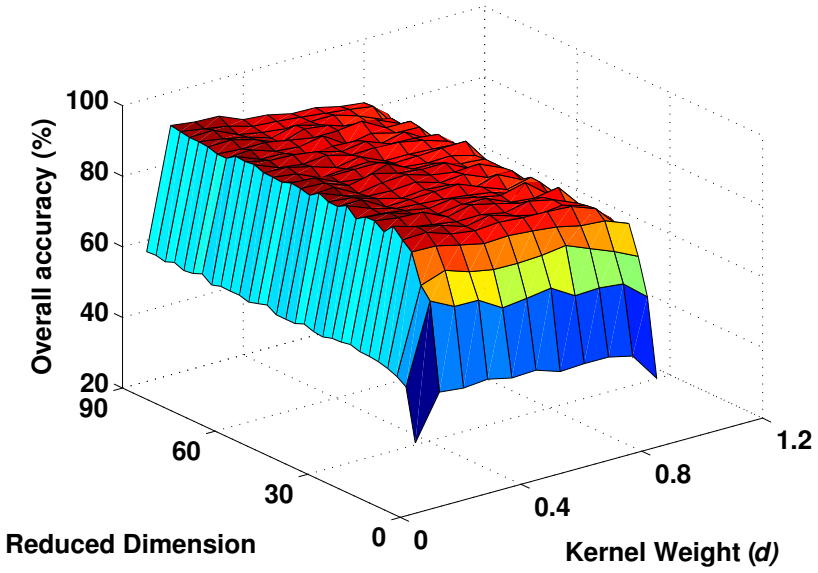
$$A(\mathbf{K}_{ideal}, \mathbf{K}_\sigma) = \frac{\langle \mathbf{K}_\sigma, \mathbf{K}_{ideal} \rangle_F}{\sqrt{\langle \mathbf{K}_\sigma, \mathbf{K}_\sigma \rangle_F \langle \mathbf{K}_{ideal}, \mathbf{K}_{ideal} \rangle_F}}, \tag{3.15}$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius distance between matrices defined as $\langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F = \sum_{i,j} K_1(\mathbf{x}_i, \mathbf{x}_j) K_2(\mathbf{x}_i, \mathbf{x}_j)$. The ideal kernel $\mathbf{K}_{ideal}$ can be computed by inner product of labels for a binary classifier, as $\mathbf{K}_{ideal} = \mathbf{y}\mathbf{y}^T$. For multiclass classification, the ideal kernel is computed as

$$K(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 0, & y_i \neq y_j \\ 1, & y_i = y_j. \end{cases} \tag{3.16}$$



(a)



(b)

Figure 3.3: Overall accuracy versus reduced dimension, and kernel weight d in CKLFDA-MLR method for (a) University of Pavia dataset and (b) UH dataset.

In this research, all basis kernels are RBF kernels with the optimal parameter σ chosen by maximizing the alignment of each candidate kernel with the output vector. For spectral-spatial classification, the parameters were $\sigma_{spec} = 0.5, \sigma_{spat} = 1$, and for multi-sensor classification

$\sigma_{hy} = \sigma_{li} = 0.5$.

We compare the proposed CKLFDA-MLR method with several state-of-the-art feature extraction and classification algorithms. The Gaussian maximum likelihood (ML) classifier, a popular linear classification method in remote sensing applications, is chosen for comparison. We then compare the results of MLR and ML classifier combined with different feature extraction methods, including PCA, LDA, KPCA, KLFDA. Since the first two methods (PCA and LDA) are not kernel-based methods, the multi-source features are stacked together in the original data space prior to feature reduction. KPCA and KLFDA are standard kernel-based feature extraction algorithms with stacked features, utilizing a single kernel.

In CKLFDA experiments, we varied the weight d for each basis kernel, such that we can determine a proper balance between the kernels. The other parameters that need to be tuned in CKLFDA are the reduced dimension and the value of k_{nn}. All parameters d, *reduced dimension* (r), k_{nn} were tuned using a grid search. Based on our experimental observations, both ML and MLR classifiers are not sensitive to the k_{nn} value if it is in a reasonable range, e.g. 5 to 41 in this experiment. Figure 3.3 depicts the overall accuracy of CKLFDA-MLR as a function of reduced dimension and kernel weight d. The kernel weight d was tuned in the range [0,1] with a step size of 0.1. From Figure 3.3 we observe that the classification results are not sensitive to the parameter d when a composite kernel is used (when $d \in [0.1, 0.9]$). The OA is much lower when only a single data source is used (when $d = 0$ or $d = 1$), in which case the composite kernel becomes a traditional single kernel. This observation justifies our hypothesis that the mixture of kernel built by multi-source data would help to enhance the classification of hyperspectral images. We also note that dimensionality has little influence on the overall accuracy of MLR classifier if it is larger than 10 for University of

Pavia dataset and 15 for UH dataset, indicating that CKLFDA is a very effective dimensionality reduction approach. However, for the ML classifier, the reduced dimensionality should be smaller than the number of training samples, otherwise the estimated scatter matrix is badly conditioned and the data are poorly classified. Based on the experimental results, the highest accuracy was achieved when the dimensionality is reduced to 20 for the ML classifier.

3.2.2.2 Spectral and Spatial Classification Results

Table 3.1 lists average overall accuracies and standard deviation for different methods from 10 repeated randomly subsampled experimental runs.

Table 3.1: Overall accuracies (OA) and standard deviation (%) of Pavia dataset

Method	Number of training samples for each class		
	30	50	80
LDA-ML	83.63 (2.62)	85.87 (2.57)	88.24 (2.02)
PCA-ML	84.55 (2.37)	85.75 (3.34)	87.53 (1.40)
KPCA-ML	84.44 (4.18)	87.19 (1.65)	88.51 (2.21)
KLFDA-ML	86.02 (3.51)	89.55 (1.99)	91.93 (1.47)
CKLFDA-ML	88.67 (2.42)	91.83 (1.55)	94.62 (1.23)
LDA-MLR	77.87 (3.80)	93.25 (1.73)	93.87 (1.60)
PCA-MLR	70.06 (3.44)	74.34 (3.36)	75.43 (2.82)
KPCA-MLR	81.58 (1.85)	84.07 (3.37)	84.55 (2.78)
KLFDA-MLR	92.93 (1.43)	95.50 (1.31)	96.65 (1.05)
CKLFDA-MLR	**93.24 (1.74)**	**96.73 (0.95)**	**97.17 (0.92)**

From the above results, the kernel feature extraction methods achieve higher classification accuracies than standard PCA and LDA methods. Moreover, KLFDA has better performance than KPCA. For example, when 80 training samples are used, the OA for KLFDA-ML is 91.93% compared to 88.51% for KPCA-ML, while for KLFDA-MLR the OA is 96.65% compared to 84.55% for KPCA-MLR. Compared with the results of KLFDA using a single kernel, CKLFDA with a composite kernel achieves better class separability. The OA of CKLFDA for both ML and MLR classifiers are higher than those of KLFDA, and the increase of the accuracy can be as much as 2.69%. In general, the CKLFDA-MLR method outperforms all the other methods

in terms of the OA. The lower standard deviation implies that CKLFDA-MLR is not only an efficient but also a robust algorithm.

3.2.2.3 Hyperspectral and LiDAR Classification Results

The overall accuracies and standard deviation for the UH multi-source data are shown in Table 3.2. By observing the results of this multi-sensor dataset, we can derive similar conclusions to those of spatial-spectral classification. The standard PCA-MLR method has the lowest OA, while the kernel-based approaches improve the classification performance by fusing different data effectively through kernel mapping. CKLFDA performs the best among all methods with the OA of 94.81% for MLR classifier and 93.59% for ML classifier with 80 samples per class.

Table 3.2: Overall accuracies (OA) and standard deviation (%) of UH dataset

Method	Number of training samples for each class		
	30	50	80
LDA-ML	75.85 (1.53)	85.26 (1.18)	89.36 (1.10)
PCA-ML	76.48 (2.03)	86.26 (1.19)	89.39 (1.10)
KPCA-ML	77.32 (1.47)	85.69 (1.14)	89.43 (1.05)
KLFDA-ML	81.53 (1.36)	86.74 (1.13)	89.82 (0.88)
CKLFDA-ML	86.06 (1.33)	90.35 (0.85)	93.59 (0.73)
LDA-MLR	76.87 (1.48)	79.39 (1.14)	84.44 (1.07)
PCA-MLR	62.36 (4.51)	63.59 (3.89)	64.41 (3.43)
KPCA-MLR	68.45 (1.03)	70.22 (0.93)	73.53 (0.88)
KLFDA-MLR	88.16 (1.23)	90.11 (0.56)	92.77 (0.48)
CKLFDA-MLR	**91.10 (0.91)**	**92.22 (0.78)**	**94.81 (0.61)**

3.2.2.4 Computational Cost

All the experiments were implemented in Matlab R2012a on a Linux system with twelve 3.2GHz Intel(R) cores and 32GB RAM. We compare the feature extraction and classification time of different methods using the University of Pavia dataset with 80 training samples per class. Including the time required for parameter selection, the mean processing time for learning the CKLFDA projection was 2.71s compared to 1.26s for KLFDA and 1.89s for KPCA. The

baseline linear methods, LDA and PCA, have relatively shorter computational time (0.18s and 0.08s, respectively).

3.3 Multiple Kernel Based Region Importance Learning

3.3.1 Proposed Framework

In BMI applications, the main goal is decoding EEG signals to predict and translate user's intention to the external device. Most machine learning methods serve as a "black box" in that we do not know what happens in the human brain and how the brain regions contribute to the decoding while people perform different tasks. The human brain consists of over a 100 billion cells, typically divided into regions by neuroanatomists. Different regions may have their specific functionalities while coordinating together to achieve normal operations. Therefore, at any time and particular location, the brain may have different functionalities, and variations in time, and in space (observed as different places on the scalp) are important to understand.

In that context, the assumption of this research is that different brain regions contribute differently to control lower-limb movements and we are interested in learning such information. MKL is based on the automatic optimization of a linear combination of multiple kernels, in which each basis kernel can be represented by a group of electrodes corresponding to regions of interest (ROIs), and consequently contribute unique biophysical information. In this work, the goal is to decode the pattern of user's internal gait states (e.g., stop, walk, turn left, turn right) from the EEG signals and simultaneously learn the relative importance of different brain regions.

Generally, the brain consists of identifiable areas, i.e., frontal (motor and sensory cortex), parietal, occipital (visual), temporal (hearing, language). Different brain functions are thought to be associated with the particular involved area. A summary of brain regions and their

associated functions in a normal or injured brain is shown in Table 3.3. We investigate the importance of these brain areas in the lower-limb movement decoding task. Specifically, the scalp is further divided into 13 topographical regions of interest (ROIs) adapted from the previous definition in [92], [93], which are anterior frontal (AF), left frontal-central (LFC), midline frontal (MF), right frontal-central (RFC), left centro-parietal (LCP), midline central (MC), right centro-parietal (RCP), left parietal (LP), middle parietal (MP) right parietal (RP), Left Temporal (LT), Right Temporal (RF) and Occipital (O). Figure 3.4 and Table 3.4 show the partition of the scalp and the names for each ROI.

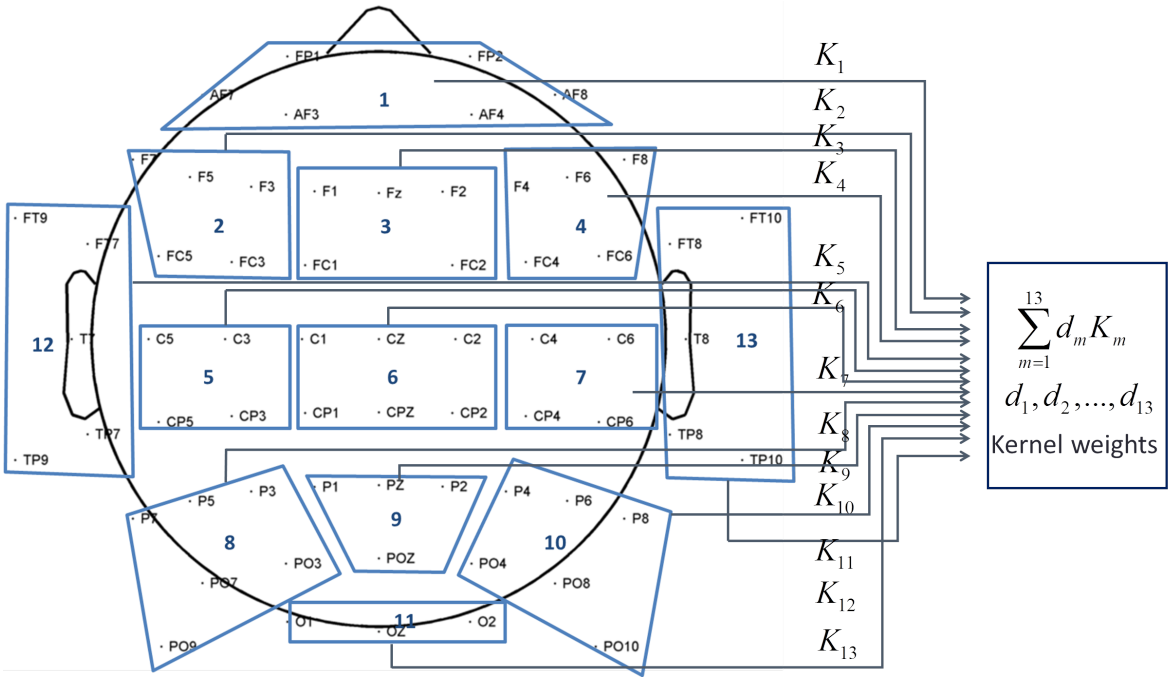


Figure 3.4: Scalp regions of interest (ROIs).

Table 3.3: Brain regions, normal brain functions and problems with brain injury [94–96]

Brain region	Function	Injured brain
Frontal lobe	Personality / emotions	Changes in behavior and personality
	Intelligence	Mood swings, irritablity
	Attention / concentration	Unable to focus on a task
	Judgement	Repetition of a single thought
	Body movement	Loss of movement (paralysis)
	Problem Solving	Difficulty with problem solving
	Speech (speak & write)	Difficulty with language
Temporal lobe	Speech (understanding language)	Difficulty understanding language
	Memory	Problems with memory
	Hearing	Difficulty identifying objects
	Sequencing	Difficulty recoginizing faces
	Organization	Increased aggressive behavior
Parietal lobe	Sense of touch, pain and temperature	Lack of awareness of body parts
	Distinguishing size, shape and color	Difficulty with eye-hand coordination
	Spatial perception	Difficulty distinguishing left from right
	Visual perception	Problems with reading, writing, naming
Occipital lobe	Vision	Defects in vision or blind spots
		Blurred vision
		Visual illusions
		Difficulty reading and writing

Table 3.4: Scalp ROI names

Index	ROI Name	Index	ROI Name
1	Anterior Frontal (AF)	8	Left Parietal (LP)
2	Left Fronto-Central (LFC)	9	Middle Parietal (MP)
3	Midline Frontal (MF)	10	Right Parietal (RP)
4	Right Fronto-Central (RFC)	11	Left Temporal (LT)
5	Left Centro-Parietal (LCP)	12	Right Temporal (RT)
6	Midline Central (MC)	13	Occipital (O)
7	Right Centro-Parietal (RCP)		

We use MKL to infer information about electrode relevance by identifying the kernel weights learned from training the machine for classification. Each "group" of features is assigned a base kernel, and the linear combination of all base kernels is optimized through gradient descent on the SVM objective function. The optimization of multiple kernels works as a feature selector providing a weighted ranking of the importance of its components. The MKL algorithm is described in detail below.

In the multi-source scenario, for a specific source p, the combined kernel function K between two pixels $\mathbf{x}_i^p$ and $\mathbf{x}_j^p$ can be represented as

$$\begin{aligned} K(\mathbf{x}_i^p, \mathbf{x}_j^p) &= \sum_{m=1}^{M} d_m K_m(\mathbf{x}_i^p, \mathbf{x}_j^p) \\ \text{s.t. } d_m \geq 0, &\text{and } \sum_{m=1}^{M} d_m = 1 \ , \end{aligned} \tag{3.17}$$

where M is the number of candidate basis kernels representing different kernel parameters, K_m is the m-th basis kernel and d_m is the weight for it. Weights can be estimated through cross-validation, which is computationally demanding when the number of basis kernels (i.e., feature sets or data sources) is large. An alternative strategy, which we adopt in this work, is based on the SimpleMKL algorithm [63]. It optimizes the weights automatically in a learning problem by utilizing a gradient descent approach. Based on the SVM optimization problem,

the SimpleMKL learning problem is expressed as

$$\begin{aligned}
&\min_{d} J(d), \text{s.t.} d_m \geq 0, \text{and} \sum_{m=1}^{M} d_m = 1 \\
&J(d) = \left\{ \begin{array}{l} \min\limits_{\mathbf{w},b,\xi} \frac{1}{2} \sum\limits_{m=1}^{M} \frac{1}{d_m} \|\mathbf{w}_m\|^2 + C \sum\limits_{i=1}^{N} \xi_i \\ \text{s.t. } y_i \left(\sum\limits_{m=1}^{M} \langle \mathbf{w}_m, \Phi_m(\mathbf{x}_i^p) \rangle + b \right) \geq 1 - \xi_i \\ \xi_i \geq 0, \forall i = 1, 2, \cdots, N \ , \end{array} \right.
\end{aligned} \tag{3.18}$$

where $\Phi_m(\mathbf{x}_i^p)$ is the kernel mapping function of $\mathbf{x}_i^p$, $\mathbf{w}_m$ is the weight vector of the m^{th} decision hyperplane, C is the regularization parameter controlling the generalization capabilities of the classifier, and ξ_i is a positive slack variable.

Similar to the standard SVM, the above MKL algorithm can also be represented in a dual form as

$$\begin{aligned}
&\max \left\{ L(\alpha_i, \alpha_j) = \sum_{i=1}^{N} \alpha_i - \tfrac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \sum_{m=1}^{M} d_m K_m(\mathbf{x}_i^p, \mathbf{x}_j^p) \right\} \\
&\text{s.t.} \left\{ \begin{array}{l} \sum\limits_{i=1}^{N} \alpha_i y_i = 0 \\ \alpha_i, \alpha_j \in [0, C], \forall i, j = 1, 2, \cdots, N \\ d_m \geq 0, \text{and} \sum\limits_{m=1}^{M} d_m = 1 \ , \end{array} \right.
\end{aligned} \tag{3.19}$$

where α_i and α_j are Lagrange multipliers. The kernel weight d_m can be optimized by updating it along the gradient descent direction of $L(\alpha_i, \alpha_j)$. The gradient of the objective function can be computed as

$$\frac{\partial L}{\partial d_m} = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j K_m(\mathbf{x}_i^p, \mathbf{x}_j^p), m = 1, 2, \cdots, M. \tag{3.20}$$

Then d is updated by using a search scheme as

$$\mathbf{d} \leftarrow \mathbf{d} + \gamma \mathbf{D} \ , \tag{3.21}$$

where γ is the step length, D is the descent direction of $L(\alpha_i, \alpha_j)$, and $\mathbf{d} = [d_1, d_2, \cdots, d_M]^{\mathrm{T}}$ is the kernel weight vector.

Following this optimization, SimpleMKL provides a predicted label for each test sample. However, many applications require a posterior class probability instead of a specific label. Platt proposed an approach to approximate the posterior class probabilities $P(y = 1|x)$ by a sigmoid function which is commonly used in single-kernel SVM implementations [97], [98]. In this research, we implement this approach for our MKL framework in a similar way.

3.3.2 Experimental Settings and Results

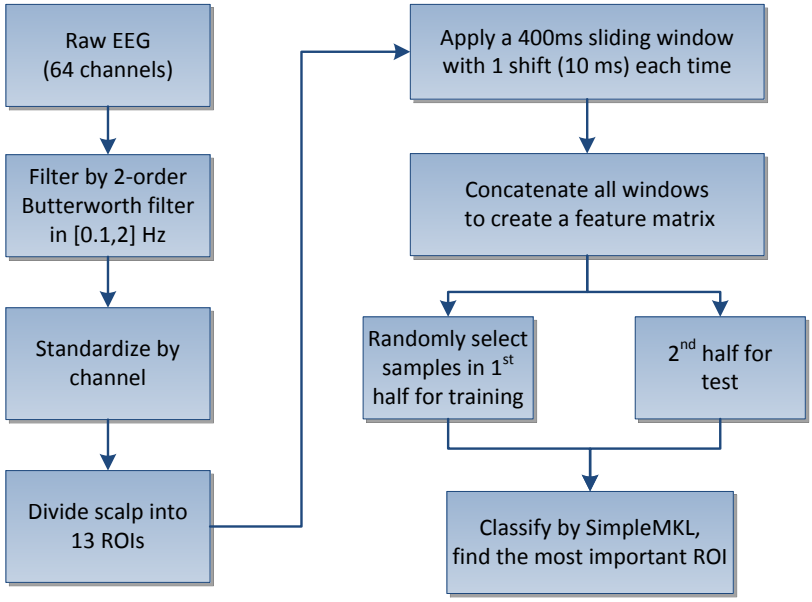


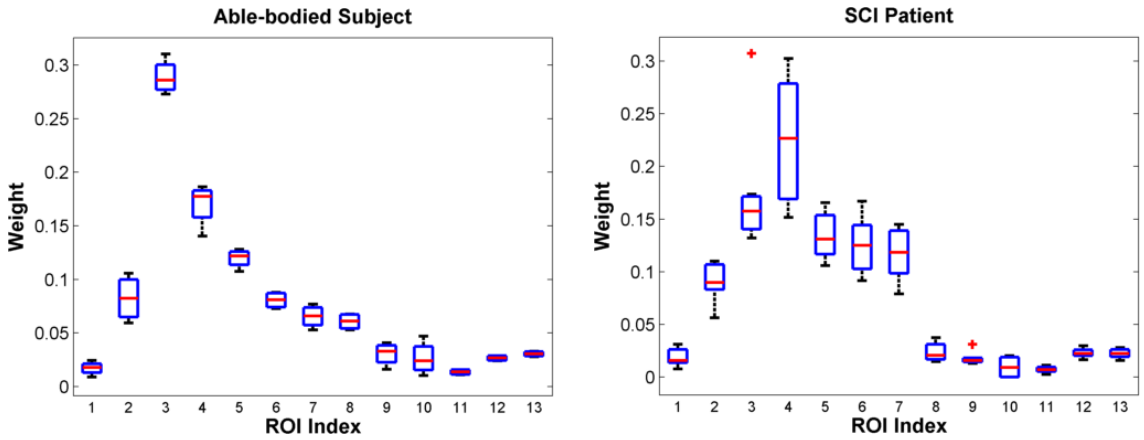
Figure 3.5: Flowchart of the region importance learning framework.

We conduct two experiments to interpret the use of kernel weights in MKL as an indicator of the region importance in decoding of user's movement intention from EEG signals recorded from an able-bodied and a paraplegic subject (who had been trained over multiple sessions to control a robotic exoskeleton (NeuroRex)). In the experiments, data were acquired (100 Hz; 64 channel electrode cap), and filtered (2nd order Butterworth filter in the 0.1-2 Hz range). After standardization, 64 channels were divided into 13 ROIs as described above. The features were

then extracted by applying a 400ms sliding window on each channel with 1 shift (10 ms) each time to acquire the amplitude modulations and concatenated as a feature matrix. To simulate a real-time decoding environment, we randomly select 500 samples from the first half of the labeled samples for training, and the remaining half were used for testing and evaluation. The testing process was repeated 10 times and the metric for evaluating the results of decoding is the average overall accuracy. The flowchart of the proposed framework is shown in Figure 3.5. In this work, RBF kernels were used with the optimal kernel parameter γ chosen by cross-validation. All the experiments were implemented in Matlab using the SimpleMKL toolbox.

1. Four-class, single session classification

First, we compare the kernel weights optimized by SimpleMKL algorithm for different ROIs from an able-bodied subject and a spinal cord injury (SCI) patient. The four motion classes for decoding are walking forward, turning left, turning right and stop. The plots of optimized kernel weights for different ROIs are shown in Figure 3.6.



(a) (b)

Figure 3.6: Comparison of kernel weights for different ROIs from (a) able-bodied subject and (b) SCI patient.

From the results, it is observed that the frontal scalp regions (MF, RF) have the highest weight among all ROIs, which is consistent with the brain regions (frontal lobe) thought to be

involved in the control of lower-limb movements. Moreover, LFC, MC and RFC also have relative high weights, while the other ROIs have low weights, which can be explained corresponding to different importance of the brain regions for movement. In a further experiment, we found that the most important EEG channel in MF region is F1, and in RF is F4. These results demonstrate that MKL can be efficiently used to infer the importance of different groups of features and thus suggest different roles in the representation of gait for different scalp brain areas.

2. Two-class, multiple sessions classification

Second, we conduct a longitudinal experiment from the subjects for a two-class (i.e., walk and stop) classification problem. We quantify electrode relevance changes across sessions to examine neural signatures that may indicate the cortical plasticity triggered by the BMI use. We first plot the weight changes along 9 sessions over a period of 30 days for the SCI patient and able-bodied subject in Figure 3.7 and Figure 3.8, respectively. From the results, we see that the weights change dramatically in the first several sessions, while becoming more stable in the later sessions. Similar to the previous results, the frontal scalp regions (ROI 4 or ROI 3) get the highest weight among all ROIs after training the user to control the exoskeleton for movements for several sessions.

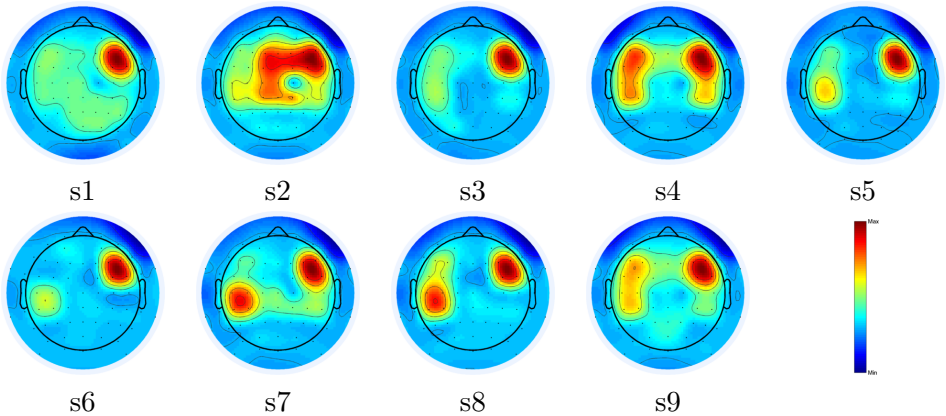


Figure 3.7: Scalp maps of weights along 9 sessions for the SCI patient.

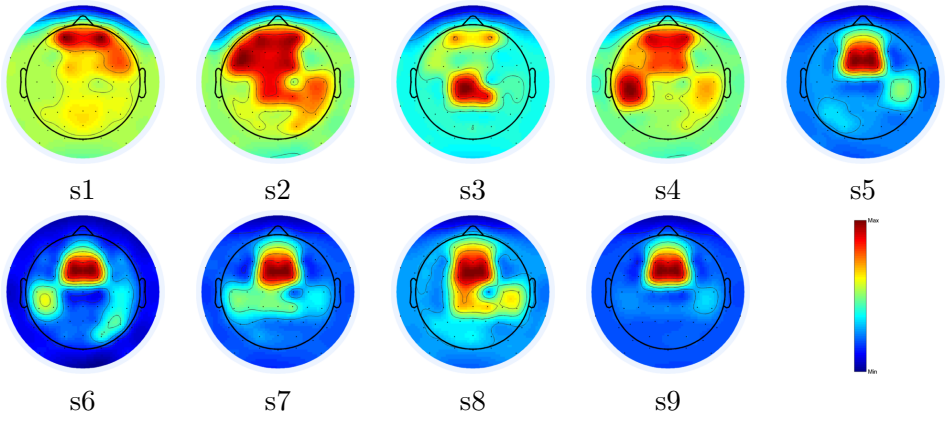
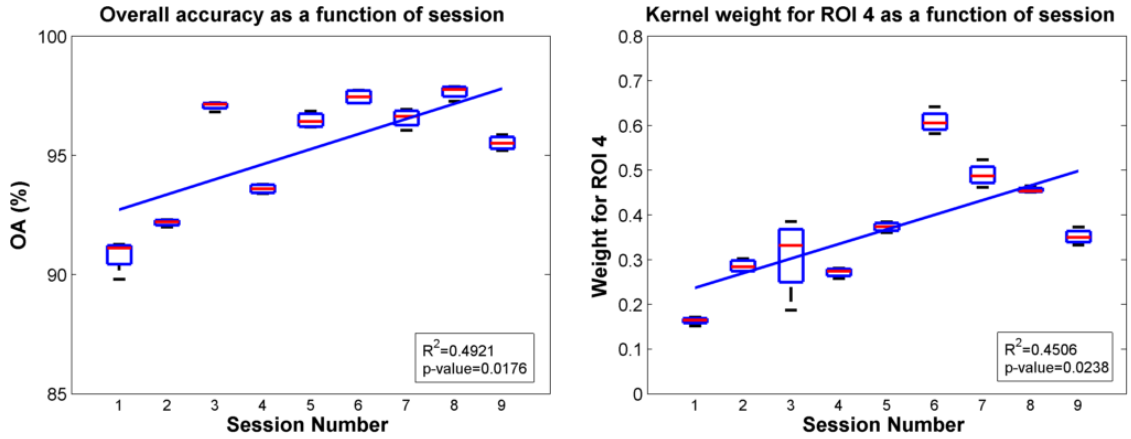
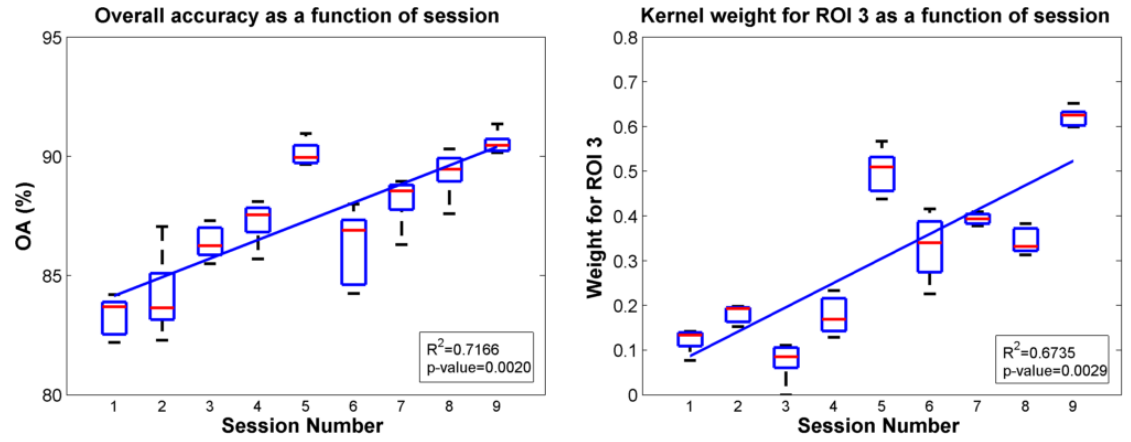


Figure 3.8: Scalp maps of weights along 9 sessions for the able-bodied subject.



(a) (b)

Figure 3.9: Plots of overall accuracy and kernel weight for ROI 4 as a function of session for the SCI patient.



(a) (b)

Figure 3.10: Plots of overall accuracy and kernel weight for ROI 3 as a function of session for the able-bodied subject.

As we know that ROI 4 (ROI 3) is the most significant region for the SCI patient (able-bodied subject) performing lower-limb movements. We further evaluate the overall accuracy and kernel weight for ROI 4 (ROI 3) as a function of session. The linear fit of the relations between OA (weight for the selected ROI) and sessions are shown in Figure 3.9 and Figure 3.10 for SCI patient and able-bodied subject, respectively.

From the longitudinal experiment results, we can see the classification accuracy generally increases as a function of session. At the same time, the weight for ROI 4 (ROI 3) also has the trend of increasing along the session over a period of 30 days. The results demonstrate the cortical plasticity triggered by the BMI use.

3.4 Ensemble Multiple Kernel Active Learning

3.4.1 Proposed Framework

Based on the MKL algorithm, we propose an EnsembleMKL-AL framework for robust classification of multi-source remote sensing data. The flowchart of the proposed framework is shown in Figure 3.11. Consider an initial small set of labeled samples extracted from various sources, noted as $L = \{(\mathbf{x}_i^1, \mathbf{x}_i^2, ..., \mathbf{x}_i^P), y_i\}_{i=1}^N$, where y_i is the label of multi-source data $(\mathbf{x}_i^1, \mathbf{x}_i^2, ..., \mathbf{x}_i^P)$, P is the number of sources, and N is the total number of labeled samples. The goal is to select a series of examples from a set of unlabeled samples $U = \{\mathbf{x}_i^1, \mathbf{x}_i^2, ..., \mathbf{x}_i^P\}_{i=1}^Q$ and add them to the training set after labeling (in a practical framework, this labeling would typically be undertaken by a human analyst, for example, via photo-interpretation or collection of ground reference information). It is desired to obtain good system performance by inducting as few training samples as possible that add the most information. Q is the total number of unlabeled samples, and satisfies $Q \gg N$. The proposed framework can be subdivided into four steps: (1) multiple features are extracted from multi-source data; (2) an ensemble of probabilistic MKL classifiers is implemented to optimize the kernel for each source or feature

set; (3) a maximum disagreement-based AL strategy is used to select the most informative samples; (4) after the learning is accomplished, a decision fusion strategy is applied to the posterior probabilities computed by the MKL classifiers to obtain a final classification map. In this work, a soft fusion strategy — a linear opinion pool (LOP) [99] is studied, and compared with the widely used majority voting (MV) method. The multiple feature extraction and the probabilistic MKL algorithm have been described in previous section 2.2.2 and section 3.3, respectively. So in this section, we only give the description of the maximum disagreement-based AL rule and the decision fusion strategies.

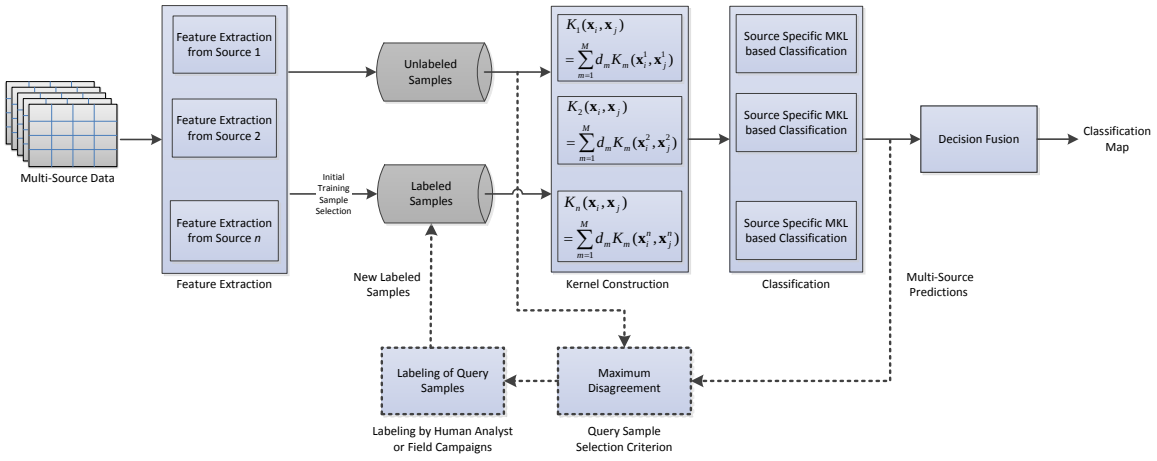


Figure 3.11: Flowchart of EnsembleMKL framework.

Maximum Disagreement Based Ensemble Active Learning

Ensemble AL is based on a committee of learners, in which each member of the committee is learned on a subset of the samples or of the feature space. Diversity is important for building a robust ensemble [75], [57], as it ensures that each subset contains additional information to improve the learner relative to the other subsets. In previous work with hyperspectral imagery, these conditions were created via a multi-view approach, in which the original set of spectral features was partitioned into disjoint subsets, i.e., different views [75]. A similar approach can

be adopted in our case, in which different sources or different types of features can be assigned to a different view of the data.

In an AL framework, the choice of criterion function used to select samples from the unlabeled set is crucial. In the context of ensemble AL we adopt the maximum disagreement criterion, which demonstrated its capabilities in identifying the most informative samples across multiple views [75]. It is based on two successive steps.

Let us suppose that the estimated label of a sample $\mathbf{x}_i^p \in U$ from source/feature-set p is obtained by learning a classification function, and $\hat{y}_i^p = f(\mathbf{x}_i^p)$. For each sample i, we define a symmetric matrix D_i, which measures the disagreement between each pair of predictions $(\hat{y}_i^1, \hat{y}_i^2, ..., \hat{y}_i^P)$. Each element of the matrix is defined as

$$D_i(p,n) = \begin{cases} \Delta(\hat{y}_i^p, \hat{y}_i^n), & \text{if } p \neq n \\ 0, & \text{if } p = n, \end{cases} \tag{3.22}$$

where

$$\Delta(\hat{y}_i^p, \hat{y}_i^n) = \begin{cases} 1, & \text{if } \hat{y}_i^p \neq \hat{y}_i^n \\ 0, & \text{if } \hat{y}_i^p = \hat{y}_i^n \ . \end{cases} \tag{3.23}$$

$p, n \in \{1, 2, ...P\}$. Then, the disagreement level of sample i over all sources can be expressed as

$$DL_i = \sum_{p=1}^{P} \sum_{n=1}^{P} D_i(p,n). \tag{3.24}$$

The maximum disagreement contention set P_{MD} is constructed by selecting unlabeled samples with the maximum uncertainty, i.e., the maximum disagreement level. If the number of samples with the highest disagreement level is less than the batch size, i.e., the number of samples to select, more samples having the maximum disagreement levels are included in

P_{MD}. In this way, P_{MD} is always larger than the batch size. Note that because we want to keep the size of the training set small, a small fixed number of samples should be carefully selected in each learning step. However, the samples belonging to the maximum disagreement set are usually characterized by strong redundancy. To limit this, and to select non-redundant samples, a pruning strategy is applied on P_{MD}.

We use weighted voting entropy (WVE) [100] to quantitatively measure the uncertainty of votes over labels provided by each source. For this purpose, we define a $P \times N_c$ weighting matrix $\mathbf{W}$ (where N_c is the number of classes), in which $\mathbf{W}(p, c)$ is the class-specific accuracy for source p and class c. The WVE value of $\mathbf{z}_i \in P_{MD}$ at the τ^{th} query is defined as

$$WVE^{\tau}(\mathbf{z}_i) = -\frac{1}{\log \varpi^{\tau}} \sum_{c=1}^{N_c} \frac{\sigma_c^{\tau}(\mathbf{z}_i)}{\varpi^{\tau}} \log \left(\frac{\sigma_c^{\tau}(\mathbf{z}_i)}{\varpi^{\tau}} \right) \ , \tag{3.25}$$

where

$$\varpi^{\tau} = \sum_{p=1}^{P} \sum_{c=1}^{N_c} \mathbf{W}^{\tau-1}(p, c) \ , \tag{3.26}$$

$$\sigma_c^{\tau}(\mathbf{z}_i) = \sum_{p=1}^{P} \mathbf{W}^{\tau-1}(p, c) \times \delta \left(f(\mathbf{z}_i^p) \right) \ , \text{ and} \tag{3.27}$$

$$\delta \left(f(\mathbf{z}_i^p) \right) = \delta(\hat{y}_i^p) = \begin{cases} 1, \text{ if } \hat{y}_i^p = y_i \\ 0, \text{ if } \hat{y}_i^p \neq y_i \ , \end{cases} \tag{3.28}$$

$\mathbf{W}^{\tau-1}$ is the weighting matrix from the last query. Following this, only samples with the highest entropy values are selected and inserted into the final set of informative samples

$$P_{WVE}^{\tau} = \left\{ \mathbf{z}_j^1, \mathbf{z}_j^2, ..., \mathbf{z}_j{}^P : \max WVE^{\tau}(\mathbf{z}_j) \right\} \ . \tag{3.29}$$

The combination of the maximum disagreement method with the pruning strategy based on WVE allows us to select samples having the highest disagreement level while simultaneously exhibiting poor classification performance.

Decision Fusion Strategy

After the learning is accomplished, each source specific MKL classifier can output the predicted labels as well as their corresponding posterior probabilities for all samples. Following this, a decision fusion strategy is applied to perform classification per pixel. Decision fusion can occur either at the class label level, known as hard fusion, or at the posterior probability level, known as soft fusion.

Majority voting — a popular approach to conduct *hard decision fusion*, achieves the final classification decision based on a vote over individual class labels from each classifier in the ensemble. A simple majority voting is given by

$$w = \underset{i \in \{1,2,...,N_c\}}{\arg\max} \; N(i) \; , \tag{3.30}$$

where w is the class label from one of the N_c possible classes for the test sample, and $N(i)$ is the number of times that the class i predicted by the ensemble of classifiers.

Soft decision fusion makes the use of posterior probabilities for making the final decision. A popular soft decision fusion scheme is a linear opinion pool, which makes the final classification decision by constructing a global membership function by using individual posterior probabilities $p_j(w_i|\mathbf{x})$ of each classifier

$$P(w_i|\mathbf{x}) = \sum_{j=1}^{P} \alpha_j p_j(w_i|\mathbf{x}) \text{ and} \tag{3.31}$$

$$w = \underset{i \in \{1,2,...,N_c\}}{\arg\max} \; P(w_i|\mathbf{x}) \; , \tag{3.32}$$

where α_j $(j = 1, 2, ..., P)$, is the classifier weight, which can either be uniformly distributed over all classifiers, or can be assigned based on the "confidence score" of each classifier. In this work, we use the uniformly distributed weight for each source classifier.

3.4.2 Experiment Settings and Results

3.4.2.1 Experimental Setting

We present experimental results using UH multi-sensor datasets to demonstrate the efficacy of the EnsembleMKL-AL approach. First, we compare SimpleMKL-AL with standard single kernel SVM-AL and verify that MKL is a suitable classifier for multiple feature AL. In the second set of experiments, we compare the proposed EnsembleMKL-AL system to SimpleMKL-AL to investigate the benefit of using ensemble classifiers. Finally, we quantify the efficacy of multiple features, especially the morphological and textual features, utilized in AL.

From the available labeled data, half of the samples were selected randomly as our query set. The remaining pixels constituted the test set. Ten randomly sampled splits were used, and the average results over these random splits are reported in all experiments. The initial training set contained 20 samples for each class randomly selected from the query set. At each learning step, 5 samples were selected from the candidate pool and added to the training set based on learning system specific query criteria. For the single-classifier learning system, the criterion was margin sampling (MS) and the baseline was random sampling (RS). For the ensemble classifier learning system, we employed maximum disagreement (MD) for AL, and a final decision was made based on majority voting (MV) or linear opinion pool (LOP) fusion.

All experiments were conducted using an RBF kernel function with relative width parameter σ. For the standard SVM, this parameter was estimated before the learning process by applying kernel alignment [101] to the initial training set. Starting from 0.05 and with a step size of 0.05, the alignment of the kernel was maximized for $\sigma = 0.9$. For the MKL-based experiments, we did not select a specific kernel parameter; instead, we defined a set of different values as candidate input parameters. In a multi-source scenario, we can build several basis kernels with different values of σ for each source, however, the number of parameters should be

kept small to reduce the computational complexity and memory requirements. In particular, four base kernels with $\sigma = [0.2, 0.5, 1, 1.5]$ were considered for all sources. This range of values was found to be reasonable after applying kernel alignment to the initial training set of each source. For all classifiers (i.e., standard SVM and MKL), the penalty parameter C was selected by cross-validation in the range of $[2^{-1}, ..., 2^{15}]$.

3.4.2.2 Comparison of SimpleMKL-AL and SVM-AL

The first experiment compares results obtained by the SimpleMKL-AL and SVM-AL algorithms, to investigate the potential of MKL-AL in processing a large number of features obtained from different sensors using different spatial feature extraction strategies. The learning curves of the different AL methods are shown in Figure 3.12.

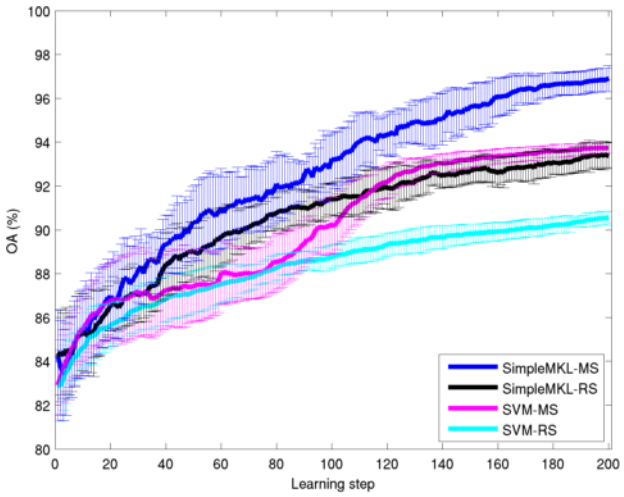


Figure 3.12: OA achieved on the UH dataset for SimpleMKL and SVM methods. RS: random sampling; MS: margin sampling.

In general, the SimpleMKL-AL methods are superior to the standard SVM-AL methods for both MS and RS query criteria throughout the learning processes. Initially, when the training samples are randomly selected and the number is the same for each class, the benefit of SimpleMKL-AL learners is not obvious. For the SimpleMKL-AL learners, the overall average accuracy is 84.24%, compared to 82.92% of SVM-AL learners. With increasing learning steps, SimpleMKL learners start to show advantage, especially for the MS strategy. This is because

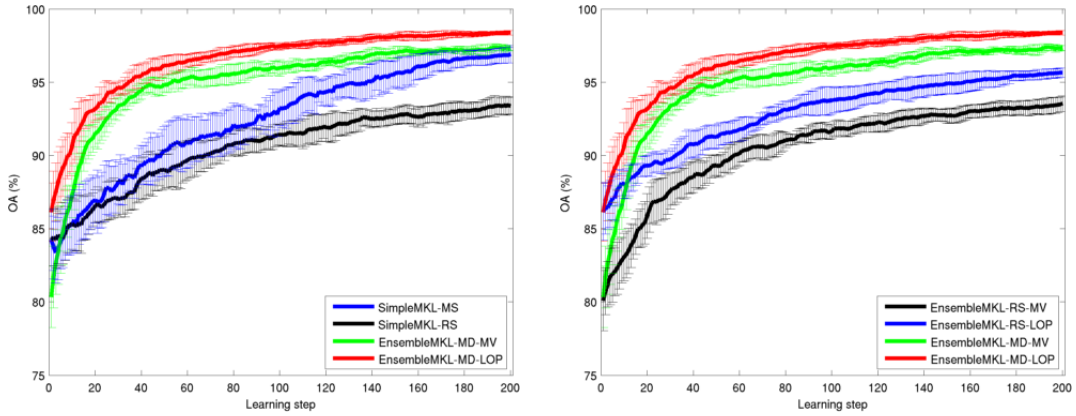
SimpleMKL is able to optimize a combination of kernels that jointly maximize the sum of margins, which are crucial to determine the hyperplane between classes. At the 200th learning step, the overall accuracies for SimpleMKL-MS and SimpleMKL-RS are 96.90% and 93.35% respectively, compared to 93.37% and 90.55% for SVM-MS and SVM-RS, respectively. We can hence conclude that SimpleMKL has greater potential for handling a larger number of features extracted from different sources than standard SVM.

3.4.2.3 Comparison of Ensemble and Single AL system

In order to adapt a classifier specific for each data source and improve the total performance of AL, we develop a source-specific MKL-AL algorithm using ensemble classifiers. As noted previously, MS is a good strategy for single-classifier AL system, but it is not suitable for an ensemble system, because it cannot exploit the potentially diverse information across different sources. In the second part of the experiments, the proposed MD-based EnsembleMKL method is compared with the single-classifier MS-based SimpleMKL method. For EnsembleMKL, two decision fusion strategies (i.e., MV and LOP) are applied after each learning step to generate the classification map and assess the classification accuracies. For completeness, the RS criterion is also considered. The obtained results are reported in Figure 3.13 (a) and (b).

In Figure 3.13 (a), results from the EnsembleMKL-MD methods are compared to those from the single-classifier based SimpleMKL-AL methods. The proposed method achieved higher overall accuracies in general. Among all learning strategies, the EnsembleMKL-MD-LOP performs the best with an overall accuracy of 86.13% at the beginning with 20 samples each class and 98.38% after 200 iterations. EnsembleMKL-MD-MV learns no better than the SimpleMKL-MS learner at the first 5 steps, but it starts to improve after that. Both ensemble methods show significant improvements compared to the baseline SimpleMKL-RS, the increase

of the accuracy is 5.00% and 3.95% respectively for LOP and MV at the final step. Furthermore, it is much faster for EnsembleMKL learners to reach a high accuracy and converge than SimpleMKL learner. In Figure 3.13 (b), we compare the MD and RS criteria in conjunction with EnsembleMKL method. It is evident for both decision fusion strategies (i.e., MV and LOP), the MD-based AL strategy produces higher accuracies than RS. Therefore, the MD criterion, which has previously been shown to work well as a multi-view method for single-source hyperspectral data [75], has been demonstrated to be also suitable for this multi-source scenario.



(a) (b)

Figure 3.13: OA achieved on the UH dataset for (a) SimpleMKL and EnsembleMKL and (b) EnsembleMKL-RS and EnsembleMKL-MD methods. RS: random sampling; MS: margin sampling; MD: maximum disagreement.

3.4.2.4 Class specific analysis

The class accuracies and statistical number of samples selected from each class by different MKL-AL strategies are shown in Figure 3.14 and 3.15. From the results in Figure 3.14, most of the samples selected by SimpleMKL-MS are from the classes with low accuracies at the beginning, such as Class 9: Road, Class 10: Highway, and Class 12: Parking Lot 1. When samples from these classes are added to the training set, there is an obvious increase of accuracy

during the learning. For example, in Figure 3.14, the accuracy of Class 9 is 74.49% at the beginning when only 20 randomly selected samples are used. As the learning progresses, the accuracy increases to 84.47% and 95.59% after 100 and 200 iterations, and the corresponding number of selected samples from Class 9 is 98 and 208 respectively. For EnsembleMKL-AL in Figure 3.15, the selection of samples from different classes seems to be more uniform than in SimpleMKL-AL. As a result, it increases the accuracies of most classes and thus overall accuracy as well.

Comparing class-specific accuracies in Figure 3.14 and 3.15, it is clear that the ensemble system with multiple features can improve the accuracies of most classes. The most significant effect of bringing multiple features in this ensemble system is the improvement in discrimination between the residential and commercial area as well as different types of roads. This is due to different morphological and textures present in these urban structures. To be more specific, the residential and commercial buildings are typically differentiated by their shapes and sizes, and these properties may be directly related to different scales of structuring elements in EMAPs. The railway and highway classes are morphologically similar, but are constructed with different materials and thus texture features can help to differentiate these classes to some extent. Table 3.4.2.4 summarizes the class specific accuracies, overall accuracy and standard deviations for different AL approaches at the final learning step. Classification maps obtained at the final step of SimpleMKL-MS and EnsembleMKL-MD-LOP are reported in Figure 3.16.

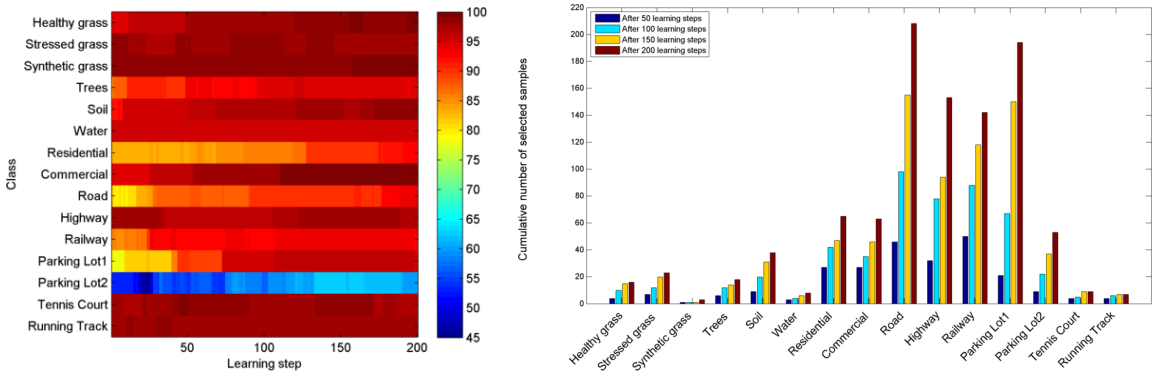


Figure 3.14: Class specific accuracies (left) and cumulative number of selected samples (right) at different learning steps for SimpleMKL-MS.

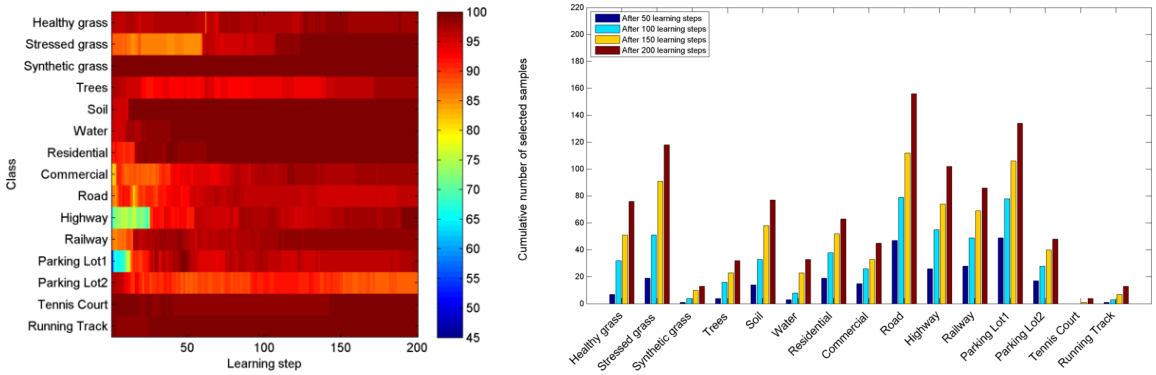


Figure 3.15: Class specific accuracies (left) and cumulative number of selected samples (right) at different learning steps for EnsembleMKL-MD-LOP.

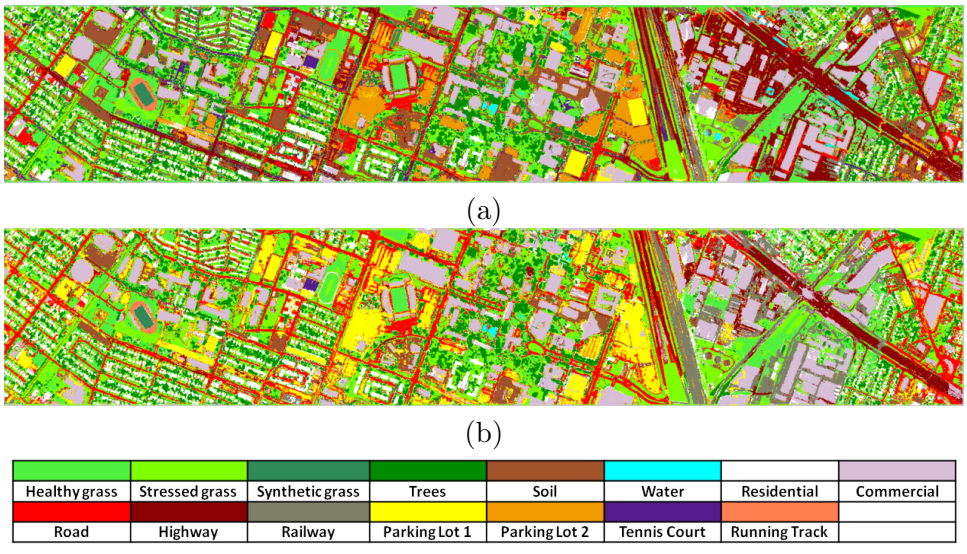


Figure 3.16: Classification maps obtained at the final AL step. (a) SimpleMKL-MS; (b) EnsembleMKL-MD-LOP.

Table 3.5: Class accuracies (%), overall accuracy (OA%), and standard deviations (Std.%) for different AL methods

Class Index	SVM RS	SVM MS	SimpleMKL RS	SimpleMKL-MS MS	Ensemble MKL LOP
1	97.27	98.80	98.10	**99.22**	98.12
2	97.45	97.37	98.87	98.14	**99.69**
3	98.23	99.39	98.27	99.38	**100**
4	96.21	96.16	97.24	94.29	**98.04**
5	97.15	98.63	98.41	98.72	**99.52**
6	93.15	95.36	93.21	95.51	**99.36**
7	90.11	92.83	90.31	92.41	**99.84**
8	90.77	90.94	94.31	**99.68**	97.94
9	88.59	93.53	91.17	94.18	**96.28**
10	93.05	96.12	92.28	97.30	**99.68**
11	90.74	92.16	91.95	93.28	**98.66**
12	75.06	77.39	73.19	95.59	**98.14**
13	78.18	83.31	79.77	65.52	**87.92**
14	97.56	98.73	96.62	96.68	**99.53**
15	96.55	96.01	98.93	97.97	**99.71**
OA	90.55	93.37;	93.35	96.9	**98.38**
Std.	0.27	0.24	0.58	0.6	**0.15**

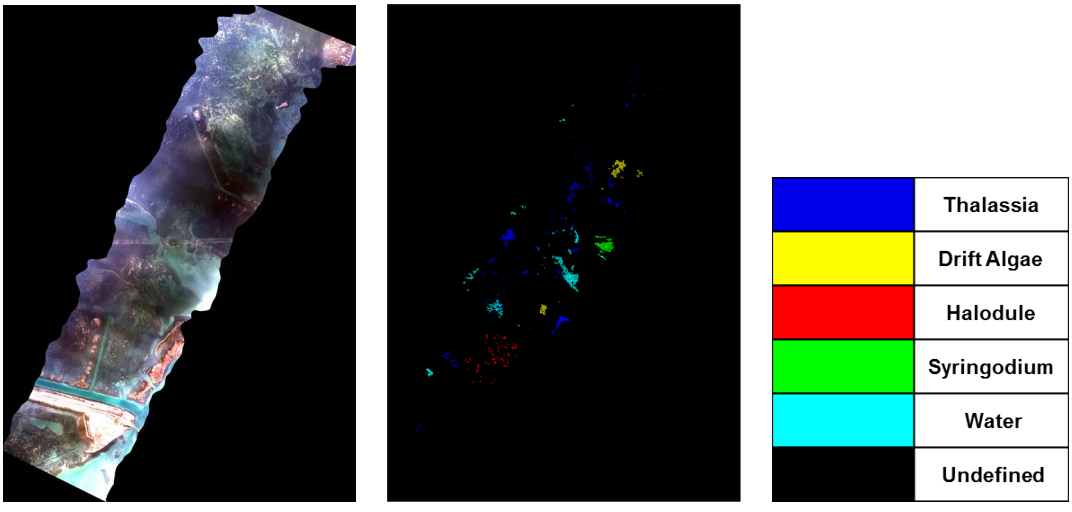
3.4.2.5 Computational Cost

We conclude the experimental analysis by empirically evaluating the computational complexity associated with the different methods investigated in this paper. All the experiments were implemented in Matlab R2012a on a Linux system with twelve 3.2GHz Intel(R) processors and 32GB RAM. The SimpleMKL toolbox [102] was adopted for implementing single kernel SVM and MKL approaches. The total processing time, which includes model selection, training phase and sample selection, for running the 200 steps of the AL process and by considering the six different sources was 9.35×10^3 s, 7.88×10^3 s and 8.19×10^3 s for SVM-MS, SimpleMKL-MS, and EnsembleMKL-MD-LOP, respectively.

3.4.3 Application on Seagrass Mapping

We apply the proposed EnsembleMKL-AL framework for the mapping of seagrass in the Redfish Bay, Texas (27°54′47.01″N 97°6′25.73″W) by using airborne hyperspectal radiance

data (acquired by CASI-500 hyperspectral sensor) and bathymetric LiDAR data (acquired by Optech Aquarius sensor). The hyperspectral image has 72 bands over the wavelength range from 366 nm to 1043 nm at a spatial resolution of 1.5 m. The LiDAR data was acquired at a wavelength of 532 nm — high resolution discrete return and full-waveform data were acquired. The Aquarius sensor head is a single frequency shallow water bathymetric LiDAR with a pulse energy of 30 μJ (at 70 kHz), and a beam divergence of 0.8 mrad. The pseudo-waveform data were then generated from the original LiDAR point cloud and used as a second source for fusion. The Coastal and Marine Geospatial Lab of the Harte Research Institute for Gulf of Mexico Studies at Texas A&M University-Corpus Christi conducted a coordinated ground survey using an airboat to collect field measurements of benthic coverage within the mapped area of Redfish Bay. The ground survey identified three different types of seagrass within the study area, which are Halodule, Syringodium, Thalassia. Halodule and Thalassia had a much higher presence in the study area compared to Syringodium. Drift algae was also commonly observed in the study area. The composite image of the hyperspectral data and its groundtruth are shown in Figure 3.17.



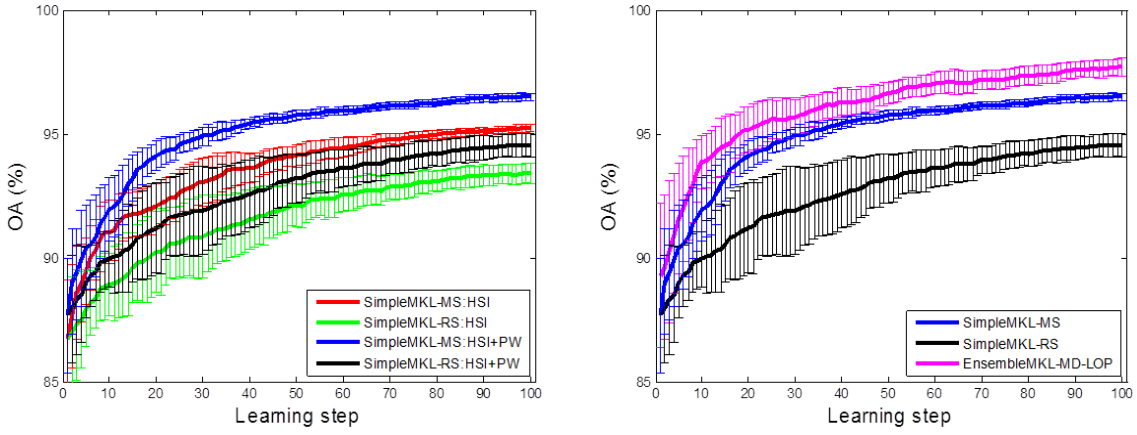
(a) (b)

Figure 3.17: Corpus Christi (CC) dataset. (a) Composite image of the hyperspectral data; (b) Groundtruth map.

Acquiring ground reference data, particularly over a bay is a very challenging task. To expand the size of the spectral library for a robust classification of the Hyperspectral and LiDAR data, we grew the area via spatial-spectral segmentation through HSeg. Then we employed the photo-interpretation using very high resolution (5 cm) color images to remove incorrectly labeled pixels. Following this strategy, we created a spectral reference library, and added additional background class — water.

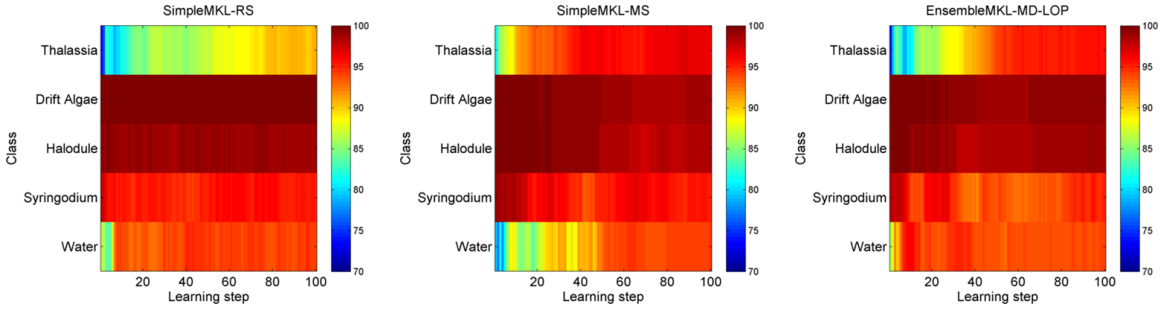
Further, we apply the active learning methods to assist us in identifying the "most informative" samples in the datasets that when labeled and added to the reference library, substantially enhance the classification mapping performance. We conduct similar experiments as stated in the previous sections. The AL experiments started with 30 labeled samples per class randomly selected from the query set, and in each iteration, 5 samples were selected from the candidate pool based on the proposed query criteria. The AL learning curves comparing different methods are shown in Figure 3.18. Particularly, we first compare the single source classification results with the multi-source fusion results, and then compare the proposed EnsembleMKL-AL with the SimpleMKL-AL. Similar to the results acquired from UH data, EnsembleMKL-MD-LOP achieves the best performance than the baseline AL algorithms.

We also show the class specific accuracies and the classification maps in Figure 3.19 and Figure 3.20, respectively. From the results, the Thalassia and Water have relative low accuracies at the beginning when the number of training samples is small. However, after several steps, the accuracies increase quickly, especially when EnsembleMKL-MD-LOP is used for AL. These results demonstrate that the proposed EnsembleMKL-AL is an effective approach to select select important samples for wetland seagrass mapping.



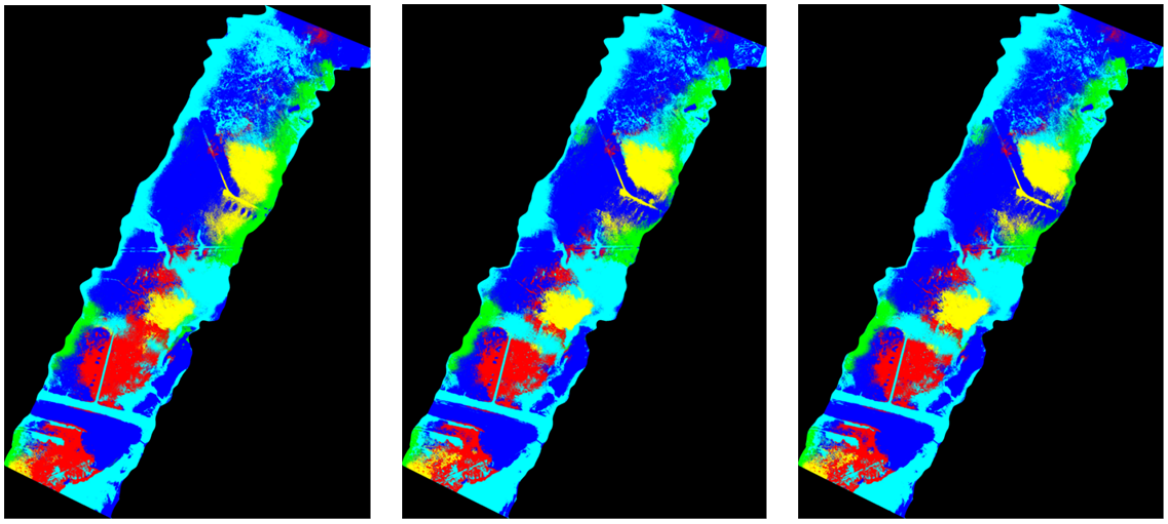
(a) (b)

Figure 3.18: AL learning curves on the CC dataset for (a) SimpleMKL-based single source AL and multi-source AL and (b) multi-source fusion results using SimpleMKL and EnsembleMKL.



(a) (b) (c)

Figure 3.19: Class specific accuracies achieved on the CC dataset for (a) SimpleMKL-RS, (b) SimpleMKL-MS and (c) EnsembleMKL-MD-LOP.



(a) (b) (c)

Figure 3.20: Classification maps obtained at the final AL steps for (a) SimpleMKL-RS, (b) SimpleMKL-MS and (c) EnsembleMKL-MD-LOP.

3.5 Summary

In this chapter, mixture-of-kernels based methods have been developed as an effective approach to fuse multi-source data for classification from various aspects. First, we propose a composite-kernel-based feature extraction method for multi-source data classification. CKLFDA is built upon the foundations of KLFDA, replacing the single kernel in KLFDA by a weighted composite kernel, which can be viewed as an effective *feature fusion* strategy. To demonstrate the benefits of CKLFDA, we conduct experiments on both multi-feature and multi-sensor remote sensing data. The experimental results validate the hypothesis that CKLFDA serves as an effective and robust feature extraction tool for linear classifiers. CKLFDA-MLR outperforms all the other traditional methods in terms of overall classification accuracy.

MKL has the advantage of learning the classifier and the optimal kernel weights simultaneously. In this chapter, we investigate this property and apply it to infer the relative importance of different groups of features (different sources of information) in a BMI application to decode

one's motion intention from the EEG signals. The experimental results demonstrate that the frontal/frontal-central regions are the most important regions for movement decoding, which is consistent with the brain regions hypothesised to be involved in the control of lower-limb movements. In addition, we demonstrate the cortical plasticity triggered by the BMI use, as the decoding accuracy and the weights for important regions generally increase while the user learns to control the exoskeleton for movement for sessions.

In addition, we present an ensemble multiple kernel based AL system to incrementally select informative samples across multi-views for labelling. Data from different sources provide the necessary diversity which is a crucial point for constructing the classifier committee in AL. This framework provides a new way to exploit multi-sensor, multi-feature remote sensing datasets for image classification. The experiments validated the efficacy of the proposed framework and provided the following conclusions — (1) MKL is a more effective and appropriate classifier for multi-source AL compared to the standard SVM classifier; (2) Ensemble classifiers improve the performance of traditional AL substantially for this multi-source data. The proposed EnsembleMKL-AL system greatly outperforms the SimpleMKL-AL approach in terms of overall and class-specific accuracies.

Chapter 4

Multi-source Data Fusion via Locality Driven Joint Sparse Representation

4.1 Introduction

From a signal processing point of view, a challenge for multi-source data analysis is how to effectively combine different features and represent them in a compact way. As most natural signals are inherently sparse in a certain basis or dictionary, they can be compactly represented by only a few coefficients that carry the most important information. In other words, the intrinsic signals in the same class typically lie in a low-dimensional subspace and the semantic information is often encoded in a sparse representation with respect to an appropriate basis.

In recent years, the compressed sensing and sparse representation theories have emerged as powerful tools to reconstruct and represent signals by decomposing the sample over a usually overcomplete dictionary generated by or learned from representative samples. Further, sparse representation based classification (SRC), which combines the discrimination power with the reconstruction property and notions of sparsity, has been demonstrated as an effective and robust method for many pattern recognition applications including hyperspectral imgae classification and BCI [35–40, 103–105].

To combine disparate features for better classification performance while utilizing the properties of SR, a simple strategy is to concatenate the features and obtain a unified sparse representation by a conventional sparse representation classifier. Although this method is straightforward, the resulting input space may be high dimensional, and can also cause overfitting. Inspired by the study of sparsity in multi-task learning [106], [107], Yuan et al. proposed a multi-task joint sparse representation based classification method (MTJ-SRC), which treats recognition with multiple features as a multi-task learning problem [108]. It is assumed that the coefficients share the same sparsity pattern among all features, and a class-level joint sparsity-inducing regularizer is used to combine features for classification. Shekhar et al. extended the MTJ-SRC work to a more general case that can be used for both multi-task and multivariate sparse representation by imposing an ℓ_1/ℓ_q regularization on the concatenated sparse coefficient matrix [109]. Zheng et al. proposed a framework based on MTJ-SRC with spatial filtering post-processing for large-scale satellite-image annotation and achieved higher classification accuracies [110]. Along a similar direction, Li et al. proposed a joint collaborative representation classification method with multitask learning (JCRC-MTL) and incorporated contextual neighborhood information for hyperspectral image classification [111].

In this chapter, we propose a joint sparse representation model with an adaptive locality weight to jointly represent multi-source data while adapting the weights to constrain the sparse coefficients for better signal representation. Different from the previous works that either fail to consider the difference between sources or only consider locality structure within one source, we adapt the locality information for each source of data in an iterative way to reduce the estimation bias. To solve the optimization problem, we apply an efficient alternative direction method. The effectiveness of the proposed weight structures for joint sparse representation is validated on the multi-source remote sensing data and the multiscale EEG data.

4.2 Limitations with the Previous Works

In MTJ-SRC, data from multiple sources are jointly represented by a sparse linear combination of the training data, based on the assumption that samples from different sources belonging to the same class have unified sparse support distributions on their coefficient vectors. To learn the joint sparsity of coefficients, the goal is to obtain a row-sparse coefficient matrix which can be modeled as an ℓ_1/ℓ_q-regularized least square problem. For a test sample $\mathbf{x} = [\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^D]$ from D sources, given the dictionary $\{\mathbf{A}^i\}_{i=1}^D$, the joint sparse coefficient matrix $\mathbf{S} = [\mathbf{s}^1, \mathbf{s}^2, ..., \mathbf{s}^D] \in \mathbb{R}^{n \times D}$ can be estimated by

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}} \frac{1}{2} \sum_{i=1}^{D} \left\| \mathbf{x}^i - \mathbf{A}^i \mathbf{s}^i \right\|_2^2 + \lambda \|\mathbf{S}\|_{1,q}, \tag{4.1}$$

where λ is a positive regularization parameter, and $\|\mathbf{S}\|_{1,q}$ is the ℓ_1/ℓ_q norm defined as $\|\mathbf{S}\|_{1,q} = \sum_{k=1}^{n} \left\|\tilde{\mathbf{s}}^k\right\|_q$, where each $\tilde{\mathbf{s}}^k \in \mathbb{R}^D$ is a row vector of $\mathbf{S}$. To make the function convex, q is commonly set to be larger than 1 (usually as 2). By solving the ℓ_1/ℓ_q optimization, the solution of the sparse coefficient matrix has common support at the column level.

It has been demonstrated that for multi-source data MTJ-SRC and its extensions can achieve better classification results than the classical SRC [108], [109]. However, the assumptions made in this approach leads to a few limitations. First, the MTJ-SRC treats different sources equally and hence does not consider the intrinsic value of each dictionary to the underlying task. In other words, all atoms from multiple sources share the same regularization for signal representation, which in reality may be too restrictive for classification and may lead to sub-optimal performance. To address this problem, a natural approach is to use a weighted regularizer to penalize sources and atoms differently for sparse representation. Ideally, if there is some prior information on the importance of each source, the weight can be adapted from prior knowledge. However, in many situations, the prior knowledge is not available or is not

accurate — thus it can be helpful to automatically estimate such a weight from the dictionary itself.

Second, as in the standard SRC, MTJ-SRC makes the assumption that there are sufficient training samples in the dictionary such that all possible variations can be covered when representing each class. Thus, such a method may fail when the dictionary has a small number of samples [112]. Additionally, for hyperspectral data, since the samples are typically highly correlated, ℓ_1-induced SRC may have unstable estimations of the representation coefficients. This problem has been illustrated in [113] wherein similar test samples (or even the same test sample) have very different representations due to such *instability* (i.e., non-uniqueness of the sparse coefficients) of sparse decompositions. The instability of sparse decompositions contributes to incorrect classification. To solve the problem of unstable estimation in SR, previous works have shown that locality information among samples is effective and crucial for addressing this issue and enhancing the classification performance [113–117]. In [113], for example, local structure of the test samples is enforced based on manifold learning for smoothing the sparse representations. Locality information has also been incorporated into linear coding [114], sparse and group sparse representation [115], [116] by calculating the similarity between the test sample and distinct classes.

Different from the previous works that either fail to consider the difference between sources or only consider locality structure within one source, we adapt the locality information for each source of data in an iterative way to reduce the estimation bias.

4.3 Proposed Method

4.3.1 Multi-source Joint Sparse Representation for Classification

Motivated by the success of MTJ-SRC, we develop a related framework for classification of multi-source data. In this work, each task is actually a specific source of disparate features from

the same or different sensors. We propose that this framework must be modified suitably to best address some of its key limitations — we provide such a modification and develop related algorithms that solve the minimization problem. The block diagram providing an overview of the proposed framework is shown in Figure 4.1 using the EEG multiscale signals for a gait phases decoding problem.

In MTJ-SRC, by employing the ℓ_1/ℓ_2 norm, all the atoms in the dictionary are treated equally for signal representation, which ignores the data differences within each source and among different sources. The regularization parameter λ in equation (4.1) that controls the sparsity level of joint sparse coefficients is uniform for all atoms in the dictionary. This condition may be too restrictive for multi-source data sparse representation, because it does not consider the differences among sources and fails to preserve the information of data locality between the test sample and training samples.

To address these problems, we impose an adaptive source-specific weight on the joint sparse regularization term. Similar to the weighted ℓ_1-minimization problem, MTJ-SRC can be modified by imposing a weight on the ℓ_1/ℓ_2 regularization as

$$\begin{aligned} \hat{\mathbf{S}} &= \arg\min_{\mathbf{S}} \tfrac{1}{2} \sum_{i=1}^{D} \left\|\mathbf{x}^i - \mathbf{A}^i\mathbf{s}^i\right\|_2^2 + \lambda\|\mathbf{W} \odot \mathbf{S}\|_{1,2} \\ &= \arg\min_{\mathbf{S}} \tfrac{1}{2} \sum_{i=1}^{D} \left\|\mathbf{x}^i - \mathbf{A}^i\mathbf{s}^i\right\|_2^2 + \lambda \sum_{j=1}^{n} \|\mathbf{w}_j \odot \mathbf{s}_j\|_2 \end{aligned}, \tag{4.2}$$

where $\odot$ denotes the element-wise multiplication. $\mathbf{W}$ is the weighting matrix, and $\mathbf{w}_j$ is the source-specific weight for atom j ($j = 1, ..., n$), which can be expressed as $\mathbf{w}_j = [w_j^1, w_j^2, ..., w_j^D]$ for D sources.

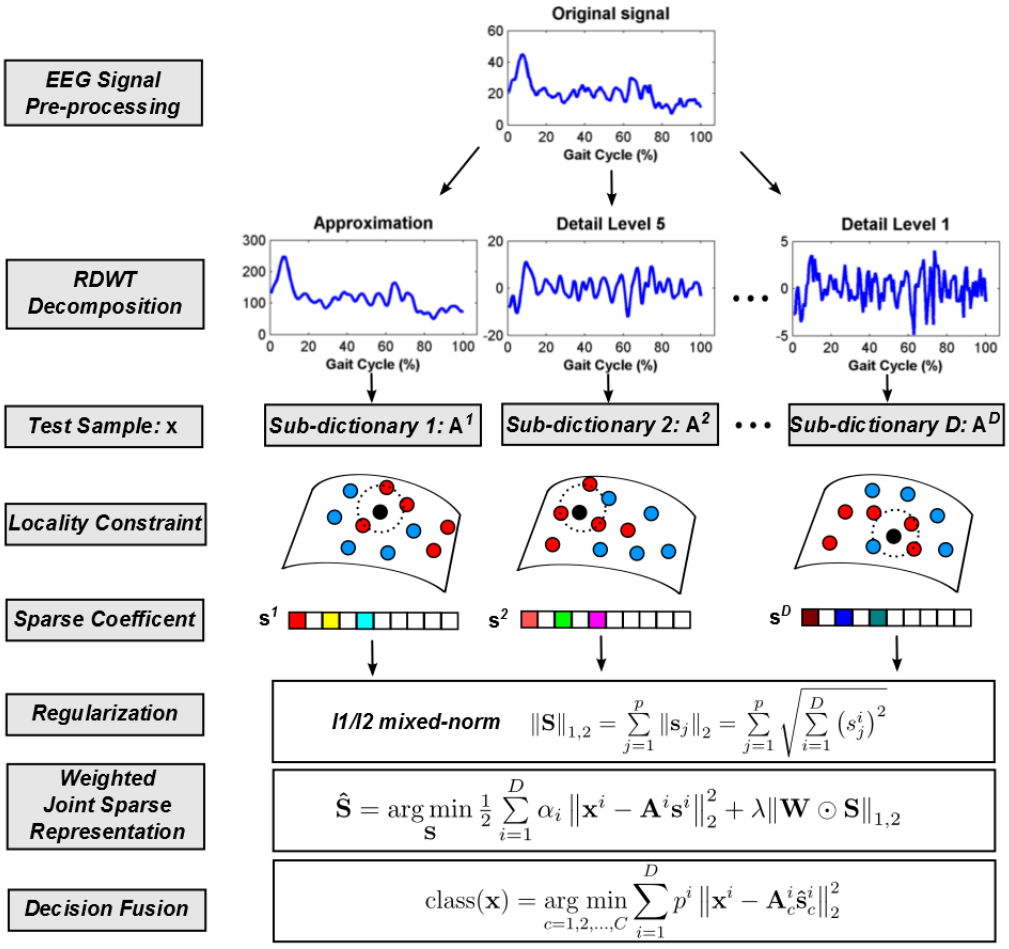


Figure 4.1: Block diagram of the proposed ALWMJ-SRC framework.

The locality or contextual information has been shown to have great importance to the classification performance, due to the fact that samples close to the input test sample are more likely to be in the same class. On the other hand, it has been pointed out that enforcing a locality constraint may promote sparsity in the standard SR [114], since only the training samples that are similar to the test sample would be selected for signal reconstruction. To preserve data locality information while reconstructing the multi-source data, we introduce a locality weight on the regularization term. Suppose d_j^i is the locality adaptor for source i that gives the measurement of similarity between $\mathbf{x}^i$ and each atom $\mathbf{a}_j^i$ in its dictionary $\mathbf{A}^i$. d_j^i can

be expressed as

$$d_j^i = \exp\left(\frac{\left\|\mathbf{x}^i - \mathbf{a}_j^i\right\|^2}{\sigma}\right), \tag{4.3}$$

where $\sigma > 0$ is a parameter that determines the decay rate of the weight. From Equation (4.3), it is clear that a smaller d_j^i indicate $\mathbf{x}^i$ is more similar to the atom $\mathbf{a}_j^i$, and vice versa.

Although adding the source-specific locality weight can bring in data locality information for classification, the estimation bias can be large due to the property that ℓ_1/ℓ_2 minimization in general is inconsistent in variable selection (also known as lack of oracle property) [118]. To reduce the estimation bias, for traditional Lasso (ℓ_1 minimization) problem, Zou et al. proposed an adaptive Lasso method, in which the adaptive weights are used to penalize coefficients in the ℓ_1 penalty [118]. A similar idea was also presented by Candes et al. known as reweighted ℓ_1-minimization [119]. Inspired by the reweigthed ℓ_1-minimization or adaptive Lasso, we propose to reweight the locality constrained ℓ_1/ℓ_2 regularization term in an iterative process. The idea behind this is to have better estimation of nonzero coefficients, which can be achieved by allowing a relatively higher penalty for zero coefficients and lower penalty for nonzero coefficients. Thus, in each iteration, the adaptive weight $\widetilde{w}_j^i$ can be computed as inversely proportional to the sparse coefficient in the previous iteration as

$$\widetilde{w}_j^i = \left(\left|s_{j,t+1}^i\right| + \varepsilon\right)^{-1}. \tag{4.4}$$

Further, combining Equation (4.4) and Equation (4.3), we obtain the adaptive locality weight defined as

$$w_j^i = \frac{\widetilde{w}_j^i d_j^i}{\max \widetilde{w}_j^i d_j^i}. \tag{4.5}$$

Note that the weight is adapted for each source, and then combined for a unified sparse representation.

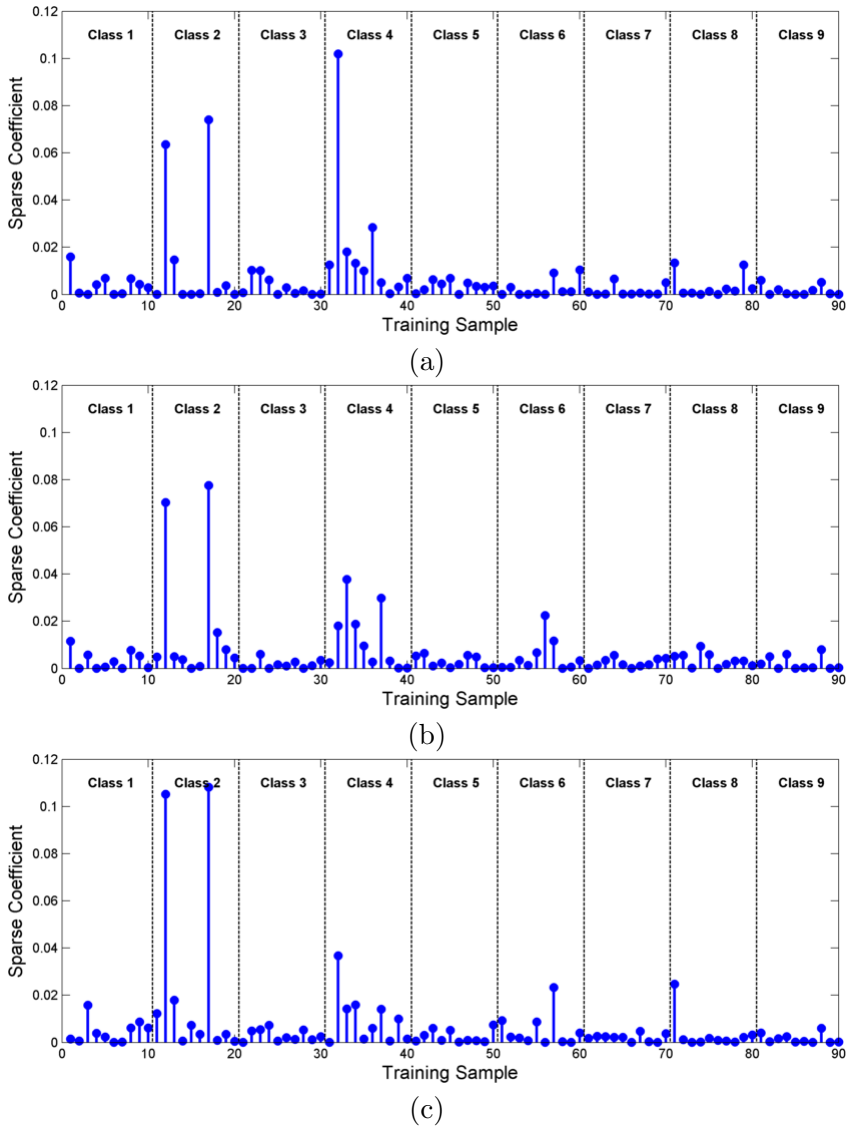


Figure 4.2: An example of sparse coefficients for MTJ-SRC methods with (a) no weight, (b) locality weight, and (c) adaptive locality weight. The class label of the test sample from Class 2 is estimated as (a) Class 4, (b) Class 2, and (c) Class 2.

Figure 4.2 gives an instance of sparse coefficients to demonstrate the benefit of using adaptive locality weight to improve the sparsity and achieve more accurate signal representation. The test sample in the example is selected from the hyperspectral dataset acquired over University of Pavia, and is represented by training samples in the dictionary. The dictionary is composed of 90 samples, which are randomly selected from 9 classes with 10 samples each class. The training samples are indexed class by class, i.e., the first 10 samples are from class 1, and the next 10 samples are from class 2 and so on. We compare the joint sparse coefficients

(defined as $\|\mathbf{S}\|_2$) for MTJ-SRC methods in three conditions — (1) no weight is used, (2) the locality weight is introduced, and (3) the adaptive locality weight is introduced. By applying the MTJ-SRC without any weights on the sparse coefficients, we can see from Figure 4.2 (a) that class 4 has larger coefficients which indicates the test sample is mostly represented by the samples from class 4. However, this would cause classification errors because the test sample is actually selected from class 2. In contrast, when the locality weight is added to constrain the coefficients, we observe that the coefficients from class 2 are promoted while the coefficients from class 4 are suppressed as demonstrated in Figure 4.2 (b) . This is because some atoms in class 2 are more similar to the test sample than others, which leads to smaller weights and larger coefficients on these atoms. The estimated label of the test sample is thus determined as class 2 rather than class 4. Even though the coefficients of class 4 are decreased because of the locality measurement, the difference of coefficients between class 2 and class 4 are not obvious. By implementing the adaptive weight, we can further increase the large coefficients in the desired class and decrease the small coefficients to zero. With the adaptive locality weight, not only is the locality information in the dictionary preserved, but also the estimation bias is reduced.

4.3.2 Optimization Algorithm

Compared to MTJ-SRC, adding the adaptive locality weight, which is pre-computed before the optimization of sparse coefficients $\mathbf{S}$ at each iteration, does not change the convexity of the optimization problem. Since Equation (4.2) is a convex optimization problem, it does not suffer the multiple local minimal issue. However, the ℓ_1/ℓ_2-regularization makes the problem non-smooth and generally considered difficult to solve. Previous work with MTJ-SRC [108], [110] used the Accelerated Proximal Gradient (APG) method to solve the optimization problem in Equation (4.1). In this research, to solve Equation (4.2), we apply the approach of alternating

direction methods of multipliers (ADMM), which has been shown to be more efficient [109], [120]. ADMM, which is based on the variable splitting technique combined with the augmented Lagrangian method, has been successfully applied to a variety of convex but non-smooth problems [121].

To apply the ADMM algorithm on Equation (4.2), the first step is to decouple the variable $\mathbf{S}$ into two convex functions by introducing a new variable $\mathbf{V}$ as

$$\min_{\mathbf{S},\mathbf{V}} L(\mathbf{S}) + \lambda\|\mathbf{V}\|_{1,2} \quad \text{s.t. } \mathbf{W} \odot \mathbf{S} = \mathbf{V}, \tag{4.6}$$

where $L(\mathbf{S}) = \sum_{i=1}^{D} \left\|\mathbf{x}^i - \mathbf{A}^i\mathbf{s}^i\right\|_2^2$, $\mathbf{S} = [\mathbf{s}^1, \mathbf{s}^2, ..., \mathbf{s}^D]$, $\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^D]$, and define $\mathbf{W}^i = \text{diag}(\mathbf{w}^i)$. We then reformulate this constrained problem as an unconstrained counterpart by introducing the augmented Lagrangian function as

$$\begin{aligned} \min_{\mathbf{S},\mathbf{V}} F(\mathbf{S}, \mathbf{V}; \mathbf{B}, \mathbf{W}) &= L(\mathbf{S}) + \lambda\|\mathbf{V}\|_{1,2} \\ &\quad + \langle \mathbf{B}, \mathbf{W} \odot \mathbf{S} - \mathbf{V} \rangle + \tfrac{\beta}{2} \|\mathbf{W} \odot \mathbf{S} - \mathbf{V}\|_F^2 \\ &= L(\mathbf{S}) + \lambda\|\mathbf{V}\|_{1,2} + \tfrac{\beta}{2} \left\|\mathbf{W} \odot \mathbf{S} - \mathbf{V} + \tfrac{1}{\beta}\mathbf{B}\right\|_F^2, \end{aligned} \tag{4.7}$$

where $\mathbf{B}$ is the multiplier of the linear constraint, and β is the positive penalty parameter.

To solve Equation (4.7), with respect to each variable (i.e., $\mathbf{S}, \mathbf{V}, \mathbf{B}, \mathbf{W}$), we keep the other variables fixed, and update the variables sequentially. In each sub-optimization problem, we can derive a closed-form solution. The derivation steps are similar to those in [109], while our solution includes the weight matrices for different sources in each variable update step. At the end of each iteration, we update the weight matrices based on the new solution of $\mathbf{S}$. The iterative process will be executed until an appropriate stopping criterion is met, i.e., the change of objective function is smaller than a pre-defined threshold. The proposed algorithm using ADMM is summarized in Algorithm 1 with the closed-form solutions for each sub-optimization problem. Note that $(\cdot)_+$ is defined as $(x)_+ = \max(x, 0)$, and τ is the stopping threshold which is set as a small positive value.

Algorithm 1: ADMM for ALWMJ-SRC

Input: Test sample $\mathbf{x}^i$, dictionary $\mathbf{A}^i$, for source $i = 1, 2, ..., D$

Initialize: $\mathbf{S}_0, \mathbf{V}_0, \mathbf{B}_0, \mathbf{W}_0$, and choose parameter λ, β

while not converged and $t < T_{max}$ **do**

  Update S: For source i, $\mathbf{s}_{t+1}^i \leftarrow$

$$\left[(\mathbf{A}^i)^T\mathbf{A}^i + \beta(\mathbf{W}_t^i)^T\mathbf{W}_t^i)\right]^{-1}\left[(\mathbf{A}^i)^T\mathbf{x}^i + \beta(\mathbf{W}_t^i)^T(\mathbf{v}_t^i - \mathbf{b}_t^i)\right]$$

  Update V: For row j, $\mathbf{v}_{j,t+1} \leftarrow \left(1 - \frac{\lambda}{\beta}\frac{1}{\|\mathbf{w}_{j,t}\odot s_{j,t+1} + \mathbf{b}_{j,t}/\beta\|_2}\right)_+ (\mathbf{w}_{j,t} \odot s_{j,t+1} + \mathbf{b}_{j,t}/\beta)$

  Update B: For source i, $\mathbf{b}_{t+1}^i \leftarrow \mathbf{b}_t^i + \beta(\mathbf{W}_t^i\mathbf{s}_{t+1}^i - \mathbf{v}_{t+1}^i)$

  Update W: For source i, row j, $w_{j,t+1}^i \leftarrow$

$$\left(\left|s_{j,t+1}^i\right| + \varepsilon\right)^{-1} d_j^i \Big/ \max_j \left\{\left(\left|s_{j,t+1}^i\right| + \varepsilon\right)^{-1} d_j^i\right\}$$

  if $|F(\mathbf{S}_{t+1}, \mathbf{V}_{t+1}; \mathbf{B}_{t+1}, \mathbf{W}_{t+1}) - F(\mathbf{S}_t, \mathbf{V}_t; \mathbf{B}_t, \mathbf{W}_t)| \leq \tau$ **then** break

  $t \leftarrow t + 1$

end while

Return: $\hat{\mathbf{S}}$

Output: $\text{class}(\mathbf{x}) = \underset{c=1,2,...,C}{\arg\min} \sum_{i=1}^{D} \left\|\mathbf{x}^i - \mathbf{A}_c^i\hat{\mathbf{s}}_c^i\right\|_F^2$

4.3.3 Multiscale Decision Fusion Strategy

For EEG multiscale signals, after the joint sparse coefficient $\hat{\mathbf{S}}$ is obtained, we propose a fusion strategy to combine the representation from different scales of features and estimate the class label for the test data. As different scales of features are not equally important for decoding, the goal is to assign them weights based on their discriminative ability. In this work, we adopt a weight motivated by the Fisher's ratio. We first calculate the within-class and

between-class reconstruction error from each sub-dictionary as

$$\begin{aligned} R_w^i &= \sum_{c=1}^{C} \sum_{j \in class\{c\}} \left\| \mathbf{a}_j^i - \mathbf{A}_c^i \hat{\mathbf{s}}_c^i \right\|^2 \\ R_b^i &= \frac{1}{C-1} \sum_{c=1}^{C} \sum_{j \in class\{c\}} \sum_{m \neq c} \left\| \mathbf{a}_j^i - \mathbf{A}_m^i \hat{\mathbf{s}}_m^i \right\|^2. \end{aligned} \tag{4.8}$$

Then the weight can be calculated as the ratio of R_b^i and R_w^i

$$p^i = R_b^i / R_w^i. \tag{4.9}$$

In this way, the scale of features that is more discriminative will be assigned a higher weight in the final decision function. Then the class label of test sample $\mathbf{x}$ will be determined as

$$\text{class}(\mathbf{x}) = \arg\min_{c=1,2,\ldots,C} \sum_{i=1}^{D} p^i \left\| \mathbf{x}^i - \mathbf{A}_c^i \hat{\mathbf{s}}_c^i \right\|_2^2. \tag{4.10}$$

4.3.4 Fusion in the Kernel Space

In the previous section, we present the ALWMJ-SRC algorithm in the original data space. However, the multi-source data may not be linearly separable in the input space. In this case, we extend the ALWMJ-SRC fusion framework to a kernel space.

Given the input data $\mathbf{x}$ in the original data space $\mathbb{R}^d$, define a nonlinear mapping $\Phi(\cdot)$ from the input space to a higher dimensional Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$ as

$$\Phi : \mathbb{R}^d \to \mathcal{H}, \mathbf{x} \to \Phi(\mathbf{x}). \tag{4.11}$$

Then by employing the "kernel trick", a kernel function K is defined as

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle, \tag{4.12}$$

where $\langle \cdot, \cdot \rangle$ is the inner product of two vectors.

As the data mapping and kernel function in the RKHS space are defined as (4.11) and (4.12), the generalized local distance between the test sample $\mathbf{x}^i$ and each atom $\mathbf{a}_j^i$ in the

feature space can be calculated as

$$\varpi_j^i = \left\|\Phi(\mathbf{x}^i) - \Phi(\mathbf{a}_j^i)\right\|^2. \tag{4.13}$$

Using the kernel trick, the generalized local distance can be computed as

$$\varpi_j^i = K(\mathbf{x}^i, \mathbf{x}^i) + K(\mathbf{a}_j^i, \mathbf{a}_j^i) - 2K(\mathbf{x}^i, \mathbf{a}_j^i). \tag{4.14}$$

Then the adaptive locality weight in the kernel space can be expressed similarly to Equation (4.4) and (4.5) as

$$\varphi_j^i = \frac{\widetilde{\varphi}_j^i \varpi_j^i}{\max \widetilde{\varphi}_j^i \varpi_j^i}, \quad \text{and} \quad \widetilde{\varphi}_j^i = \left(\left|p_j^i\right|\right)^{-1}, \tag{4.15}$$

where p_j^i is the sparse coefficient associated with atom $\mathbf{a}_j^i$ in the kernel space.

With the adaptive locality weight defined in the kernel space, the objective function (4.2) can be modified using the mapping function as

$$\hat{\mathbf{P}} = \arg\min_{\mathbf{P}} \frac{1}{2} \sum_{i=1}^{D} \left\|\Phi(\mathbf{x}^i) - \Phi(\mathbf{A}^i)\mathbf{p}^i\right\|_F^2 + \lambda \|\Psi \odot \mathbf{P}\|_{1,2}, \tag{4.16}$$

where Ψ is the weighting matrix, and $\mathbf{P} = [\mathbf{p}^1, \mathbf{p}^2, ..., \mathbf{p}^D]$ is the sparse coefficient matrix in the kernel space. Equation (4.16) can also be written in terms of kernel matrices as

$$\hat{\mathbf{P}} = \sum_{i=1}^{D} \left[\mathbf{Tr}\left((\mathbf{p}^i)^T K_{\mathbf{a},\mathbf{a}}^i \mathbf{p}^i\right) - 2\mathbf{Tr}\left(K_{\mathbf{a},\mathbf{x}}^i \mathbf{p}^i\right)\right] + \lambda \|\Psi \odot \mathbf{P}\|_{1,2}, \tag{4.17}$$

where the kernel matrices are defined as $K_{\mathbf{a},\mathbf{a}}^i(\mathbf{a}_m^i, \mathbf{a}_n^i) = \left\langle \Phi(\mathbf{a}_m^i), \Phi(\mathbf{a}_n^i) \right\rangle$, $K_{\mathbf{a},\mathbf{x}}^i(\mathbf{a}_m^i, \mathbf{x}_n^i) = \left\langle \Phi(\mathbf{a}_m^i), \Phi(\mathbf{x}_n^i) \right\rangle$, and $\mathbf{Tr}(\cdot)$ is the trace of the matrix.

Similar to the linear fusion algorithm, the problem in the kernel space can also be solved by ADMM method. Once the sparse coefficient $\hat{\mathbf{P}}$ is obtained, the class label of the test sample $\mathbf{x}$ can be determined by

$$\text{class}(\mathbf{x}) = \arg\min_{\text{c}=1,2,...,\text{C}} \sum_{\text{i}=1}^{\text{D}} \left\|\Phi(\mathbf{x}^i) - \Phi(\mathbf{A}_c^i)\hat{\mathbf{p}}_c^i\right\|_2^2, \tag{4.18}$$

or in terms of kernel matrices as

$$\text{class}(\mathbf{x}) = \underset{\text{c}=1,2,\ldots,\text{C}}{\arg\min} \sum_{i=1}^{D} \left[\begin{array}{l} \mathbf{Tr}\left(K^i_{\mathbf{x},\mathbf{x}}\right) - 2\mathbf{Tr}\left((\hat{\mathbf{p}}^i_c)^T K^i_{\mathbf{a}_c,\mathbf{x}} \hat{\mathbf{p}}^i_c\right) \\ +\mathbf{Tr}\left((\hat{\mathbf{p}}^i_c)^T K^i_{\mathbf{a}_c,\mathbf{a}_c} \hat{\mathbf{p}}^i_c\right) \end{array} \right], \tag{4.19}$$

where $\hat{\mathbf{p}}^i_c$ is the subset of the sparse coefficient vector $\hat{\mathbf{p}}^i$ associated with class c, and $\mathbf{a}^i_c$ is the atom in the corresponding sub-dictionary.

4.4 Experimental Results

4.4.1 Multi-Source Geospatial Data Fusion

4.4.1.1 Experimental Settings

In the experiments, the raw spectral data and EMAPs from hyperspectral data (or LiDAR pesudo-waveform data) were used as the multi-source input for spectral-spatial feature fusion (or sensor fusion). The efficacy of the proposed fusion algorithm (ALWMJ-SRC) was evaluated and compared with some state-of-the-art algorithms, including the linear SVM, the standard SRC with stacking features, the MTJ-SRC without weight implementation and the recently proposed JCRC-MTL [111]. We also tested the performance of the proposed kernel fusion algorithm (ALWMJ-KSRC), and compared with the corresponding kernel fusion baselines, i.e., the composite-kernel SVM, the KSRC with stacking features, and the MTJ-KSRC. Note that the composite-kernel is defined as a weighted summation kernel to balance data from different sources (e.g., spectral and spatial) as

$$K(\mathbf{x}_i, \mathbf{x}_j) = dK_{spec}(\mathbf{x}^1_i, \mathbf{x}^1_j) + (1-d)\, K_{spa}(\mathbf{x}^2_i, \mathbf{x}^2_j), \tag{4.20}$$

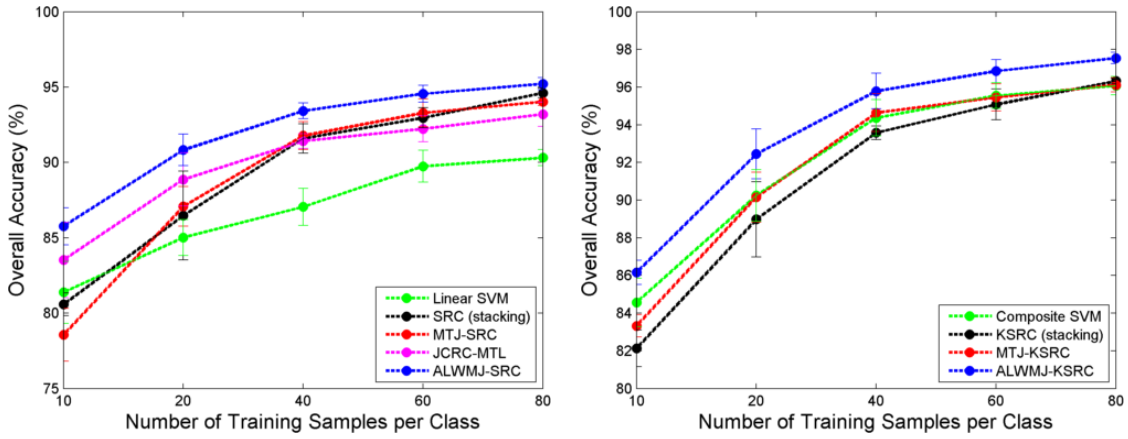
where K_{spec} and K_{spa} are spectral and spatial basis kernels, respectively. d is the weight for spectral kernel — varying d provides different composite kernels. In this work, the kernel weight was optimized in the range [0,1] with a step size of 0.1 through cross-validation, and the optimal value of d was 0.3 for both datasets in the feature fusion. For the sensor fusion, d

was selected as 0.7 for hyperspectral data. We note that the proposed algorithm in the kernel space is conceptually consistent with the composite kernel approach, because both methods make the use of a linear combination of basis kernels from different sources, and thus effectively fuse spectral and spatial information — our method is able to do so by exploiting the sparsity structure in the representation in the composite kernel space.

For SRC-based classification methods, the optimizations (both ℓ_1 and ℓ_1/ℓ_2 problems) were based on ADMM algorithm. The JCRC-MTL approach was implemented using the demo codes provided by the authors, and the parameters were optimized according to the paper [111]. For SVM-based classification methods, we used LIBSVM toolbox to run the experiments. In the kernel fusion experiments, RBF kernels were used with the optimal parameter γ chosen by cross-validation. In this work, the optimal σ is 0.5 for hyperspectral and LiDAR data, and 1.0 for spatial data. Other parameters in ALWMJ-SRC, i.e., regularization parameter λ and penalty parameter β were set as 10^{-2}, 10^{-2}, respectively, through cross-validation. The stopping threshold τ was set as 10^{-5}.

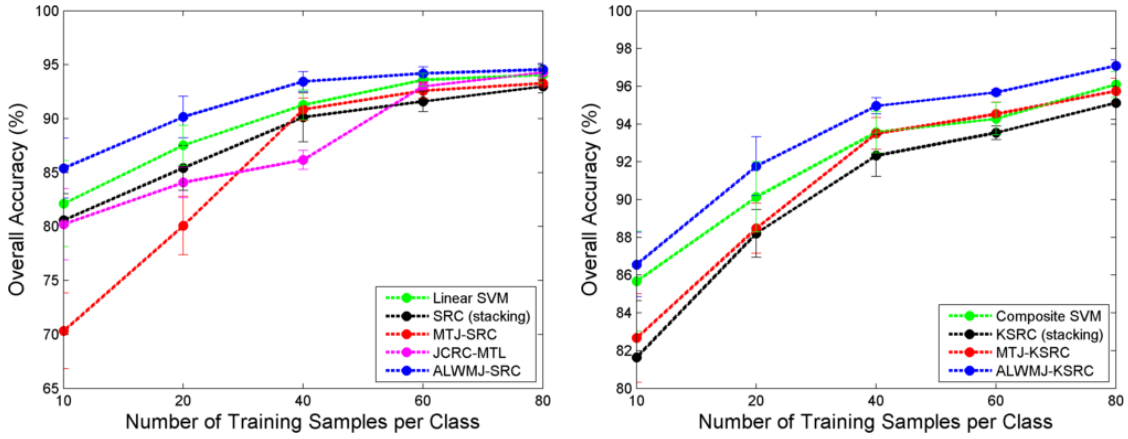
4.4.1.2 Spectral and Spatial Classification

First, we evaluate the proposed algorithm as well as the baseline algorithms as a function of the number of training samples versus overall classification accuracies. We randomly select 10, 20, 40, 60, 80 samples per class to build the dictionary, and the remaining samples are used for testing. Each experiment was repeated for 10 times and the average overall accuracies (OA) and standard deviations are shown in Figure 4.3 and 4.4 for UH and Pavia datasets, respectively.



(a) (b)

Figure 4.3: Overall accuracies achieved on the UH dataset using spectral and spatial features for (a) linear fusion and (b) kernel fusion.



(a) (b)

Figure 4.4: Overall accuracies achieved on the Pavia dataset using spectral and spatial features for (a) linear fusion and (b) kernel fusion.

Considering the proposed ALWMJ-SRC and ALWMJ-KSRC, the results outperform all other methods, including the baseline experiments using linear and composite kernel SVMs. All the kernel fusion methods outperform their linear fusion counterparts by about $2\% - 3\%$ generally, which indicates a non-linear classifier in the kernel space is more suitable to classify the spectral-spatial data. Compared to the MTJ-SRC, which does not use any weight for the ℓ_1/ℓ_2 penalty, by incorporating the locality weight, the ALWMJ-SRC clearly outperforms the MTJ-SRC with higher OA and lower standard deviation, especially when the dictionary size

is small. For example, when the dictionary is composed of 10 training samples per class, the increase of OA by ALWMJ-SRC is 15.08% for Pavia dataset and 7.21% for UH dataset. This indicates when the size of dictionary is not large enough, the locality constraint is important to penalize different coefficients in the ℓ_1/ℓ_2 regularization. The similar atoms in the dictionary are penalized less heavily than those atoms dissimilar to the test sample, and thus result in larger sparse coefficients. However, when the size of dictionary becomes large, the locality information is less important considering the abundant atoms in each class for representation, and thus the benefit of locality weight is less obvious. In addition, by adding an adaptive weight that is inversely proportional to the previous solution, smaller coefficients will be penalized more in the subsequent iteration and encouraged to be more close to zero. This can reduce the estimation bias, and achieve more accurate signal representation.

4.4.1.3 Hyperspectral and LiDAR Classification

Similar experiments were implemented on UH multi-sensor data for sensor fusion. The overall accuracies and standard deviations for hyperspectral and LiDAR fusion are shown in Figure 4.5.

By observing the results of this multi-sensor dataset, we can reach similar conclusions to those of spatial-spectral classification — the proposed ALWMJ-SRC and ALWMJ-KSRC algorithms perform the best among all fusion methods. When a small number of training samples (e.g., 10 per class) are used, the improvement of average classification accuracies by ALWMJ-SRC are substantial, i.e., 12.32%, 9.69%, 5.70%, 3.98% compared to JCRC-MTL, MTJ-SRC, the standard SRC and the linear SVM, respectively. In addition, the kernel fusion methods improve the classification results significantly compared to the linear fusion methods. This means when the data are acquired from different sensors, it would be more efficient to map the heterogeneous data to a high dimensional kernel space for joint classification.

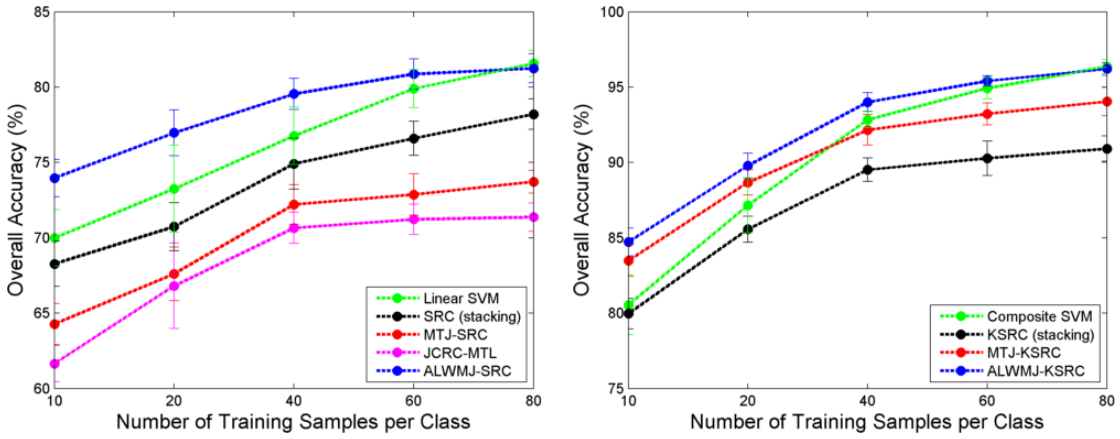


Figure 4.5: Overall accuracies achieved on the UH multi-source dataset using hyperspectral and LiDAR pseudo-waveform data for (a) linear fusion and (b) kernel fusion.

4.4.1.4 Class Specific Analysis

Here we consider the situation where the size of dictionary is small, and illustrate the results using 20 training samples per class for spectral-spatial classification. The class-specific accuracies for different SRC-based fusion algorithms are listed in Table 4.1 and 4.2 for UH and Pavia datasets, respectively. The critical classes are marked **bold** in the table, and are determined based on their similar spectral signatures.

First, we compare the fusion results with the single source classification results using spectral and spatial data separately. For all classes, we can see clear improvements when incorporating both spectral and spatial features, which demonstrates the benefit of data fusion for classification. For most classes, ALWMJ-SRC has the best classification performances. In particular, for those critical classes which are not easily classified by single source spectral or spatial data, the accuracy enhancement by fusion is significant. For example, in the UH dataset, Highway and Railway have similar spectral signatures, and class accuracies are both less than 40% classified by raw spectral data. While after fusion by ALWMJ-SRC the accuracies are 91.24%, 89.48% compared to the baseline results of MTJ-SRC, which are 81.55%, 81.54%

respectively. The reason for accuracy improvement by fusion is that even though these classes are spectrally similar, they are either constructed with different materials or have different shapes and sizes, which makes the EMAPs with morphological and textual features useful to differentiate these classes. Further, by using adaptive locality weight to constrain the coefficients, the test sample is only represented by its nearest neighbors adaptively, and thus the classification error is reduced. Similar results can also be found in other critical classes from Pavia dataset, e.g., Gravel vs Bricks, Asphalt vs Bitumen.

To visualize the classification performance, we provide the classification maps generated using 80 training samples per class as shown in Figure 4.6 for Pavia dataset. By comparing the maps, we can see a much smoother and more accurate map for ALWMJ-SRC than other approaches.

4.4.1.5 Computational Complexity

By using the ADMM algorithm to solve the proposed objective function, the variables are updated alternatively, and the optimization problem can be divided into sub-optimization problems. The solution for each sub-optimization problem is in closed-form and thus the calculation is efficient. Compared to MTJ-SRC, ALWMJ-SRC adds a weight update step, which however only involves matrix computations. Therefore the time complexity is the same for MTJ-SRC and ALWMJ-SRC.

Table 4.1: Class-specific accuracies and overall accuracies (%) for the University of Houston dataset.

Class	Single Source		Linear Fusion				Kernel Fusion		
Index	spectral	spatial	SVM	MTJ	JCRC	ALWMJ	SVM	MTJ	ALWMJ
1	85.09	91.90	97.02	95.21	92.83	98.32	98.08	97.28	**98.36**
2	91.72	90.52	91.36	95.44	93.58	98.05	98.45	97.87	**98.59**
3	99.80	99.61	97.21	97.28	98.44	98.14	**100**	98.88	**100**
4	98.12	91.31	98.25	98.05	98.72	**98.75**	97.79	97.15	98.04
5	65.53	82.31	91.56	88.17	**100**	91.24	91.65	92.24	93.15
6	100	98.78	98.56	97.20	72.97	**100**	**100**	98.98	**100**
7	67.39	88.76	80.59	**94.22**	69.18	87.58	87.62	89.28	91.11
8	51.21	54.08	61.23	77.45	70.58	92.40	80.25	81.55	**92.51**
9	23.53	48.38	65.27	66.68	69.02	86.21	81.46	78.77	**86.87**
10	36.91	70.65	86.76	80.48	**95.02**	90.44	89.78	91.08	94.66
11	36.52	60.63	85.24	80.46	88.56	88.23	87.12	80.25	**90.56**
12	24.85	46.68	72.68	75.22	87.40	83.16	82.78	83.40	**90.74**
13	50.89	67.98	71.26	85.32	87.17	89.56	80.24	87.14	**89.92**
14	97.05	94.61	94.56	96.02	**100**	94.48	95.28	95.39	98.58
15	85.45	99.41	98.08	97.85	**99.57**	97.90	97.02	97.32	98.14
OA	66.19	78.96	85.02	87.08	88.87	90.83	90.25	90.16	**92.45**

Table 4.2: Class-specific accuracies and overall accuracies (%) for the University of Pavia dataset

Class	Single Source		Linear Fusion				Kernel Fusion		
Index	spectral	spatial	SVM	MTJ	JCRC	ALWMJ	SVM	MTJ	ALWMJ
1	27.76	72.66	84.14	76.60	72.55	92.38	90.52	90.32	**94.13**
2	53.10	69.44	85.12	75.56	87.14	89.42	90.21	85.24	**91.21**
3	41.70	82.36	71.29	80.02	88.38	91.86	90.24	85.13	**94.74**
4	83.55	74.36	85.08	82.14	62.60	81.30	**88.55**	86.69	87.25
5	98.10	97.42	94.28	96.57	98.90	**99.08**	98.42	98.26	98.68
6	46.28	79.08	86.14	84.26	87.89	**94.43**	90.53	90.04	94.22
7	48.93	86.31	85.65	86.58	85.26	90.58	90.02	81.57	**91.65**
8	23.61	67.97	84.34	65.24	67.14	87.06	88.38	87.46	**90.18**
9	98.25	87.82	97.81	91.78	90.14	97.42	98.05	**99.82**	99.28
OA	59.29	78.78	87.54	80.09	84.09	90.16	90.14	88.48	**91.77**

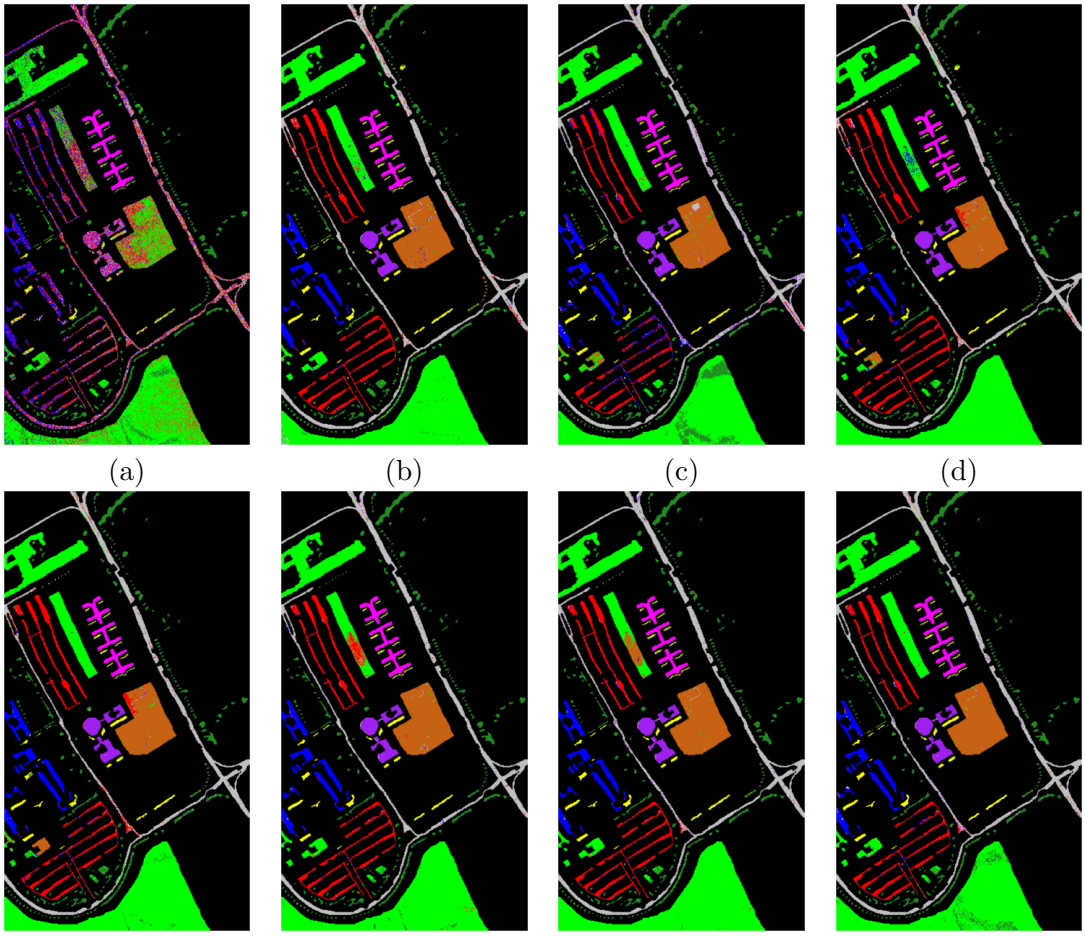


Figure 4.6: Classification maps of Pavia dataset using (a) SRC-spectral (b) SRC-spatial (c) Linear-SVM (d) MTJ-SRC (e) ALWMJ-SRC (f) Composite-SVM (g) MTJ-KSRC (h) ALWMJ-KSRC.

4.4.2 Gait Phase Decoding from EEG signals

Over the past years, gait phase detection has been extensively studied using foot pressure, kinematic and electromyography (EMG) data [122–129]. However, gait phase decoding through Brain Computer Interface (BCI) from electroencephalography (EEG) signals has just emerged as a new research problem in recent years. BCI systems have been successfully used to help patients that lose motor ability (e.g., stroke, paraplegia, and quadriplegia) to relearn and recover walking ability. Previous research [130] has shown the feasibility of decoding human gait into two successive phases of stance and swing using time-domain features from EEG

signals in low delta band (0.1-2 Hz). Classification was performed using a Linear Discriminant Analysis (LDA) classifier with selected EEG channels and achieved high decoding accuracy. However, in this study, only the features from delta band were used for decoding, which may neglect important information from other frequency bands. In fact, many studies have demonstrated that spectral power changes in different frequency bands during different phases of a gait cycle. Severens et al. found two types of modulations in spectral power [131] during walking, i.e., inter-stride and intra-stride modulations. First, an overall power decrease was observed in the mu and beta bands along the whole period during walking. Furthermore, gait event related spectral perturbations (ERSPs) in mu, beta and gamma band were found that are coupled to the gait cycle. In addition, Gwin et al. found that alpha- and beta-band spectral power increases during the heel strike and approximately in the middle of the double support phases, and the intra-stride high-gamma spectral power changes in anterior cingulate, posterior parietal, and sensorimotor cortex [132]. Motivated by the above findings, in [133], walking and no-working states in both actual walking and imaginary walking are classified using ERSPs from different frequency bands, i.e., mu (8-12 Hz), beta (12-25 Hz) and mu-beta (8-25Hz). The experimental results consistently show that the combined mu and beta bands were complementary and gave better performances than taking them separately. Previous research results motivate our hypothesis that it would be promising to systematically integrate EEG features from different frequency bands to improve the decoding accuracy of various phases of gait.

Different from previous works that only extracted features from limited frequency bands [130, 131], we build an over-complete dictionary covering different frequency bands from low

delta to gamma. We hypothesize that features from different frequency bands are complementary and it would be effective to combine their information to enhance the decoding performance. To effectively fuse different scales of features, ALWMJ-SRC algorithm is validated on multiple sessions of EEG data recorded from five healthy subjects walking on a treadmill in a BCI virtual reality application.

4.4.2.1 Data Recording and Gait Segmentation

Five healthy subjects with no history of neurological disease or gait pathology participated in this study for four sessions after each of them submitted a consent form. The experimental protocol was approved by the Institutional Review Board at the University of Houston, USA. At the beginning of each trial, the subject was instructed to stand still for 2 minutes on a treadmill while minimizing eye blinks. The treadmill was then slowly sped up to 1 mph by an experimenter and the subject kept this walking speed for 10 minutes.

Multichannel EEG (64 channels) was recorded by combining two 32-channel amplifiers (actiCap system, Brain Products GmbH, Germany). The electrodes were placed and labeled in accordance with the extended 10-20 international system. EEG data were referenced to FCz channel and sampled at 100 Hz. Lower limb joint angles (hip, knee, and ankle) were recorded by goniometer sensors (SG150 & SG110/A Gonio, Biometrics Ltd, UK) at 100 Hz. Kinematic data (accelerations) were sampled at 128 Hz by using three wireless OPAL sensors (OPAL, APDM Inc., Portland, OR) placed at the forehead, left and right heel of the subject. Kinematic data of the heel would be used to segment all the data into gait cycles. Recording of EEG data, goniometer data, and OPAL data were synchronized using our custom C++ program.

Gait segmentation was accomplished by identifying heel strike and toe off events of both the right and left leg. These events were identified using acceleration data from the Opal IMUs

including the acceleration in the directions parallel to the foot's sagittal plane of motion — anterior/posterior and proximal/distal. Specific parameters were used to identify peaks based on individual metrics tuned for each subject including minimum peak thresholds and minimum spacing between peaks. The timing of these events was compared to synchronized joint angle positions measured with goniometers as shown in Figure 4.7, closely matching published data regarding the relative location of heel strike and toe off in the gait cycle as determined with alternative instrumentation [134], [135]. A secondary validation method was to compare the number of gait cycles in which the right and left leg events occurred in the correct sequence to the number of gait cycles counted by large knee angle peaks. The segmentation accuracy was over 99% for all subjects. The identified timing of gait events were then used as class labels for supervised learning in the decoding experiments.

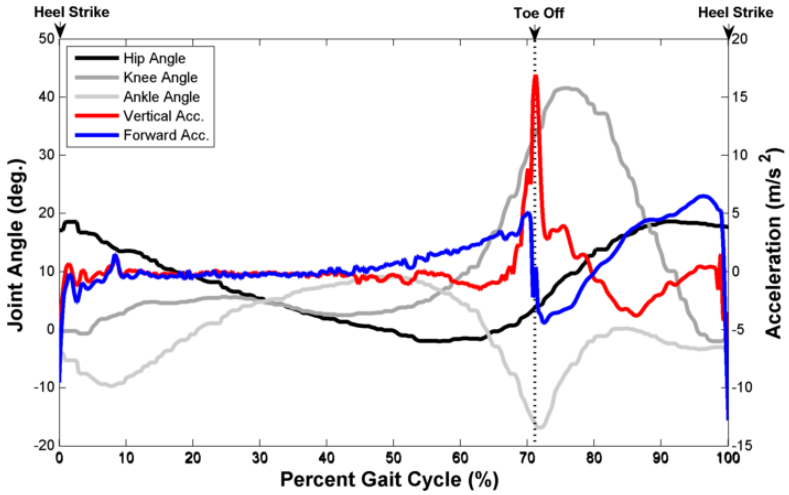


Figure 4.7: Gait segmentation determined by acceleration data with comparison to the joint angle positions in a single gait cycle of the right leg.

4.4.2.2 Feature Extraction and Selection

Before building the RDWT-based dictionary and decoding the gait phases using ALWMJ-SRC, the recorded EEG data were first preprocessed. The EEG signals were high pass filtered at 0.1 Hz with a zero-phase 2nd order Butterworth filter, and then standardized by channel by subtracting the mean and dividing by the standard deviation. The preprocessed EEG data

synchronized with the segmented gait events are shown in Figure 4.8 for selected channels.

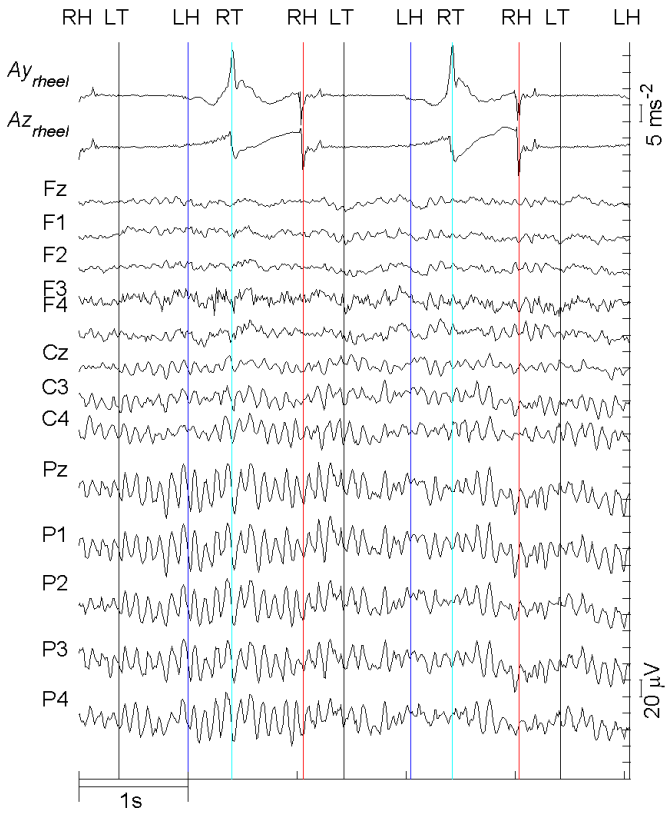


Figure 4.8: Preprocessed EEG data synchronized with four gait events. RH, LT, LH, RT represents four classes for decoding, i.e., right heel strike, left toe off, left heel strike and right toe off.

From the preprocessed EEG data, the detail and approximation wavelet coefficients were calculated from the RDWT decomposition and used to build a multiscale dictionary. The length of wavelet coefficients for all scales are the same as the original signal length. Selection of the appropriate wavelet type and the number of decomposition levels is very important for signal analysis and decoding. The Daubechies family of wavelets has been shown to have advantages over other types of wavelet for multiple EEG signal classification problems [50], [51], [136]. We experimentally determined that Daubechies wavelet 4 (db4) wavelet, which is effective to detect small changes of the EEG signals, is suitable for this research. The selection

of the decomposition levels depends on the sampling frequency and the frequency bands of interest. In this study, since the EEG signals were sampled at 100 Hz, the highest frequency that the signal could contain would be 50 Hz based on the Nyquist theorem. We decomposed the signal into five levels, so each level of coefficients would encompass the information from the frequency bands we are interested in. Frequency bands corresponding to five decomposition levels are listed in Table 4.3. The final dictionary is thus comprised of six sub-dictionaries respect to the decomposition levels — Approximation (A), Detail 1-5 (D5, D4, D3, D2 and D1).

Table 4.3: RDWT decomposition scales and the corresponding frequency bands.

Level	Frequency (Hz)	Band Name
D1	25 - 50	gamma
D2	12.5 - 25	beta
D3	6.25 - 12.5	alpha
D4	3.12 - 6.25	theta
D5	1.06 - 3.12	delta
A	0.1 - 1.06	low delta

The decomposed multiscale EEG signals were then segmented into pre-movement epochs of 200 ms duration using a rectangular window of size 20 based on the gait segmentation results. Each epoch is consisted of data from 50 ms before a gait event happened. The epochs from the same scale was then concatenated into a single time series containing different gait events in all gait cycles. The concatenated EEG epochs were then formed as a four-class classification problem for each trail, in which each epoch is labeled as 1 to 4 corresponding to right heel strike (RH), left toe off (LT), left heel strike (LH) and right toe off (RT), respectively.

To build the dictionary for decoding, each epoch was transformed to a feature vector which represented a data point in the feature space. The feature vector was formed by concatenating 20 lags for all 64 channels into a single vector of length 20×64. Therefore, each trail contains

six scales of features, and each scale is a feature matrix of size $N \times 1280$, where N is the number of epochs in a trail. To avoid the overfitting problem in classification, we further reduced the dimensionality of the feature matrix. Many techniques have been proposed for dimensionality reduction. It is commonly known that not all EEG channels contain useful information for decoding. So in this research, we reduced the number of features by selecting the best combination of channels based on the Differential Evolution Feature Selection (DEFS) approach [137]. DEFS employs a variation of differential evolution (DE) as the search engine, and we developed the fitness function using the classification error rate based on the SRC classifier. The population size and the number of iterations were both set as 50. We changed the number of channels used for decoding from 1 to 63 and ran the DEFS for 10 times to find the best sets with the highest channel occurrence for each scale of features and for each subject. Following this, the sub-dictionaries was comprised of the features from the selected channels, and the final dictionary was a combination of all six sub-dictionaries.

By changing the number of channels selected from 1 to 63, we experimentally determined that the optimal number of channels is between 18 and 25 for all subjects. By referring to "optimal", we imply that the decoding accuracy reaches its peak and either drops or saturates afterwards. Following this, we fixed the number of optimal selected channels for each trial and ran the DEFS for 10 times. 100 samples were randomly selected from each class to train the SRC classifier. We calculated the number of times that each channel was selected within 10 runs, in which case the maximum selected time is 10 and the minimum is 0. The channels with the highest occurrences implies that the corresponding regions may contain the most discriminative information for decoding. We ran the DEFS on all 20 trails, and found that the channel selection results were different for different trails. We compare the results from the best trial with the highest decoding accuracy and the worst trial with the lowest decoding

accuracy in Figure 4.9.

From the results in the best trial, we observed that multiple central channels (e.g., Cz, C1, C2, C4, Cp1) were selected more times than the others, which indicates information that is useful for decoding mainly concentrated on the midline central area of the scalp. In addition, for different scales of signals, the selected important areas are complementary in the best trial. In contrast, when the important information are not centralized in the midline central area and the important areas are identical and limited across different frequency bands, the decoding accuracy is much lower. For all the trials, although the selected important channels are different, we found the most commonly selected channels are centered on the midline central area, and among which Cz is the most selected channel.

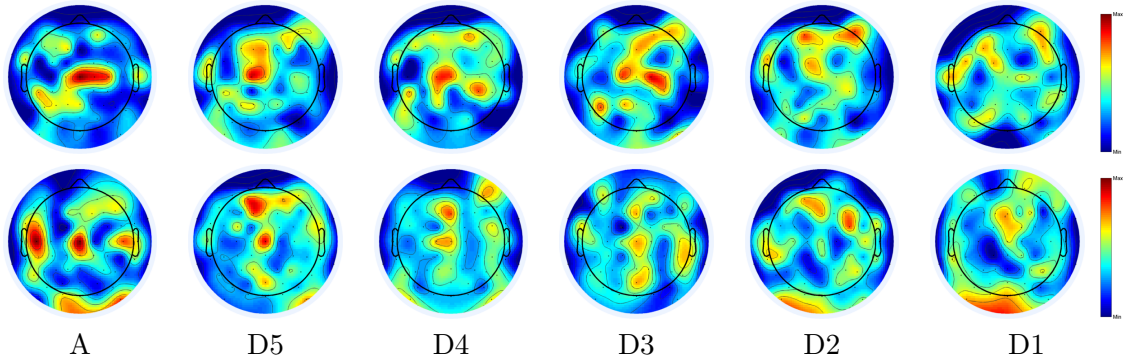


Figure 4.9: Scalp maps showing the occurrence of selected channels in ten different runs for different scales of features. The first row shows the results in the best trail, and the second row shows the results in the worst trail.

4.4.2.3 Classification and Fusion

After feature extraction and selection, for each trail, we built a dictionary based on the selected features for different scales. To simulate a real-time decoding environment, in each trail, the first half of the labeled samples (epochs) were used to build the dictionary for training, and the remaining half were used for testing and evaluation. The testing process was repeated 10 times for each trial of data and the metric for evaluation is the average overall accuracy,

where the overall accuracy is defined as the number of correctly classified samples divide by the total number of samples in the experiment. In all the experiments, the parameters, i.e., regularization parameter and penalty parameter, were set as 10^{-2}, 10^{-2}, respectively, through cross-validation. The stopping threshold was set as 10^{-5}.

1. Comparison of Single Scale and Multiscale Classification

First, we compare the classification results using each separate scale of features to the fusion results using all scales of features. The results for each subject are averaged over all trials are shown in Figure 4.10.

As the scale of features goes from Approximation to Detail (the frequency increases), it is observed that the decoding accuracy decreases for all subjects. The overall accuracy is around 55% - 70% for the approximation scale which corresponds to the low delta band. This result indicates that the low delta band EEG signals carries the most discriminative information for different gait phases. At scale D5, the accuracies are 15% - 20% lower than the approximation scale. As the scale becomes finer, the individual accuracies at the scale continue to decrease, implying that the higher frequencies contain less discriminative information than lower frequencies for gait phases decoding. However, we note that even the most fine-scale dictionary in $25 - 50$ Hz range has a decoding accuracies over 30% which are higher than the chance level (25%). This indicates that there is some discriminative information in higher frequency bands.

The results of fusion further validates our hypothesis that different scales of EEG features can be used to provide complementary information for decoding. The average decoding accuracies increase by around 6.5% compared to those where only the approximation scale is used.

2. Comparison of Weighted and Non-weighed Fusion Algorithm

To demonstrate the effectiveness of the proposed weighted algorithm for combining different scales of features for decoding, we compare the weighted fusion algorithm (ALWMJ-SRC) with the non-weighted fusion algorithm (MTJSRC). The average decoding accuracies and standard deviations are shown in Figure 4.11.

First, we note that both fusion algorithms are effective at improving the decoding accuracies compared to the single scale features. For MTJSRC, which does not consider the different importance of different scales of features for decoding, the improvement of decoding accuracies is around 2% compared to the approximation scale. In contrast, by implementing an adaptive weight considering the locality, the discriminative ability, and the stability of the estimation, the decoding accuracies are further improved by 5%. The results demonstrate that the proposed adaptive weight multiscale sparse representation algorithm is more effective at combining different scales of features while considering the difference of each scale for improving the decoding accuracy.

3. Class-wise Accuracy

Next, we give some insight on the classification results regarding different gait phases. We show the confusion matrices in terms of class-wise decoding accuracies and misclassification rate. Generally, four classes have similar decoding accuracies. The ALWMJ-SRC algorithm does not favor one class over the others.

It is observed that the off-diagonal values of the confusion matrix are generally lower than others, implying that misclassification rates between heel strike and toe off events for the same foot are generally lower than those for different feet. In particular, LT and LH are observed as the least confused class pairs among all subjects, while RH and LH have a relatively high misclassification rate. This observation indicates that the heel strike and toe off have rather

distinct features to be differentiate, while the same event (heel strike or toe off) has similar features and is difficult to classify between different feet.

In addition, we show the results of simulated real-time decoding of gait phases for subject 5 in Figure 4.13. The classifier was trained on the first half of the recorded EEG trial and tested on the consecutive 106 seconds of the trial to simulate the real-time decoding decisions. The decoding accuracy for this period was 75 %. The results also indicate that the heel strike and toe off are the most misclassified classes.

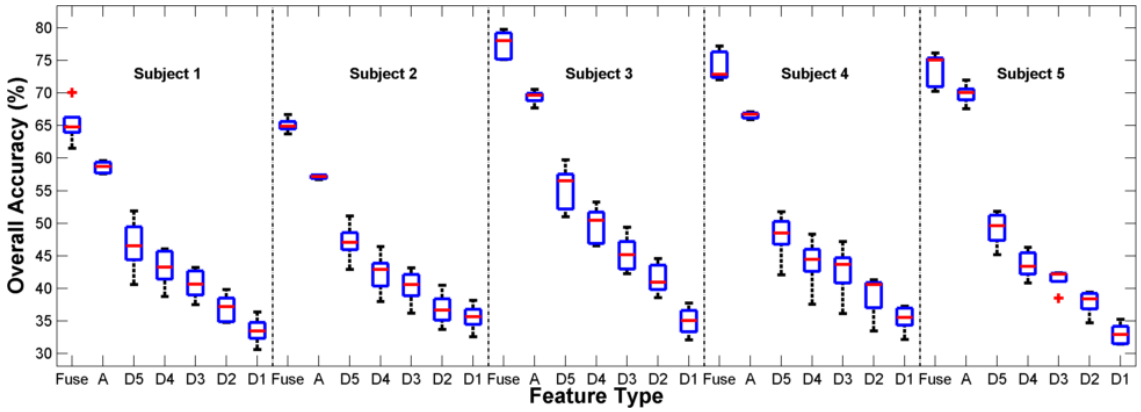


Figure 4.10: A comparison of classification overall accuracies using different scales of features and a combination of all features. Fuse represents the fusion results, and A, D1-D5 represents the results by approximation and five detail scales.

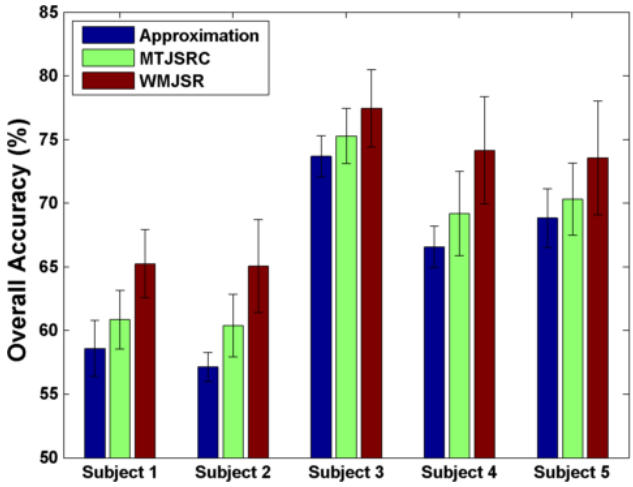


Figure 4.11: Average decoding accuracies (%) and standard deviations (%) for different methods and subjects.

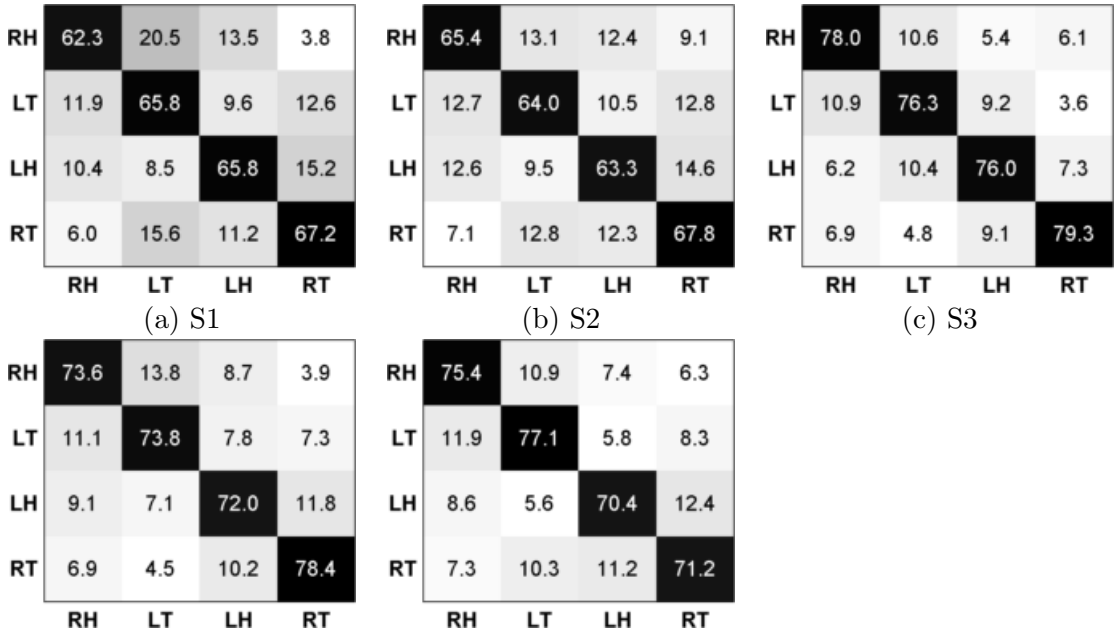


Figure 4.12: Confusion matrices (%) for subject 1 to 5. RH, LT, LH, RT represents four classes for decoding, i.e., right heel strike, left toe off, left heel strike and right toe off.

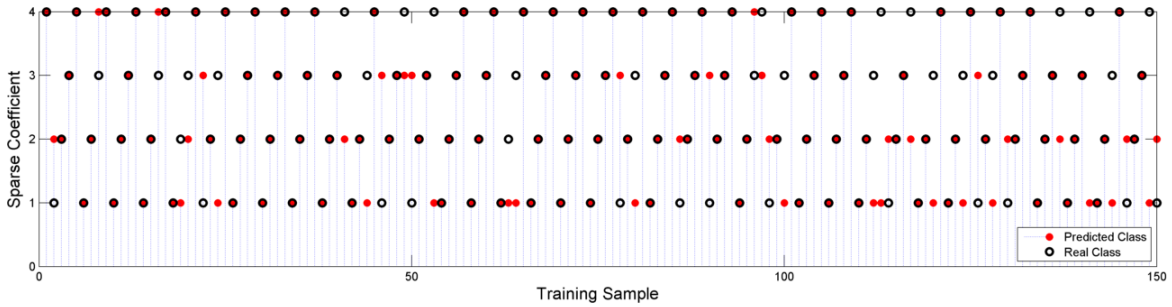


Figure 4.13: Simulation of real-time decoding of gait phases for one subject. The figure contains a time series of simulated real-time classification decisions from the consecutive 106 seconds of the trial.

4.5 Summary

In this chapter, the ALWMJ-SRC algorithm is proposed for multi-source data fusion, which is based on multi-task joint sparse representation framework and incorporates an adaptive locality constrained weight. The proposed algorithm is designed to overcome the limitation that differences between atoms and sources are ignored in previous works. By adding the adaptive locality constraint weight, it considers the locality information between the test sample and

the atoms in the dictionary, and adapts it by penalizing different coefficients for better signal reconstruction. The efficacy of the proposed algorithm is validated on two different applications — multi-source geospatial data classification and gait phases decoding from multiscale EEG signals.

For multi-source geospatial data classification, the overall classification accuracies of ALWMJ-SRC exhibit consistently better performance than state-of-the-art algorithms for both sensor and feature fusion, especially when the dictionary size is small. The class-specific accuracies demonstrate that ALWMJ-SRC is particularly efficient for discriminating critical classes with similar spectral signatures. For gait phases decoding, EEG signals were first decomposed into multiple scales of features based on RDWT decomposition to build an overcomplete dictionary, following which a joint sparse representation framework was applied to fuse different scales of features for compact representation. By selecting the important channels for decoding, we determined that useful information for decoding the gait phases mainly concentrated on the midline central area of the scalp. The experimental results using independent scale of features indicate that the most discriminative information for gait phases are contained in the approximation scale of features (i.e., low frequency). For the higher frequency sub-bands, the decoding accuracies decreased compared to the low frequency (coarse) sub-bands, however, they clearly possess discriminative information useful for decoding. Results with the proposed approach for "optimal" fusion using all scales of features validate our hypothesis that different scales of EEG features can be used in a complementary manner for highly accurate decoding.

Chapter 5

Summary and Conclusions

Incorporating disparate features from multiple sources can provide valuable diverse information for data analysis in many applications. In this dissertation, we develop and demonstrate the value of multi-source information fusion techniques for robust classification. The proposed algorithms are categorized under two broad categories — a mixture of kernels approach and a joint sparse representation approach. The joint sparse representation in the kernel space makes the use of a linear combination of base kernels which expends the mixture of kernel ideas into a sparse representation framework. In this chapter, we first summarize the main contributions of the dissertation and then give the possible directions for future work.

5.1 Dissertation Contribution

The key contributions of this dissertation are summarized as follows:

1. Locality Preserving Composite Kernel Feature Extraction

A composite-kernel-based feature extraction algorithm (CKLFDA) is proposed to efficiently fuse the multi-source data in a lower dimensional subspace, which results in features derived from the multiple sources that possess optimal class separability. To demonstrate the benefits of CKLFDA, we conduct experiments on both multi-feature and multi-sensor remote sensing data. The experimental results validate the hypothesis that CKLFDA serves as a very effective and robust feature extraction tool for various classifiers, such as Gaussian ML and MLR — we note that the composite kernel projection results in a feature space wherein data are linearly separable, making it feasible to utilize a simple classifier such as ML or MLR at the backend.

CKLFDA-MLR outperforms all the other traditional methods in terms of overall classification accuracy while with similar computational cost.

2. Multiple Kernel Based Region Importance Learning

MKL has the advantage of learning the classifier and the optimal kernel weights simultaneously. In this dissertation, the MKL algorithm is successfully applied to infer the relative importance of different scalp brain regions while decoding user's gait movement intention from EEG signals. The experimental results demonstrate that the frontal/frontal-central regions are the most important regions for movement decoding, which is consistent with the brain regions believed to be involved in the control of lower-limb movements. In addition, from the longitudinal experiment results, we conclude that the decoding accuracy generally increases while the user learns to control the exoskeleton for movement and the important regions get increasing weights along sessions for decoding. The results demonstrate the cortical plasticity triggered by the BMI use.

3. Ensemble Multiple Kernel Active Learning

The ensemble multiple kernel active learning (EnsembleMKL-AL) framework provides a novel approach to exploit multi-sensor, multi-feature remote sensing datasets with limited number of labeled samples for image classification. The experiments validate the efficacy of the proposed framework and provide the following conclusions — (a) MKL is a more effective and appropriate classifier for multi-source AL compared to the standard SVM classifier; (b) Ensemble classifiers improve the performance of traditional AL substantially for the multi-source data. The proposed EnsembleMKL-AL system greatly outperforms the SimpleMKL-AL approach in terms of overall and class-specific accuracies. The computational time for EnsembleMKL-MD-LOP is slightly higher than SimpleMKL-Ms, but is much more efficient than the SVM-MS approach.

4. Locality Driven Joint Sparse Representation

A locality (as measured in the feature space) driven joint sparse representation model is proposed for effective multi-source data fusion. The proposed algorithm, built on the notion of multi-task joint sparse representation, incorporates an adaptive locality weight to overcome the key shortcomings (*e.g., uniform weights, unstable estimation of coefficients*) in prior related work. By adding the adaptive locality weight, we not only take into consideration the locality information between the test sample and the dictionary, but also adaptively penalize the coefficients to reduce estimation bias. This algorithm is also "kernelized" in the dissertation. The proposed algorithm is validated through feature and sensor fusion of multi-source geospatial data. The efficacy of the proposed algorithm is validated via experiments for two fusion scenarios — spectral-spatial classification and hyperspectral-LiDAR sensor fusion. The overall classification accuracies of ALWMJ-SRC exhibit consistently better performance than the baseline algorithms, especially when the dictionary size is small. The class-specific accuracies demonstrate that ALWMJ-SRC is particularly efficient for discriminating critical classes with similar spectral signatures.

5. Multiscale Joint Sparse Representation for Gait Phases Decoding

As a novel application of data fusion, we apply the proposed weighted joint sparse representation algorithm to analyze the gait patterns from EEG signals for brain machine interface. EEG signals were first decomposed into multiple scales of features based on RDWT decomposition to build an overcomplete dictionary, following which a joint sparse representation framework was proposed to fuse different scales of features for compact representation for decoding. The experimental results confirm that the important information for decoding the gait phase primarily centralize in the midline central area of the scalp, and that low frequency

features carry the most discriminative information. Although the higher frequency dictionaries are less effective for classification by themselves, they provide complementary information and improve the decoding performance when optimally combined with dictionaries from other frequency regions through the proposed data fusion strategy.

5.2 Future Work

Based on the results and findings of the current work, we give some promising directions for the future research.

1. Multi-source Active Learning

Up to now, the majority of AL algorithms are developed for single source classification and based on a certain type of classifier, such as SVM. In this dissertation, we provide a new way to exploit multi-sensor, multi-feature remote sensing datasets through ensemble active learning for image classification. A future direction for AL is to develop a multi-source AL framework based on the sparse representation based classification.

In [138], the authors developed an AL framework based on convex programming which can be used on SRC. The principles of sample section are classifier uncertainty and sample diversity. In [139, 140], the construction errors and sparse representation based classification errors are used as query criteria in the AL. An interesting research direction for multi-source AL is to build a query strategy selecting important samples given consideration to both uncertainty of each source and disagreement over different sources based on the MTJ-SRC algorithm.

2. Combine Data Fusion with Domain Adaptation

In the current work, a key assumption is that the training and test data are drawn from the same feature space and have the same distribution. However, in real-world applications, the assumption is sometimes not satisfied. For remote sensing data analysis, take the hyperspectral imagery as an example, there usually exists some shift in the spectral distribution due to

different illumination or atmospheric conditions between disjoint areas. To solve this problem, domain adaptation or transfer learning can be incorporated into the fusion framework to overcome the influence of distribution bias between the source and target domains. For BMI applications, insofar as the problem of effectively utilizing training samples and existing models from one subject towards effectively decoding other subjects, the field of transfer learning is also pertinent. In this sense, methods that combine domain transfer with efficient fusion algorithms (e.g., multiple kernel learning) need be explored for information fusion in the future research.

3. Combine Joint Sparse Representation with Manifold Learning

This dissertation has demonstrate the feasibility of utilizing joint sparse representation for robust multi-source data classification, however, the data still reside in the high dimensional space which may cause inefficient computation. Dimensionality reduction via manifold learning offers an elegant representation of data whereby the high dimensional feature space is parameterized by a lower dimensional space where the data resides. A further improvement for multi-source data classification could incorporate manifold learning and jointly optimize it with the sparse representation. We may make use of the Laplacian graph embedding method and add it as a regularization term in the SRC objective function to reduce the dimension of the dictionary.

References

[1] C. Pohl and J. L. Van Genderen, "Review article multisensor image fusion in remote sensing: concepts, methods and applications," *International journal of remote sensing*, vol. 19, no. 5, pp. 823–854, 1998.

[2] R. C. Luo, C.-C. Yih, and K. L. Su, "Multisensor fusion and integration: approaches, applications, and future research directions," *Sensors Journal, IEEE*, vol. 2, no. 2, pp. 107–119, Apr. 2002.

[3] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 24–38, Jan. 1997.

[4] N. H. Nguyen, N. M. Nasrabadi, and T. D. Tran, "Robust multi-sensor classification via joint sparse representation," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on.* IEEE, 2011, pp. 1–8.

[5] G. Shaw and D. Manolakis, "Signal processing for hyperspectral image exploitation," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 12–16, Jan. 2002.

[6] X. Huang and L. Zhang, "A comparative study of spatial approaches for urban mapping using hyperspectral ROSIS images over Pavia City, northern Italy," *International Journal of Remote Sensing*, vol. 30, no. 12, pp. 3205–3221, Jun. 2009.

[7] E. M. DuPont, D. Chambers, J. Alexander, and K. Alley, "A spatial-spectral classification approach of multispectral data for ground perspective materials," in *2011 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2011, pp. 3125–3129.

[8] C. Hladik and M. Alber, "Classification of salt marsh vegetation using edaphic and remote sensing-derived variables," *Estuarine, Coastal and Shelf Science*, vol. 141, pp. 47 – 57, Mar. 2014.

[9] M. A. Wulder, J. C. White, R. F. Nelson, E. Næsset, H. O. Ørka, N. C. Coops, T. Hilker, C. W. Bater, and T. Gobakken, "Lidar sampling for large-area forest characterization: A review," *Remote Sensing of Environment*, vol. 121, pp. 196–209, Jun. 2012.

[10] M. Dalponte, L. Bruzzone, and D. Gianelle, "Fusion of hyperspectral and LIDAR remote sensing data for classification of complex forest areas," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 5, pp. 1416–1427, May 2008.

[11] M. Shimoni, G. Tolt, C. Perneel, and J. Ahlberg, "Detection of vehicles in shadow areas using combined hyperspectral and lidar data," in *2011 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2011, pp. 4427–4430.

[12] A. H. S. Solberg, T. Taxt, and A. K. Jain, "A Markov random field model for classification of multisource satellite imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 34, no. 1, pp. 100–113, Jan. 1996.

[13] B. C. Tso and P. M. Mather, "Classification of multisource remote sensing imagery using a genetic algorithm and Markov random fields," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 3, pp. 1255–1260, May 1999.

[14] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 3, pp. 809–823, Mar. 2012.

[15] P. Ghamisi, J. A. Benediktsson, and M. O. Ulfarsson, "Spectral–spatial classification of

hyperspectral images based on hidden Markov random fields," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 2565–2574, May 2014.

[16] M. Pedergnana, P. R. Marpu, M. Dalla Mura, J. A. Benediktsson, and L. Bruzzone, "Fusion of hyperspectral and lidar data using morphological attribute profiles," in *SPIE Remote Sensing*, 2011, pp. 81 801G–81 801G.

[17] D. Tuia, F. Ratle, A. Pozdnoukhov, and G. Camps-Valls, "Multisource composite kernels for urban-image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 1, pp. 88–92, Jan. 2010.

[18] G. Camps-Valls, L. Gomez-Chova, J. Muñoz-Marí, J. Vila-Francés, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 1, pp. 93–97, Jan. 2006.

[19] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, Jun. 2002.

[20] E. C. Leuthardt, G. Schalk, J. R. Wolpaw, J. G. Ojemann, and D. W. Moran, "A brain–computer interface using electrocorticographic signals in humans," *Journal of neural engineering*, vol. 1, no. 2, p. 63, Jun. 2004.

[21] A. Vallabhaneni, T. Wang, and B. He, "Brain-computer interface," in *Neural Engineering*. Springer, 2005, pp. 85–121.

[22] N. Birbaumer and L. G. Cohen, "Brain–computer interfaces: communication and restoration of movement in paralysis," *The Journal of physiology*, vol. 579, no. 3, pp. 621–636, Jan. 2007.

[23] S.-P. Kim, J. D. Simeral, L. R. Hochberg, J. P. Donoghue, G. M. Friehs, and M. J. Black, "Point-and-click cursor control with an intracortical neural interface system by humans with tetraplegia," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 2, pp. 193–203, Apr. 2011.

[24] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, and P. van der Smagt, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, no. 7398, pp. 372–375, May 2012.

[25] A. Kilicarslan, S. Prasad, R. G. Grossman, and J. L. Contreras-Vidal, "High accuracy decoding of user intentions using EEG to control a lower-body exoskeleton," in *35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2013, pp. 5606–5609.

[26] A. Venkatakrishnan, G. E. Francisco, and J. L. Contreras-Vidal, "Applications of brain–machine interface systems in stroke recovery and rehabilitation," *Current Physical Medicine and Rehabilitation Reports*, vol. 2, no. 2, pp. 93–105, Jun. 2014.

[27] P. J. Cherian, R. M. Swarte, and G. H. Visser, "Technical standards for recording and interpretation of neonatal electroencephalogram in clinical practice," *Annals of Indian Academy of Neurology*, vol. 12, no. 1, p. 58, Jan. 2009.

[28] T. Bulea, A. Kilicarslan, R. Ozdemir, W. Paloski, and J. Contreras-Vidal, "Simultaneous scalp electroencephalography (EEG), electromyography (EMG), and whole-body segmental inertial recording for multi-modal neural decoding." *Journal of visualized experiments: JoVE*, no. 77, pp. 244–250, Jul. 2013.

[29] Y. He, K. Nathan, A. Venkatakrishnan, R. Rovekamp, C. Beck, R. Ozdemir, G. E. Francisco, and J. L. Contreras-Vidal, "An integrated neuro-robotic interface for stroke rehabilitation using the NASA X1 powered lower limb exoskeleton," in *36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2014.

[30] L. Gupta, B. Chung, M. D. Srinath, D. L. Molfese, and H. Kook, "Multichannel fusion models for the parametric classification of differential brain activity," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 11, pp. 1869–1881, Nov. 2005.

[31] M. B. Malarvili, P. Colditz, and B. Boashash, "A multi-channel fusion based newborn seizure detection," *Journal of Biomedical Science and Engineering*, vol. 7, no. 8, pp. 533–545, 2014.

[32] J. R. White, T. Levy, W. Bishop, and J. D. Beaty, "Real-time decision fusion for multimodal neural prosthetic devices," *PloS one*, vol. 5, no. 3, p. e9493, Mar. 2010.

[33] B. A. Olshausen, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, Jun. 1996.

[34] K. Lee, S. Tak, and J. C. Ye, "A data-driven sparse GLM for fMRI analysis using sparse dictionary learning with MDL criterion," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1076–1089, May 2011.

[35] B. Hamner, R. Chavarriaga, and J. d. R. Millán, "Learning dictionaries of spatial and temporal EEG primitives for brain-computer interfaces," in *ICML Workshop on Structured Sparsity: Learning and Inference*, no. EPFL-CONF-166740, 2011.

[36] P. Guo, J. Wang, X. Z. Gao, and J. Tanskanen, "Epileptic EEG signal classification

with marching pursuit based on harmony search method," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2012, pp. 283–288.

[37] Q. Yuan, W. Zhou, S. Yuan, X. Li, J. Wang, and G. Jia, "Epileptic EEG classification based on kernel sparse representation," *International journal of neural systems*, vol. 24, no. 04, p. 1450015, 2014.

[38] K. Huang and S. Aviyente, "Sparse representation for signal classification," in *Advances in neural information processing systems*, 2006, pp. 609–616.

[39] Y. Shin, S. Lee, J. Lee, and H.-N. Lee, "Sparse representation-based classification scheme for motor imagery-based brain–computer interface systems," *Journal of neural engineering*, vol. 9, no. 5, p. 056002, 2012.

[40] Y. Shin, S. Lee, M. Ahn, H. Cho, S. C. Jun, and H.-N. Lee, "Noise robustness analysis of sparse representation based classification method for non-stationary EEG signal classification," *Biomedical Signal Processing and Control*, vol. 21, pp. 8–18, 2015.

[41] B. Settles, "Active learning literature survey," *University of Wisconsin, Madison*, 2010.

[42] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "A survey of active learning algorithms for supervised remote sensing image classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 606–617, Jun. 2011.

[43] M. Crawford, D. Tuia, and H. Yang, "Active learning: Any value for classification of remotely sensed data?" *Proceedings of the IEEE*, vol. 101, no. 3, pp. 593–608, Mar. 2013.

[44] J. Jung, E. Pasolli, S. Prasad, J. C. Tilton, and M. M. Crawford, "A framework for land cover classification using discrete return LiDAR data: Adopting pseudo-waveform and

hierarchical segmentation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 2, pp. 491–502, Feb. 2014.

[45] J. Jung and M. M. Crawford, "Extraction of features from LIDAR waveform data for characterizing forest structure," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 3, pp. 492–496, May 2012.

[46] J. Tilton, Y. Tarabalka, P. Montesano, and E. Gofman, "Best merge region-growing segmentation with integrated nonadjacent region object aggregation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 11, pp. 4454–4467, Nov. 2012.

[47] M. Dalla Mura, J. Atli Benediktsson, L. Bruzzone, and B. Waske, "Morphological attribute profiles for the analysis of very high resolution images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3747–3762, Oct. 2010.

[48] M. Dalla Mura, A. Villa, J. Benediktsson, J. Chanussot, and L. Bruzzone, "Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 542–546, May 2011.

[49] I. Güler and E. D. Übeyli, "Adaptive neuro-fuzzy inference system for classification of EEG signals using wavelet coefficients," *Journal of neuroscience methods*, vol. 148, no. 2, pp. 113–121, 2005.

[50] A. Subasi, "EEG signal classification using wavelet feature extraction and a mixture of expert model," *Expert Systems with Applications*, vol. 32, no. 4, pp. 1084–1093, 2007.

[51] P. D. Velu and V. R. de Sa, "Single-trial classification of gait and point movement preparation from human EEG," *Frontiers in neuroscience*, vol. 7, 2013.

[52] J. E. Fowler, "The redundant discrete wavelet transform and additive noise," *Signal Processing Letters, IEEE*, vol. 12, no. 9, pp. 629–632, 2005.

[53] V. N. Vapnik and V. Vapnik, *Statistical learning theory*. Wiley New York, 1998, vol. 1.

[54] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Cambridge, MA: MIT Press, 2002.

[55] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv preprint arXiv:1304.5634*, 2013.

[56] K. Nigam and R. Ghani, "Analyzing the effectiveness and applicability of co-training," in *Proceedings of the ninth international conference on Information and knowledge management*. ACM, 2000, pp. 86–93.

[57] I. Muslea, S. Minton, and C. A. Knoblock, "Active learning with multiple views," *Journal of Artificial Intelligence Research*, pp. 203–233, 2006.

[58] S. Yu, B. Krishnapuram, R. Rosales, and R. B. Rao, "Bayesian co-training," *The Journal of Machine Learning Research*, vol. 12, pp. 2649–2680, 2011.

[59] V. Sindhwani, P. Niyogi, and M. Belkin, "A co-regularization approach to semi-supervised learning with multiple views," in *Proceedings of ICML workshop on learning with multiple views*. Citeseer, 2005, pp. 74–79.

[60] G. R. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *The Journal of Machine Learning Research*, vol. 5, pp. 27–72, 2004.

[61] F. R. Bach, G. R. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality,

and the smo algorithm," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 6.

[62] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf, "Large scale multiple kernel learning," *The Journal of Machine Learning Research*, vol. 7, pp. 1531–1565, 2006.

[63] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "SimpleMKL," *Journal of Machine Learning Research*, vol. 9, pp. 2491–2521, 2008.

[64] F. R. Bach, "Consistency of the group lasso and multiple kernel learning," *The Journal of Machine Learning Research*, vol. 9, pp. 1179–1225, 2008.

[65] Z. Xu, R. Jin, H. Yang, I. King, and M. R. Lyu, "Simple and efficient multiple kernel learning by group lasso," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 1175–1182.

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

[67] T. Diethe, D. R. Hardoon, and J. Shawe-Taylor, "Multiview fisher discriminant analysis," in *NIPS workshop on learning from multiple sources*, 2008.

[68] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[69] Y. Freund and R. Schapire, "A short introduction to boosting," *Journal-Japanese Society For Artificial Intelligence*, vol. 14, no. 771-780, p. 1612, 1999.

[70] G. Schohn and D. Cohn, "Less is more: Active learning with support vector machines," in *ICML*. Citeseer, 2000, pp. 839–846.

[71] P. Mitra, B. Uma Shankar, and S. K. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recognition Letters*, vol. 25, no. 9, pp. 1067–1074, Jul. 2004.

[72] E. Pasolli, F. Melgani, and Y. Bazi, "Support vector machine active learning through significance space construction," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 431–435, May 2011.

[73] H. L. Yang, Y. Zhang, S. Prasad, and M. Crawford, "Multiple kernel active learning for robust geo-spatial image analysis," in *Proc. 2013 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Jul. 2013, pp. 1218–1221.

[74] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. Fifth Annual Workshop on Computational Learning Theory*, 1992, pp. 287–294.

[75] W. Di and M. M. Crawford, "View generation for multiview maximum disagreement based active learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 5, pp. 1942–1954, May 2012.

[76] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.

[77] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 11, pp. 4085–4098, Nov. 2010.

[78] T. Luo, K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, and T. Hopkins,

"Active learning to recognize multiple types of plankton," in *Proc. IEEE International Conference on Pattern Recognition (ICPR)*, vol. 3, 2004, pp. 478–481.

[79] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, "SVM active learning approach for image classification using spatial information," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 4, pp. 2217–2233, Apr. 2014.

[80] A. Stumpf, N. Lachiche, J.-P. Malet, N. Kerle, and A. Puissant, "Active learning in the spatial domain for remote sensing image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 2492–2507, May 2014.

[81] D. Tuia and J. Munoz-Mari, "Learning user's confidence for active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 2, pp. 872–880, Feb. 2013.

[82] E. Pasolli, F. Melgani, N. Alajlan, and N. Conci, "Optical image classification: A ground-truth design framework," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 6, pp. 3580–3597, Jun. 2013.

[83] A. Liu, G. Jun, and J. Ghosh, "Spatially cost-sensitive active learning." in *SDM*. SIAM, 2009, pp. 814–825.

[84] J. Li, P. Reddy Marpu, A. Plaza, J. Bioucas-Dias, and J. A. Benediktsson, "Generalized composite kernel framework for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 9, pp. 4816–4829, Sep. 2013.

[85] G. R. Lanckriet, T. De Bie, N. Cristianini, M. I. Jordan, and W. S. Noble, "A statistical framework for genomic data fusion," *Bioinformatics*, vol. 20, no. 16, pp. 2626–2635, May 2004.

[86] D. Tuia, G. Camps-Valls, G. Matasci, and M. Kanevski, "Learning relevant image features with multiple-kernel classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3780–3791, Oct. 2010.

[87] Y. Gu, C. Wang, D. You, Y. Zhang, S. Wang, and Y. Zhang, "Representative multiple kernel learning for classification in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 7, pp. 2852–2865, Jul. 2012.

[88] G. Thoonen, Z. Mahmood, S. Peeters, and P. Scheunders, "Multisource classification of color and hyperspectral images using color attribute profiles and composite decision fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 510–521, April 2012.

[89] Z. Wang and X. Sun, "Multiple kernel local Fisher discriminant analysis for face recognition," *Signal Processing*, vol. 93, no. 6, pp. 1496–1509, Jun. 2013.

[90] B. Krishnapuram, L. Carin, M. A. Figueiredo, and A. J. Hartemink, "Sparse multinomial logistic regression: Fast algorithms and generalization bounds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp. 957–968, Jun. 2005.

[91] N. Cristianini, J. Kandola, A. Elisseeff, and J. Shawe-Taylor, "On kernel target alignment," in *Innovations in Machine Learning*. Springer, 2006, pp. 205–256.

[92] E. W. Gobel, T. B. Parrish, and P. J. Reber, "Neural correlates of skill acquisition: decreased cortical activity during a serial interception sequence learning task," *Neuroimage*, vol. 58, no. 4, pp. 1150–1157, 2011.

[93] C. Kranczioch, S. Athanassiou, S. Shen, G. Gao, and A. Sterr, "Short-term learning of

a visually guided power-grip task is associated with dynamic changes in EEG oscillatory activity," *Clinical Neurophysiology*, vol. 119, no. 6, pp. 1419–1430, 2008.

[94] https://askabiologist.asu.edu/what-your-brain-doing.

[95] http://www.neuroskills.com/brain-injury/brain-function.php.

[96] M. Schwarzbold, A. Diaz, E. T. Martins, A. Rufino, L. N. Amante, M. E. Thais, J. Quevedo, A. Hohl, M. N. Linhares, and R. Walz, "Psychiatric disorders and traumatic brain injury," *Neuropsychiatric disease and treatment*, vol. 4, no. 4, p. 797, 2008.

[97] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large Margin Classifiers*, vol. 10, no. 3, pp. 61–74, Mar. 1999.

[98] H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on Platts probabilistic outputs for support vector machines," *Machine Learning*, vol. 68, no. 3, pp. 267–276, Oct. 2007.

[99] S. Prasad and L. M. Bruce, "Decision fusion with confidence-based weight assignment for hyperspectral target recognition," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 5, pp. 1448–1456, May 2008.

[100] W. Di and M. M. Crawford, "Multi-view adaptive disagreement based active learning for hyperspectral image classification," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2010, pp. 1374–1377.

[101] N. Shawe-Taylor and A. Kandola, "On kernel target alignment," *Advances in Neural Information Processing Systems*, vol. 14, pp. 367–373, 2002.

[102] SimpleMKL toolbox, Online available: http://asi.insa-rouen.fr/enseignants/~arakoto/code/mklindex.html.

[103] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[104] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10, pp. 3973–3985, Oct. 2011.

[105] Y. Zhang, B. Du, and L. Zhang, "A sparse representation-based binary hypothesis model for target detection in hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1346–1354, Mar. 2015.

[106] K. Lounici, M. Pontil, A. B. Tsybakov, and S. Van De Geer, "Taking advantage of sparsity in multi-task learning," *arXiv preprint arXiv:0903.1468*, 2009.

[107] S. Negahban and M. J. Wainwright, "Joint support recovery under high-dimensional scaling: Benefits and perils of l1,-regularization," *Advances in Neural Information Processing Systems*, vol. 21, pp. 1161–1168, 2008.

[108] X.-T. Yuan, X. Liu, and S. Yan, "Visual classification with multitask joint sparse representation," *IEEE Transactions on Image Processing*, vol. 21, no. 10, pp. 4349–4360, Oct. 2012.

[109] S. Shekhar, V. Patel, N. Nasrabadi, and R. Chellappa, "Joint sparse representation for robust multimodal biometrics recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 113–126, Jan 2014.

[110] X. Zheng, X. Sun, K. Fu, and H. Wang, "Automatic annotation of satellite images via

multifeature joint sparse coding with spatial relation constraint," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 4, pp. 652–656, Jul. 2013.

[111] J. Li, H. Zhang, L. Zhang, X. Huang, and L. Zhang, "Joint collaborative representation with multitask learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 9, pp. 5923–5936, Sept. 2014.

[112] S. Shafiee, F. Kamangar, and V. Athitsos, "A multi-modal sparse coding classifier using dictionaries with different number of atoms," in *2015 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2015, pp. 518–525.

[113] Y. Y. Tang, H. Yuan, and L. Li, "Manifold-based sparse representation for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 12, pp. 7606–7618, Dec. 2014.

[114] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 3360–3367.

[115] C.-Y. Lu, H. Min, J. Gui, L. Zhu, and Y.-K. Lei, "Face recognition via weighted sparse representation," *Journal of Visual Communication and Image Representation*, vol. 24, no. 2, pp. 111–116, Feb. 2013.

[116] X. Tang and G. Feng, "Weighted group sparse representation based on robust regression for face recognition," in *Biometric Recognition*. Springer, 2012, pp. 42–49.

[117] Q. Shi, L. Zhang, and B. Du, "Semisupervised discriminative locally enhanced alignment for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 9, pp. 4800–4815, Sept. 2013.

[118] H. Zou, "The adaptive LASSO and its oracle properties," *Journal of the American statistical association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[119] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted l1 minimization," *Journal of Fourier analysis and applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[120] W. Deng, W. Yin, and Y. Zhang, "Group sparse optimization by alternating direction method," in *SPIE Optical Engineering Applications*. International Society for Optics and Photonics, 2013, pp. 88 580R–88 580R.

[121] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends ® in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[122] I. P. Pappas, M. R. Popovic, T. Keller, V. Dietz, and M. Morari, "A reliable gait phase detection system," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 9, no. 2, pp. 113–125, 2001.

[123] I. P. Pappas, T. Keller, S. Mangold, M. R. Popovic, V. Dietz, and M. Morari, "A reliable gyroscope-based gait-phase detection sensor embedded in a shoe insole," *IEEE Sensors Journal*, vol. 4, no. 2, pp. 268–274, 2004.

[124] S. J. M. Bamberg, A. Y. Benbasat, D. M. Scarborough, D. E. Krebs, and J. Paradiso, "Gait analysis using a shoe-integrated wireless sensor system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, no. 4, pp. 413–423, 2008.

[125] A. De Stefano, J. Burridge, V. Yule, and R. Allen, "Effect of gait cycle selection on EMG

analysis during walking in adults and children with gait pathology," *Gait & posture*, vol. 20, no. 1, pp. 92–101, 2004.

[126] C. M. Senanayake, S. Senanayake, and M. Arosha, "Computational intelligent gait-phase detection system to identify pathological gait," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 5, pp. 1173–1179, 2010.

[127] A. Mannini and A. M. Sabatini, "Gait phase detection and discrimination between walking–jogging activities using hidden Markov models applied to foot motion data from a gyroscope," *Gait & posture*, vol. 36, no. 4, pp. 657–661, 2012.

[128] J. Bae and M. Tomizuka, "Gait phase analysis based on a hidden markov model," *Mechatronics*, vol. 21, no. 6, pp. 961–970, 2011.

[129] C. D. Joshi, U. Lahiri, and N. V. Thakor, "Classification of gait phases from lower limb emg: Application to exoskeleton orthosis," in *2013 IEEE Point-of-Care Healthcare Technologies (PHT)*. IEEE, 2013, pp. 228–231.

[130] F. S. M. Jorquera, S. Grassi, P.-A. Farine, and J. L. Contreras-Vidal, "Classification of stance and swing gait states during treadmill walking from non-invasive scalp electroencephalographic (EEG) signals," in *Converging Clinical and Engineering Research on Neurorehabilitation*. Springer, 2013, pp. 507–511.

[131] M. Severens, B. Nienhuis, P. Desain, and J. Duysens, "Feasibility of measuring event related desynchronization with electroencephalography during walking," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2012, pp. 2764–2767.

[132] J. T. Gwin, K. Gramann, S. Makeig, and D. P. Ferris, "Electrocortical activity is coupled

to gait cycle phase during treadmill walking," *Neuroimage*, vol. 54, no. 2, pp. 1289–1296, 2011.

[133] M. P. Hernández, "A brain-computer interface for walking using EEG," 2012.

[134] B. R. Umberger and P. E. Martin, "Mechanical power and efficiency of level walking with different stride rates," *Journal of Experimental Biology*, vol. 210, no. 18, pp. 3255–3265, 2007.

[135] S. J. Lee and J. Hidler, "Biomechanics of overground vs. treadmill walking in healthy individuals," *Journal of applied physiology*, vol. 104, no. 3, pp. 747–755, 2008.

[136] M. Renfrew, R. Cheng, J. J. Daly, and M. C. Cavusoglu, "Comparison of filtering and classification techniques of electroencephalography for brain-computer interface," in *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* IEEE, 2008, pp. 2634–2637.

[137] R. N. Khushaba, A. Al-Ani, and A. Al-Jumaily, "Differential evolution based feature subset selection," in *19th International Conference on Pattern Recognition.* IEEE, 2008, pp. 1–4.

[138] E. Elhamifar, G. Sapiro, A. Yang, and S. S. Sasrty, "A convex optimization framework for active learning," in *Computer Vision (ICCV), 2013 IEEE International Conference on.* IEEE, 2013, pp. 209–216.

[139] J. Xu, H. He, and H. Man, "Active dictionary learning in sparse representation based classification," *arXiv preprint arXiv:1409.5763*, 2014.

[140] L. Shi and Y. Zhao, "Batch mode sparse active learning," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on.* IEEE, 2010, pp. 875–882.