Automated Remote Sensing Image Interpretation with Limited Labeled Training Data

by

Fan Li

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2015

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Automated remote sensing image interpretation has been investigated for more than a decade. In early years, most work was based on the assumption that there are sufficient labeled samples to be used for training. However, ground-truth collection is a very tedious and time-consuming task and sometimes very expensive, especially in the field of remote sensing that usually relies on field surveys to collect ground truth. In recent years, as the development of advanced machine learning techniques, remote sensing image interpretation with limited ground-truth has caught the attention of researchers in the fields of both remote sensing and computer science.

Three approaches that focus on different aspects of the interpretation process, i.e., feature extraction, classification, and segmentation, are proposed to deal with the limited ground truth problem. First, feature extraction techniques, which usually serve as a pre-processing step for remote sensing image classification are explored. Instead of only focusing on feature extraction, a joint feature extraction and classification framework is proposed based on ensemble local manifold learning. Second, classifiers in the case of limited labeled training data are investigated, and an enhanced ensemble learning method that outperforms state-of-the-art classification methods is proposed. Third, image segmentation techniques are investigated, with the aid of unlabeled samples and spatial information. A semi-supervised self-training method is proposed, which is capable of expanding the number of training samples by its own and hence improving classification performance iteratively. Experiments show that the proposed approaches outperform state-of-the-art techniques in terms of classification accuracy on benchmark remote sensing datasets.

Acknowledgements

First and foremost, I would like to express my gratitude to my advisors Dr. David Clausi and Dr. Alexander Wong for the continuous support of my Ph.D study and related research, for their patience, motivation, and immense knowledge. I appreciate the freedom Prof. Clausi gave me on choosing the research topics, and his help in all aspects from building up my thesis framework and shaping the research questions to improving my writing skills. I would also like to thank Prof. Wong for the numerous helpful discussions in the lab and critical comments on my papers and thesis. Without their guidance and encouragement in the last four years, I would not have accomplished my thesis research.

I am also grateful to my Ph.D committee members, Dr. Andrea Scott, Dr. John Zelek, and Dr. Zhou Wang at University of Waterloo and Dr. David Messinger at Rochester Institute of Technology for their enlightening ideas and discussions, as well as valuable comments and suggestions on my thesis.

Special thanks are given to Dr. Linlin Xu, Steven Leigh, Lei Wang, Jiange Liu, Jason Deglint, Robert Amelard, Dr. Parthipan Siva, Dr. Zhijie Wang, and all the other colleagues at the Vision and Image Processing Lab for their help, encouragement and friendship, as well as staff members in the Department of Systems Design Engineering, Ms. Vicky Lawrence, Ms. Janine Blair, Ms. Colleen Richardson, and Ms. Angie Muir for their kind help.

My thanks also go to MacDonald, Dettwiler and Associates (MDA) for providing the RADARSAT-2 imagery and ice analyst Don Isaacs for providing the corresponding ground truth. I also want to thank the Natural Science and Engineering Research Council of Canada (NSERC), the Canadian Space Agency (CSA), and the Canada Research Chairs (CRC) program for the financial support.

Finally and most importantly, I own my greatest debt of gratitude to my parents, for their support, understanding, and endless love.

Table of Contents

List of Tables

List of Figures

List of Symbols

β	Weight parameter
η_i	Set of neighbors of site i
$\gamma_l^{i,k}$	Localized weight for x_l^i related to cluster $\mathcal{C}_k$
$\lambda_l^{i,j}$	Local weight of sample x_l^i for class j
$\mathbf{F}$	Feature set
$\mathbf{X}$	Set of samples
$\mathbf{Y^r}$	Set of labels for regions
$\mathbf{Y}$	Set of labels for all the samples
$\mathcal{D}$	Training set
$\mathcal{L}$	Set of class labels
$\mathcal{M}_k$	The manifold related to the k^{th} cluster
$\mathcal{S}$	Set of integers indicating the discrete rectangular lattice
$\mathcal{S}_{Tr}$	Set of integers indicating the image locations of training samples
μ	Mean vectors
Ω	Set of regions in the current RAG
ϕ	Unary clique potentials
Σ	Variance matrix

σ	Localized weight
ξ	Pairwise clique potentials
A	Adjacency matrix
D	Diagonal weight matrix
H	Classifier
I	Identity matrix
L	Laplacian matrix
$M_j(x_l^i)$	Local mean of sample x_l^i for class j
P_i	Prior probability of class i
R^a	Rotation matrix
$S^{(b)}$	Between-class scatter
$S^{(w)}$	Within-class scatter
U	Localized weight matrix
W	Transformation matrix
Z	Normalization factor

Chapter 1

Introduction

1.1 Overview

Remote sensing is the acquisition of information about the Earth's surface by the electromagnetic energy emitted by or scattered from the surface, using sensors installed on airplanes or satellites [98]. Compared to site observation, remote sensing technology enables real-time collection of data on the surface in a large scale, especially in dangerous and inaccessible areas. Remote sensing imagery has been widely used in a variety of terrestrial, oceanographic, and atmospheric applications such as natural resource management, sea-ice monitoring, natural hazard assessment, etc. According to the Union of Concerned Scientists satellite database [49], the total number of operating satellites was 1,265 by January 31, 2015, including 309 satellites for earth observation.

Remote sensing systems can be divided into passive sensors and active sensors. Passive sensors receive energy that is naturally available, such as sunlight and thermal radiation that is emitted or reflected by the objects on the Earth's surface or in the atmosphere. The passive imagery can be further divided based on the number of spectral bands provided by the imaging system, such as panchromatic, multispectral, and hyperspectral imagery. Each spectral band covers a narrow wavelength range of the electromagnetic spectrum including visible radiation, infrared, and microwave. Active sensors, instead, provide their own energy source for illumination by emitting radiation directly to the target to be investigated. A representative active sensor is synthetic aperture radar (SAR), which is a microwave remote sensing technology that uses synthetic antenna aperture to create finer resolution imagery.

The key to make the imagery created by all these sensors useful is the effective conversion from the quantized pixel values of an image into some values that have physical meanings or some properties of the surface being sensed [98]. Usually a thematic map is finally created to display the geographical distributions of a specific phenomenon [98], such as land surface elevation and ice concentration. The most direct thematic map from a remote sensing imagery is a surface cover map which displays the existing land or marine covers in the area. The process of identifying objects based on the corresponding locations of pixels in the imagery is called interpretation.

So far, most of the interpretation tasks for remote sensing imagery are still performed manually. Some tasks require the interpretation of the entire image with very high accuracy, which usually take considerable time and efforts even for an experienced human interpreter. For some other tasks, even though the pixelwise mapping is not required, efficient automated mapping techniques are still valuable. Take the interpretation of sea ice imagery as an example. An ice expert first divides an image into multiple irregular polygons. The boundaries of those polygons are often between regions belonging to different ice types, and the number of classes in a single polygon can be reduced to be less than the number of classes in the whole image. Then, an egg code [37] is created for each polygon, recording the total concentration of ice, different ice types (stages of development), the concentration of each type, and the floe size in each type. This is sometimes not accurate because the concentration is estimated by the ice experts. The accuracy is dependent on the experience of the experts. Sometimes, the ice maps are inconsistent among different ice experts [72], in which case automated or semi-automated methods can help ice experts generate more accurate mapping with little or no inter-operator bias.

1.2 Motivation and research objectives

This thesis addresses the automated interpretation of remote sensing imagery with limited label information. The main research objective is to develop algorithms that can improve classification accuracy using limited labeled training data. Recently, the advancing imaging techniques of optical sensors have facilitated remote sensing image analysis by abundant images with rich spatial, spectral, and temporal information. Also, the development of data processing and data analysis technologies such as image processing and machine learning enable the fast acquisition of useful information from images with little or no human intervention.

Considering the huge volume of remote sensing images, it is necessary to develop automated approaches for remote sensing image interpretation. Over the past decade, a

lot of automated interpretation algorithms have been proposed for various types of remote sensing imagery. Many of these methods were developed based on an assumption that there are sufficient labeled samples used for training. However, ground truth is usually collected by field survey, which is expensive and sometimes very difficult to access in the remote sensing community. Consider the sea ice imagery as an example. The backscatter characteristics and location of sea ice change very quickly, so that the validation set may sometimes be inconsistent with the images even in a day. Even without any field survey, the manual delineation of ground truth on a remote sensing image of large size is very tedious and time-consuming. Moreover, due to the Hughes effect [64], large training sets are usually required for supervised methods, especially on the high-dimensional data such as hyperspectral data. Therefore, the difficulty of obtaining sufficient and accurate ground truth is a bottleneck of the automated interpretation of remote sensing imagery.

In recent years, the small-sample-size (SSS) problem for remote sensing image interpretation has attracted the interest of researchers in various fields including remote sensing, machine learning, and computer vision. The methods that have been proposed mainly focus on one or more of the following techniques:

- Feature extraction, or dimension reduction techniques that seek more discriminative features, usually in lower dimensionality compared to the original feature space using supervised, semi-supervised, or unsupervised techniques.
- Classification techniques that are capable of dealing with limited labeled training samples, including supervised and semi-supervised classifiers.
- Segmentation techniques that use spatial context to aid the imperfect pixelwise classification due to limited training samples, and finally generate a thematic map for the whole image.

In this thesis, the term classification only denotes the pixelwise classification that assigns each pixel a label independently without spatial context, while the term segmentation refers to grouping pixels into regions using spatial context without loss of generality [17, 88]. The above three techniques are partially or all included in the remote sensing interpretation approaches: feature extraction can serve as a pre-classification step, and segmentation can serve as a post-classification step. A simple way to include these techniques is to perform "feature extraction - classification - segmentation" in a three-step processing chain. In fact, however, one set of features may not be discriminative enough because the reduction of dimensionality may result in loss of information, and the incorporation of spatial context may be helpful for classification. Also, the problem

of limited label information tends to affect all the steps that require labeled samples for training.

1.3 Thesis structure

In this thesis, techniques for remote sensing image interpretation in the situation of limited labeled training data are investigated. To improve the accuracy of the machine interpretation, three approaches focused on feature extraction (Chapter 4), classification (Chapter 5), and segmentation (Chapter 6 & 7) respectively using limited labeled training data are designed, implemented, and tested. The rest of the thesis is organized as follows.

Chapter 2 reviews the literature of automated interpretation of remote sensing imagery. We focus on two typical types of remote sensing imagery: hyperspectral imagery and SAR imagery. Algorithms related to feature extraction, classification, and segmentation are reviewed, with an emphasis on examples using limited labeled training data. The hyperspectral data and SAR data used in the experiments of the thesis are described in Chapter 3.

Chapter 4 introduces a framework of joint feature extraction and classification based on the ensemble of localized manifolds. Multiple linear manifolds emphasizing on different locations in the feature space are learned to characterize the input data, and a classification ensemble is then trained using the features extracted via the different manifolds. Such manifolds are localized in the feature space, and can overcome the challenges and limitations associated with learning a single global manifold for characterizing complex data structures.

Chapter 5 investigates the performance of different classifiers using limited labeled training data in a supervised manner. An enhanced ensemble method which combines two ensemble algorithms is developed. The proposed classifier performs inherent feature extraction without reducing the dimensionality, and outperforms other ensemble methods in the situation of inadequate training samples and high dimensionality. Further, the proposed method innately produces posterior probabilities inherited from AdaBoost, which are used for the unary potentials of the conditional random field (CRF) model in order to incorporate spatial context information. This chapter is largely based on a published paper [86].

Chapter 6 proposes a joint classification and segmentation algorithm in a semi-supervised manner. The algorithm is also derived from the CRF framework. A multimodal Gaussian model is used to estimate the probabilities that serve as the unary potential of

CRF, and the edge strength is naturally incorporated into the energy function. Also, region merging is concatenated with the CRF inference to reduce the number of nodes iteratively. Moreover, the self-training technique is used, which iteratively enlarges the training sample set and retrains the classifier. The proposed method achieves reasonable classification accuracy for hyperspectral imagery using five labeled training samples per class.

Chapter 7 extends the algorithm proposed in Chapter 6 for automated ice-water interpretation using dual-pol RADARSAT-2 imagery. The daily interpretation of SAR sea ice imagery is very important for ship navigation and climate monitoring. SAR sea ice imagery usually has large image size and complex image properties. In the proposed method, backscatter, texture, and edge strength features are incorporated in a CRF model. The algorithm is tested on a large-scale RADARSAT-2 dual-polarization dataset. This chapter is largely based on a published paper [85].

Finally, Chapter 8 concludes the thesis and suggests future research directions.

Chapter 2

Literature review

2.1 Remote sensing imagery

There are many types of remote sensing image data based on a variety of imaging sensors, including panchromatic, multispectral, hyperspectral, and SAR images. This thesis focuses on hyperspectral imagery and SAR imagery. These two types of imagery have been widely used in many applications, and both of them have their own unique properties that make the automated interpretation challenging.

2.1.1 Hyperspectral imagery

Hyperspectral imagery is capable of providing rich spectral information of objects, and has been applied to many purposes such as agriculture, mineralogy, and land use/land cover management. Similar to other spectral imaging, hyperspectral imaging collects signals from across the range of electromagnetic spectrum by breaking into groups of bands of different wavelengths. By expanding and improving capability of multispectral sensors, hyperspectral imaging sensors have the capability of collecting a large number of very narrow wavebands. For example, the airborne visible/infrared imaging spectrometer (AVIRIS) sensor by National Aeronautics and Space Administration (NASA) acquires data in 224 bands in the range 0.4 to 2.5 μm with a bandwidth of 10nm [98]. Despite the fine spectral resolution, hyperspectral imagery can also be acquired with fine spatial resolution. The spatial resolutions of the hyperspectral imagery acquired from satellite systems are usually 30m or finer, and those from airborne systems are usually 5m or finer [46]. For

example, the imagery acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor used in the experiments of this thesis has a geometric resolution of 1.3m.

The main difficulty of processing hyperspectral data is the "Hughes phenomenon [64]" caused by the high dimensionality of spectral bands. The high-dimensionality of hyperspectral data leads to difficulties in estimating of statistical parameters such as covariance matrix for some classification approaches [98]. As the dimensionality of the data increases, large volumes of test and training data need to be collected for the parameter estimation, which is sometimes impracticable. Therefore, feature extraction methods that can reduce the dimensionality and robust classifiers for the SSS problem are required. Also, the abundant image details due to high spectral and spatial resolution lead to large within-class variation, so only applying pixelwise classification techniques may not be enough. In recent years, spectral-spatial methods [93,101,142] have been widely used for hyperspectral imagery to refine the classification result using spatial context.

2.1.2 SAR imagery

SAR imagery has been widely applied for oceanology, such as sea ice mapping, owing to its all-weather and all-day imaging capability. Sea ice is an important issue for both safety of navigation and climate monitoring. Owing to the development of remote sensing technology, rapid and large-scale mapping of sea ice using SAR satellites is available. Currently, one of the most advanced satellites for sea ice monitoring is the RADARSAT-2 satellite launched in December 2007 by the Canadian Space Agency [156].

The SAR system is a type of radar imaging system. Different from the real aperture radar, it uses relative motion between an antenna and the target region to achieve a finer spatial resolution. The SAR system is installed on an aircraft or spacecraft. Pulses of radio waves at different wavelengths are repeatedly emitted from a single beam-forming antenna to a target scene, and the echo waveforms received at different antenna positions are detected, stored and post-processed to resolve elements in an image of the target region [26].

SAR has multiple polarization modes because the radar waves are polarized. Anisotropic materials such as ice can reflect different intensities in different polarizations. A fully polarimetric mode includes horizontal transmit and horizontal receive (HH), vertical transmit and vertical receive (VV), horizontal transmit and vertical receive (HV), and vertical transmit and horizontal receive (VH). The first two polarization modes are called co-polarized modes, and the last two are called cross-polarized modes. Also, the signal is complex including both magnitude and the phase.

Different from optical imagery, SAR imagery is affected by speckle noise, which increases the difficulty of interpretation tasks. Speckle noise is a form of multiplicative, locally correlated noise [171]. Many filters have been proposed to reduce the noise [45,82,110,171]. However, the noise filters will affect the image details and change the texture.

Another problem for interpretation is the incidence-angle effect accentuated by a long swath. An incidence angle is defined by the angle between the perpendicular to the imaged surface and the direction of the incident radiation. Normally, backscatter will decrease with increasing incidence angle, causing the intensity inhomogeneity along SAR range direction in SAR images. However, the rate of variation depends on the land cover type and the polarization, which makes the incidence-angle effect difficult to be corrected.

The currently operational SAR sensors include TerraSAR-X (X-band SAR with 3.1 cm wavelength), RADARSAT-2 (C-band SAR with 5.6 cm wavelength), and ALOS-PALSAR (L-band SAR with 23.6 cm wavelength). For the RADARSAT-2 satellite, a very important purpose is to monitor sea ice. Compared to RADARSAT-1 satellite launched earlier, the RADARSAT-2 satellite can offer dual-polarization modes with wide swath coverage [111].

2.2 Feature extraction techniques

Feature extraction is the process of mapping the original measurement variables into informative and less redundant features, facilitating the subsequent tasks, such as data compression [34], classification [76], target detection, and spectral unmixing [87]. The reduction of feature dimensionality can avoid the curse of dimensionality and reduce computational costs. Generally speaking, there are two types of features: image features and non-image features.

2.2.1 Extraction of image features

Apart from intensity that is directly obtained from an image, image features such as texture and shape features are usually extracted based on a square window or a region. One of the most commonly-used texture features for remote sensing imagery is the gray-level covariance matrix (GLCM) [51], which provides a second-order method for generating texture features by capturing the correlations of pairwise pixels in a spatial window of interest [23]. The probability measure is defined as

$$Pr(x) = \{C_{ij} \mid (\delta, \theta)\} \tag{2.1}$$

where C_{ij} is the co-occurrence probability between grey levels i and j, δ is the interpixel distance and θ is the orientation. C_{ij} is defined as:

$$C_{ij} = \frac{p_{ij}}{\sum_{i,j=1}^{G} P_{ij}} \tag{2.2}$$

where P_{ij} is the number of occurrences of grey levels i and j, G is the quantized number of grey levels. Common GLCM statistics include uniformity, entropy, dissimilarity, contrast, inverse difference, inverse different moment, and correlation [23].

In the last decade, GLCM has been successfully used for classification using SAR imagery, especially for distinguishing open water or different ice types [2, 62, 84, 135]. Most of the previous work is related to analyzing the effectiveness of a single or multiple grey level co-occurrence matrix (GLCM) parameters (e.g., energy, contrast, variance, correlation, entropy and inverse difference moment) and testing on the pure samples. According to these papers, the methods reach satisfactory accuracy on the classification of most ice types in the experiments using some grey level co-occurrence probability (GLCP) coefficients, and the results can be further improved by using multiple displacement values to capture textures of sea ice in different scales [135].

Apart from GLCP, other texture features have also been explored. Clausi [22] analyzed and compared the performance of GLCPs, Markov random fields (MRF) parameters, and Gabor filters for sea ice classification, and the results show that GLCPs have the best classification accuracy, a little higher than Gabor filters. While a single MRF is much poorer than those two texture features, the combination of MRF and GLCPs can reach the highest accuracy. However, some classes are very visually similar and difficult to separate using only texture feature, e.g., to separate calm open water from smooth first-year ice, or to separate young ice floes from rough water. [22]. Kandaswamy et al. [71] introduced patch reoccurrences, and showed that the approximated texture features extracted using the GLCM can improve efficiency without significant degradation in the classification results. Aytekin et al. [1] proposed a feature descriptor called local primitive (LP) pattern based on an adaptive neighborhood instead of a fixed size window, and information such as sizes, intensity levels, and repetitiveness of the LPs is incorporated in the feature vectors.

The shape feature is another type of commonly-used feature. One example of shape descriptors is based on the morphological filter [7, 113, 147]. The residual between the original image and the open or close morphological transformation result is used as a morphological characteristic. There are some other ways of extracting shape features, for example, by calculating the area and the ratio between the length and width of the

regions [13]. For SAR sea ice imagery, Soh et al. [136] has summarized some shape-based features such as roundness, elongation, and irregularity that can be helpful for sea ice classification. Yu and Clausi [169] also included shape information into the energy function to detect leads and ice floes. However, extracting effective shape-based features is very difficult. The main reason is that the boundaries can be vague, making the extraction of shape features difficult. Moreover, regions generated by image segmentation techniques are either oversegmented or undersegmented, and thus do not match well with the real objects [169].

2.2.2 Extraction of non-image features

The non-image feature extraction methods are used to transform the original features into new features in lower dimensionality based on some criteria. These methods are also called dimension reduction or manifold learning methods, which are mainly applied to hyperspectral imagery [69, 79, 94, 95, 150]. To minimize the impact of the Hughes phenomenon, various feature extraction techniques have been developed as a preprocessing step before classification, so that useful information such as feature structure and class separability can be maintained in the new feature space, while the dimensionality of data is significantly reduced. A recent review of non-image feature extraction techniques for hyperspectral image classification is by Lunga et al. [96]. Compared to the texture or shape features, the new features extracted by these methods usually have no physical meaning. These methods can be divided into supervised and unsupervised methods. A well-known unsupervised feature extraction method is the principal component analysis (PCA), which seeks for a small set of linearly uncorrelated variables called principal components that can preserve most of the variance. The problem is solved by eigen-decomposition. Another similar method is the minimum noise fraction (MNF) [47], which can be considered as a noise-adjusted version of PCA, and has also been widely used for hyperspectral imagery [98].

However, the drawback of the variance criterion in both PCA and MNF is that, since these methods are both unsupervised, they are not directly related to classification. In contrast, linear discriminant analysis (LDA), which is another commonly-used technique, seeks the projection that maximizes the between-class variation and minimizes the within-class variation using the label information:

$$W = \arg \max_{W \in \Re^{d \times r}} \left[tr \left\{ (W^T S^{(w)} W)^{-1} W^T S^{(b)} W \right\} \right] \tag{2.3}$$

where $S^{(w)}$ and $S^{(b)}$ are the within-class scatter and the between-class scatter in the original feature space, and d and r represent the number of features in the original space and the new space respectively.

Similar to PCA, LDA can be also solved by eigen-decomposition, i.e., finding the eigenvalues of $(S^{(w)})^{-1}S^{(b)}$. The main problem of LDA includes its unimodal assumption and rank-deficiency problem (i.e., the number of extracted features can be at most the number of classes). Many variants of LDA have been proposed to overcome the rank-deficiency problem. A notable LDA-based algorithm is the local Fisher discriminant analysis [138]. It combines LDA and locality preserving projections (LPP), which is an unsupervised dimensionality reduction algorithm that works well with multimodal data. Li et al. [94] used local Fisher discriminant analysis (LFDA) for hyperspectral image classification, and found its combination with Gaussian mixture model (GMM) classifiers outperforms traditional approaches. Another LDA-based algorithm is the nonparametric weighted feature extraction (NWFE) algorithm [79]. It is a nonparametric extension of LDA. In NWFE, different weights for every sample are assigned based on the distance from the sample to the "weighted mean" to compute the scatter matrix. Both LFDA and NWFE are capable of dealing with multimodal data and generating full-rank scatter matrices.

In recent years, more and more attention has been given to the effects of limited training data on the supervised feature extraction methods. For LDA, a large number of training samples are usually required for estimating the within-class and between-class scatter matrices if the feature dimensionality is high [98]. Kuo and Chang proposed regularized feature extraction (RFE), and found that the RFE-regularized NWFE outperforms RFE-regularized LDA for the SSS problem [78]. Ly et al. [104] used a sparse graph-based method which does not require evaluation of the within-class scatter matrix and between-class scatter matrix. Imani and Ghassemian [65, 66] addressed the situation of limited training samples using "attraction points" and cluster-based feature extraction methods. Also, some supervised feature extraction methods have been extended into semi-supervised methods by incorporating unlabeled training samples. For example, a semi-supervised version of LFDA [139] is the combination of LFDA and PCA. Liao et al. [95] proposed a framework to combine LDA and several unsupervised local linear feature extraction methods. Shi et al. [131] proposed a semi-supervised graph-based feature extraction method, and selected unlabeled samples based on the segmentation result.

Most of the above-mentioned manifold learning methods are linear methods. Considering the complexity of some high-dimensional data such as hyperspectral image data, nonlinear feature extraction methods have been explored in recent years. Representative methods include Isomap [143], local linear embedding [121], Laplacian

Eigenmaps [5], etc. An alternative way to capture the nonlinearity is to kernelize a linear method using the "kernel trick" [16] to implicitly map the features to the induced feature space and compute the inner product. For example, kernel PCA [127] and generalized discriminant analysis [4] are the kernelized versions of PCA and LDA respectively. Besides kernelization, there are also approaches of adapting the data to multiple local manifolds. Hastie and Tibshirani [53] first developed an algorithm that combines with nearest neighbor classification and uses local LDA to estimate the metric for computing the neighborhood of each sample. Kim and Kittler [75] proposed a nonlinear discriminant analysis method using a set of linear transformations called locally linear discriminant analysis to deal with multimodally distributed face classes. Wang et al. [151] also used the same idea for metric learning, i.e., learning multiple local metrics.

2.3 Classification techniques

Classification is the process of assigning or allocating individual objects into different classes [98]. In this thesis, classification techniques refer to methods without using spatial proximity. There are different categories of classification techniques depending on the availability of labeled training samples, including unsupervised methods, supervised methods, and semi-supervised methods. For unsupervised methods such as k-means and GMM, the labels of samples are unavailable, and the cluster label is learned directly from the data and has no physical meaning. In this thesis, we focus on supervised and semi-supervised methods.

Given sufficient labeled training data, the difference between different classifiers is negligible because they all converge at or close to the Bayes error rate [54]. However, this is unrealistic in practice, especially for classification of remote sensing data for which the acquisition of ground truth is usually expensive and time-consuming. Due to the difficulty and costs of obtaining ground truth for remote sensing imagery, classification using few labeled samples, i.e., classification in the SSS scenario has attracted the attention of remote sensing researchers in recent years [68, 163].

2.3.1 Supervised classification methods

A well-known classifier in the remote sensing community is the maximum likelihood classifier (MLC) [98]. Usually the conditional probability is assumed to follow a Gaussian distribution assumption, and the parameters (i.e., the mean and variance) are estimated

by training samples based on maximum likelihood. The predicted labels are determined by the maximum a posteriori (MAP) criterion:

$$\hat{y} = \underset{\forall k}{\operatorname{argmax}}\{P(\mathbf{x_i} \mid \mathbf{y} = \mathbf{k})\mathbf{P}(\mathbf{y} = \mathbf{k})\} \tag{2.4}$$

where $\hat{y}$ is the predicted label, $P(\mathbf{x_i} \mid \mathbf{y} = \mathbf{k})$ is the conditional probability, and $P(y = k)$ is the prior probability which is pre-determined in the supervised context.

In practice, however, one class may have more than one Gaussian modality due to the complexity of data. Li et al. [94] assumed that each class consists of multiple Gaussian modes for hyperspectral image classification. They first estimated the number of modalities in each class using some criterion such as Akaike information criterion (AIC) or Bayesian information criterion (BIC), and then estimate the parameters using expectation maximization (EM). However, the main problem of MLC is that it requires sufficient training samples to learn the sample variance for high-dimensional data [98]. Mather [97] suggests based on empirical experience that the number of labeled training samples for each class should be at least 30 times more than the number of features.

Another commonly-used supervised classifier is the support vector machine (SVM) [25]. Different from MLC, SVM is a nonparametric model because it does not require the estimation of statistical parameters. SVM was designed originally for binary classification. It is designed for finding the maximum gap between two classes by solving a convex quadratic programming problem. One theoretical advantage of SVM over some other classification methods is that it approximately implements the structural risk minimization (SRM) by minimizing the empirical error and the VapnikChervonenkis dimension simultaneously, so that the upper bound of the expected risk can be minimized [107], even though such a upper bound is very conservative in practice. Also, the same kernel trick as that in feature extraction can be used for SVM to make it a nonlinear classifier.

For the standard SVM, there are mainly two limitations. First, the theory is established mainly for binary classification problems; second, it returns "hard" binary predictors, while in many applications probabilistic outputs are required. An attempt to overcome both limitations is relevance vector machine (RVM) [145]. RVM is a Bayesian sparse kernel technique which can generate fewer relative vectors than the support vectors in SVM, and thus has faster performance in testing. However, the training computational complexity of RVM is $O(N^3)$ given N training samples, which is higher than SVM for large datasets [27]. To address the multi-class problem, Weston and Watkins [155] proposed a multi-class SVM algorithm that solves the optimization problem in a single framework. In practice, however, methods based on multiple binary classifiers, especially one-against-all (OAA)

has been widely used in spite of their practical limitations [8]. To generate probabilistic outputs, Platt [114] trained the parameters of an additional sigmoid function to map the SVM outputs into probabilities. This method was applied by Wu et al. [157] who proposed two OAA-based approaches to estimate the probability for multi-class classification. Both methods can be reduced to linear systems so that they are easy to implement, and one of the methods is later implemented in the LibSVM library [20].

In contrast to MLC and SVM that are both single classifiers, there is another category of classification methods called ensemble learning, which combines multiple weak classifiers into a strong ensemble classifier. One state-of-the-art ensemble method is the random forest, which has been applied for remote sensing data [44,109]. It consists of many decision trees and outputs the class which is the mode of the output classes by individual trees [54]. It combines the "bagging" method (bootstrap aggregating) and the random selection of features to construct a collection of decision trees with controlled variation. There are also some variants of the random forest algorithm. The extremely randomized tree method [43] is one of them, which is designed for reducing the variation by further de-correlating the individual trees. The cut-point of each tree is selected fully at random, that is, independent of the target variable. Another variant is the rotation forest algorithm [120], which performs an innate feature transformation on the data for each base classifier. Compared to SVM, ensemble methods are more natural to be used for multi-label classification problems. Also, they can deal with thousands of input variables without variable deletion and gives estimates of what variables are important in the classification. So, a dimension reduction preprocessing step is usually not required. In fact, a random forest itself can be used for feature selection [122]. Moreover, ensemble methods have been demonstrated to work very well for the SSS problem [153, 162, 163].

2.3.2 Semi-supervised classification methods

Semi-supervised classification methods are designed for learning a classification model using both labeled and unlabeled training samples. They are developed for the scenarios that a small amount of labeled data and a large amount of unlabeled data are available. There are different types of semi-supervised learning methods, including generative models [105], low-density separation methods [70], and graph-based methods [100]. A survey of semi-supervised learning methods can be found in a previous paper [181].

Semi-supervised classification methods have also been used for remote sensing imagery. In fact, the first approach to investigate the effect of unlabeled samples to address the SSS problem was proposed in the remote sensing community, and it dates back to the

year 1994 [130]. Later, Jackson and Landgrebe [67] proposed an iterative procedure to adaptively estimate the covariance matrices for MLC. Recently, more and more self-supervised learning algorithms have been applied to remote sensing data, such as low-density separation methods based on SVM [15, 21], graph-based methods [15], and self-training [33]. These methods have been demonstrated to outperform supervised methods using the same number of training samples. Beyond the improvement, however, the classification accuracy is usually still unsatisfactory due to the insufficiency of labeled training samples. Also, previous empirical experiments showed that semi-supervised methods might make no improvement or even harm the classification performance [181]. Especially for the remote sensing imagery, there may be some pixels in the image which belong to none of the classes in the training areas. There might be negative effect on the classification performance if these pixels are selected for training. Even though, most semi-supervised methods select the training samples ad hoc over the whole image. One exception is a recently-published paper [33], in which only samples spatially surrounding the training samples have the chance to be selected.

2.4 Image segmentation techniques

Image segmentation is the process of partitioning an image into meaningful parts. Segmentation can be either used for a pre-classification step in a region-based approach to obtain superpixels, or a post-classification step to refine the pixelwise classification result. There is a huge amount of literature on image segmentation algorithms. Here we only focus on two types of segmentation methods: methods based on region growing and methods based on random fields.

2.4.1 Region growing methods

Region growing methods are bottom-up methods that start at an over-segmentation result, and then merge regions which have similar property into larger ones. The over-segmented result is commonly performed by the watershed algorithm [149]. The basic idea of the watershed algorithm is to simulate the forming of the real watershed by first putting seeds into an image like drops of water, and then flooding the image and building barriers when different water sources meet. For the region merging step, some merging and stopping criteria are defined. Regions that belong to the same object should be merged to make the following task more efficient. Moreover, regions that belong to different objects should not be merged because the merging process is usually irreversible.

Image cues such as intensity and edge strength can be used in the merging criteria. Hojjatoleslami et al. [61] proposed two more discontinuity measurements: average contrast and peripheral contrast. To improve the merging efficiency, Haris et al. [52] proposed a fast region merging method based on an edge-preserving noise reduction preprocessing step and a region adjacency graph. The merging criteria are usually based on statistical tests. Nock and Nielsen [106] developed a statistical region merging method which uses a merging predicate that is simple to implement, and a simple ordering to achieve low segmentation error. The algorithm can be approximated in linear time. Peng et al. [112] also used a statistical test, i.e., the sequential probability ratio test with the minimal cost criterion for the merging criterion.

Region growing methods have also been used for remote sensing imagery, mainly for SAR imagery to reduce the computational cost and resist the speckle noise. Carvalho et al. [19] used a statistical region growing procedure by combining the Kolmogorov-Smirnov test with a hierarchical stepwise optimization algorithm for SAR image segmentation. Yu et al. [166] used a context-based hierarchical merging method containing a coarse merging and a fine merging stage for SAR image segmentation, and different types of features including intensity, texture, edge, and spatial information are incorporated into the model. Specifically for SAR sea ice imagery, high level features such as texture, shape, and edge smoothness have been used due to the immense within-class variation of backscatter intensity. Sun et al. [140] compared a pixel-based method based on adaptive filtering with a region-based method based on segmentation using edge detection and region growing algorithms. However, the accuracy of the region-based method is just slightly higher than the pixel-based method because the performance of the former is largely dependent on the performance of the edge detector. Haverkamp et al. [55] proposed a method in which an image is first separated into regions, and a dynamic local thresholding technique is then used to merge those regions. It uses several high-level features such as perimeter, area and wiggliness to build a rule-based expert system based on the features. The expert system was later improved and developed into an intelligent system for satellite sea ice image analysis named ARKTOS [136], which used Dempster-Shafer theory to imitate the reasoning process of ice analysts. ARKTOS first uses the watershed algorithm to obtain initial over-segmentation results, and then merges the neighboring regions iteratively based on multiple classification rules. Yu and Clausi [169, 170] developed a joint segmentation and classification method named iterative region growing using semantics (IRGS) for SAR sea-ice analysis. It is a hierarchical algorithm in which the segmentation part is based on a region-growing technique and the classification part is a region-based MRF approach. The two processes are integrated under the Bayesian framework to enable the classification results to guide the region-growing process in a new iteration.

Apart from SAR imagery, region merging methods are also used for images with large homogeneous regions, such as high spatial resolution images. Bruzzone and Carlin [13] used region merging to perform multi-scale feature extraction, and used the SVM for classification. Li et al. [91] used multiple features including texture, spectral, and shape to derive a distance measure for the region adjacency graph. Tilton et al. [144] integrated region growing with nonadjacent region object aggregation, and their refined algorithm is capable of processing large-size hyperspectral images. Region merging methods are mainly used for unsupervised segmentation. For those methods used for supervised classification, region merging is used either as a preprocessing step [13] to facilitate feature extraction, or as a post-processing step to refine the result [144].

2.4.2 Random field methods

Random field methods, most predominantly MRF, have been used to incorporate spatial context for different types of remote sensing images in both supervised and unsupervised manners. For SAR imagery, MRF can help generate smooth segmentation results when the images are corrupted by severe speckle noise. Rignot and Chellappa [118] first used MRF for the segmentation of polarimetric SAR data. The classification accuracy is shown to be much better than the methods without taking spatial context into account. Even though, there are some drawbacks. For example, the line segments such as roads may disappear partially or entirely, and region borders fail to be located accurately. To overcome these drawbacks, Smits and Dellepiane [134] proposed MRF with adaptive neighborhoods, in which a Bayesian inference is used to choose the shapes of the local neighborhood systems for the region label process based on various information sources. Tison et al. [146] used the Fisher distribution in the MRF framework to deal with the non-Gaussian scattering statistics in SAR images when the resolution is high or the area is man-made. Deng and Clausi [28] used a varying weighting parameter in an annealing step. The feature modeling component is first dominant in the MRF model they proposed, and then the weight of the region labeling component increases so as to refine the segmentation results. Another modification on the standard MRF model in recent years is the triplet Markov field model [6] to solve the nonstationary property of real images. A third random field is introduced into an earlier proposed pairwise Markov field model and the Markovianity of the triplet is assumed. This model is further improved later by either adding an edge penalty [159] or using a multiscale model in wavelet domain [174].

For supervised classification methods, random fields are often used in a post-processing step after the probabilistic estimates by pixelwise classification methods are generated. Any classifier that is capable of generating probabilistic estimates can be combined with

a random field, such as GMM [93], SVM [142], multinomial logistic regression [90], and rotation forest [160]. However, the traditional MRF model that is derived from a Bayesian framework requires the conditional probabilities to be independent. Also, the pairwise potentials can be only related to the labels, so that data-dependent features such as edge strength are not allowed to be incorporated. The conditional random field (CRF), another type of random fields, overcomes these limitations. The CRF models the posterior probabilities directly without using the Bayes' rule, so the rigid conditional independence assumption is relaxed. Also, features related to both the observations and the labels can be arbitrarily integrated into the pairwise potentials. The CRF was first proposed by He et al. for image labeling [58]. In the remote sensing community, Zhong and Wang [177] introduced CRF to classification of hyperspectral images, and demonstrated its superiority to MRF. Zhang and Jia [173] incorporated a boundary constraint in order to reduce model parameters.

In a CRF model, the unary clique potentials can be naturally represented by a multinomial logistic regression (MLR) [173, 176, 177]. Theoretically it is the most suitable model to fit into CRF because logistic regression is by itself a simple form of CRF. More generally, other classifiers such as boosting [132] and SVM [178] can be also used. For most image labeling approaches, the parameters of the unary potentials are trained independently. For the pairwise energy term, generalized Ising model [173, 177] and multiscale region connection calculus model [137] that are dependent on the observed data can be used to model the spatial interaction.

One important issue of random field-based methods is the determination of the weight parameters between the unary and pairwise potentials. For supervised classification, the weight parameters can be selected by exhaustive grid search. To avoid the high computational cost resulted from the grid search, Serpico and Moser [129] proposed an automatic supervised procedure for optimizing the weighting parameters based on the Ho-Kashyap algorithm. However, when the number of labeled training samples is limited, grid search or other automatic selection methods might not be reliable.

2.5 Research questions

Despite the huge volume of literature related to remote sensing image classification, the situation when there is limited labeled training data available has not been fully explored. In this thesis, we aim to develop algorithms that can achieve reasonable classification accuracy by using limited labeled training data, with emphasis on feature extraction, classification, and segmentation, respectively.

1. For feature extraction, the performances of supervised, semi-supervised, and unsupervised methods for classification are measured in the case of limited labeled training samples. Also, due to the limitation of using single linear manifolds, methods using multiple linear manifolds are investigated. We want to see if the classification accuracy can be improved by using multiple linear manifolds for limited training samples. These questions will be investigated in Chapter 4.

2. For classification, the performance of common classifiers such as SVM and random forests for limited training samples are measured. Is it possible to design an ensemble classifier that has both small bias and variance? Moreover, we would like to measure the improvement of classification performance by using spatial context models as a post-processing step. These questions will be investigated in Chapter 5.

3. For segmentation, we would like to know if segmentation methods, including random fields and region merging methods introduced before, can used not only as a post-processing step. Also, can semi-supervised methods that make use of the unlabeled samples can address the problem of limited label information? These questions will be investigated in Chapter 6 and Chapter 7.

Chapter 3

Datasets

3.1 Hyperspctral datasets

Three standard hyperspectral datasets [3], including the University of Pavia dataset, the Kennedy Space Center dataset, and the Salinas dataset, are used for evaluation of algorithms in the thesis (Fig. 3.1).

The University of Pavia dataset, as shown in Fig. 3.1 (a), was acquired by the ROSIS sensor in Pavia, Italy. The original image has 115 spectral bands with a spectral range from 0.4μm to 0.9μm. A total number of 103 spectral reflectance bands are used for analysis after the noisy bands are removed. The spatial resolution of the image is 1.3 meters. The size of the dataset used in the experiment is 610×340 pixels. There are nine classes in the ground-truth image. The number of samples in each class in the training areas is shown in Table 3.1.

The Salinas image, as shown in Fig. 3.1 (b), was acquired by the AVIRIS sensor over Salinas Valley, California. Its geometric resolution is 3.7 meters. The area covered contains 512 lines by 217 samples. It has 224 spectral reflectance bands ranging from 0.4μm to 2.5μm, and 24 bands covering the water absorption bands have been discarded. There are 16 classes in the ground-truth image. The number of samples in each class in the training areas is shown in Table 3.2.

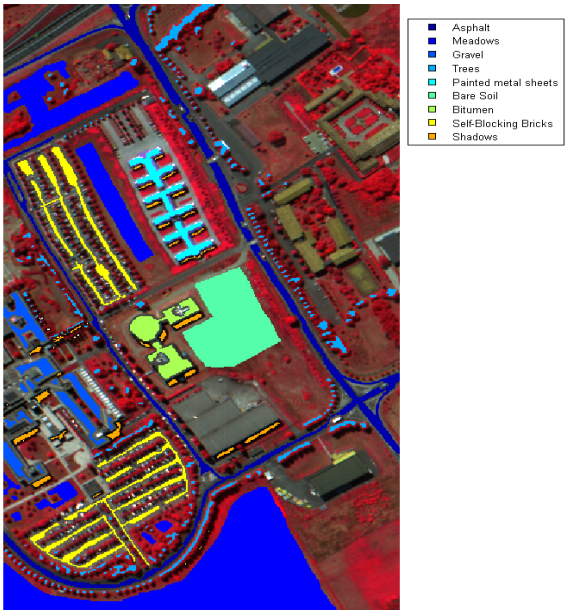
The Kennedy Space Center dataset, as shown in Fig. 3.1 (c), was acquired by the AVIRIS instrument over the Kennedy Space Center, Florida on March 23, 1996 [50]. The original image has 224 bands with a spectral range from 0.4μm to 2.5μm. A total number of 176 bands are used in the experiment after the water absorption and noisy bands are

Table 3.1: Class names and the number of samples of the University of Pavia dataset.

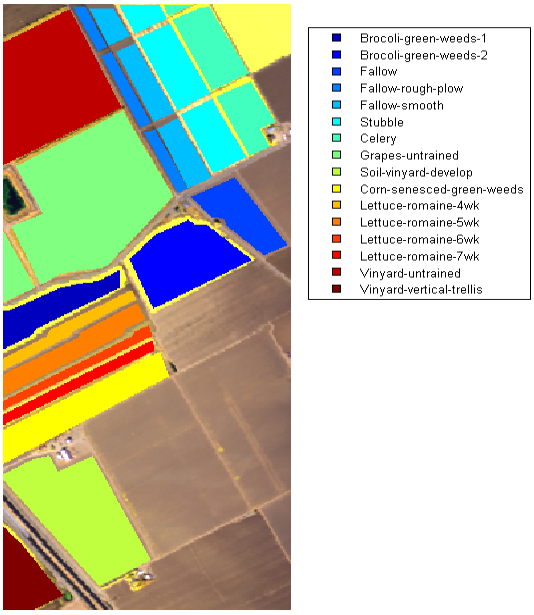
Class	Name	No. of Samples
1	Asphalt	6631
2	Meadows	18649
3	Gravel	2099
4	Trees	3064
5	Painted metal sheets	1345
6	Bare Soil	5029
7	Bitumen	1330
8	Self-Blocking Bricks	3682
9	Shadows	947

Table 3.2: Class names and the number of samples of the Salinas dataset.

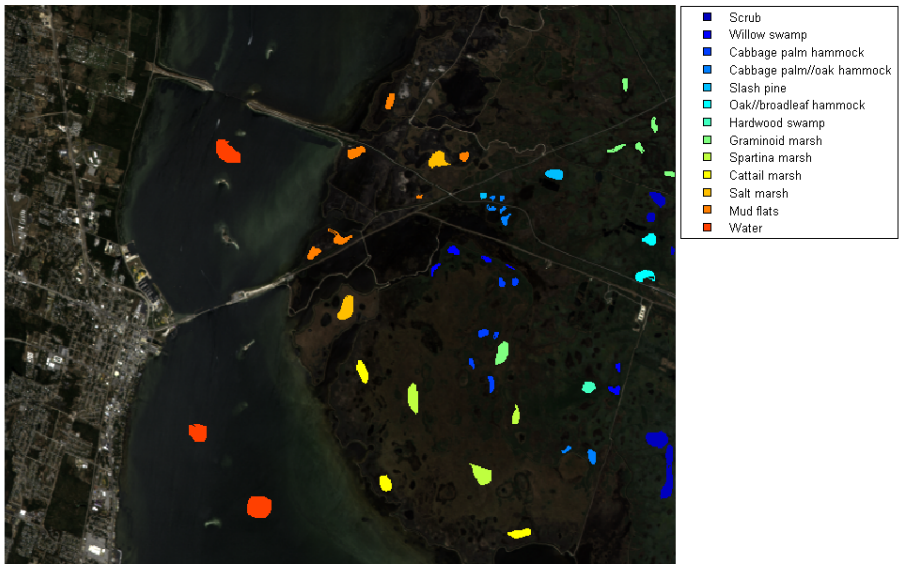
Class	Name	No. of Samples
1	Brocoli-green weeds-1	2009
2	Brocoli-green weeds-2	3726
3	Fallow	1976
4	Fallow-rough plow	1394
5	Fallow-smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes-untrained	11271
9	Soil-vinyard develop	6203
10	Corn-senesced green weeds	3728
11	Lettuce romaine-4wk	1068
12	Lettuce romaine-5wk	1927
13	Lettuce romaine-6wk	916
14	Lettuce romaine-7wk	1070
15	Vinyard-untrained	7268
16	Vinyard-vertical trellis	1807



(a) Rosis University of Pavia image



(b) AVIRIS Salinas image



(c) AVIRIS Kennedy Space Center image

Figure 3.1: False-color composition of three hyperspectral images with their training areas overlaid

removed. The spatial resolution of the image is 18 meters. The size of the dataset used in the experiment is 512×614 pixels. There are 13 land cover types in the training area of the reference data. The number of samples in each class in the training areas is shown in Table 3.3.

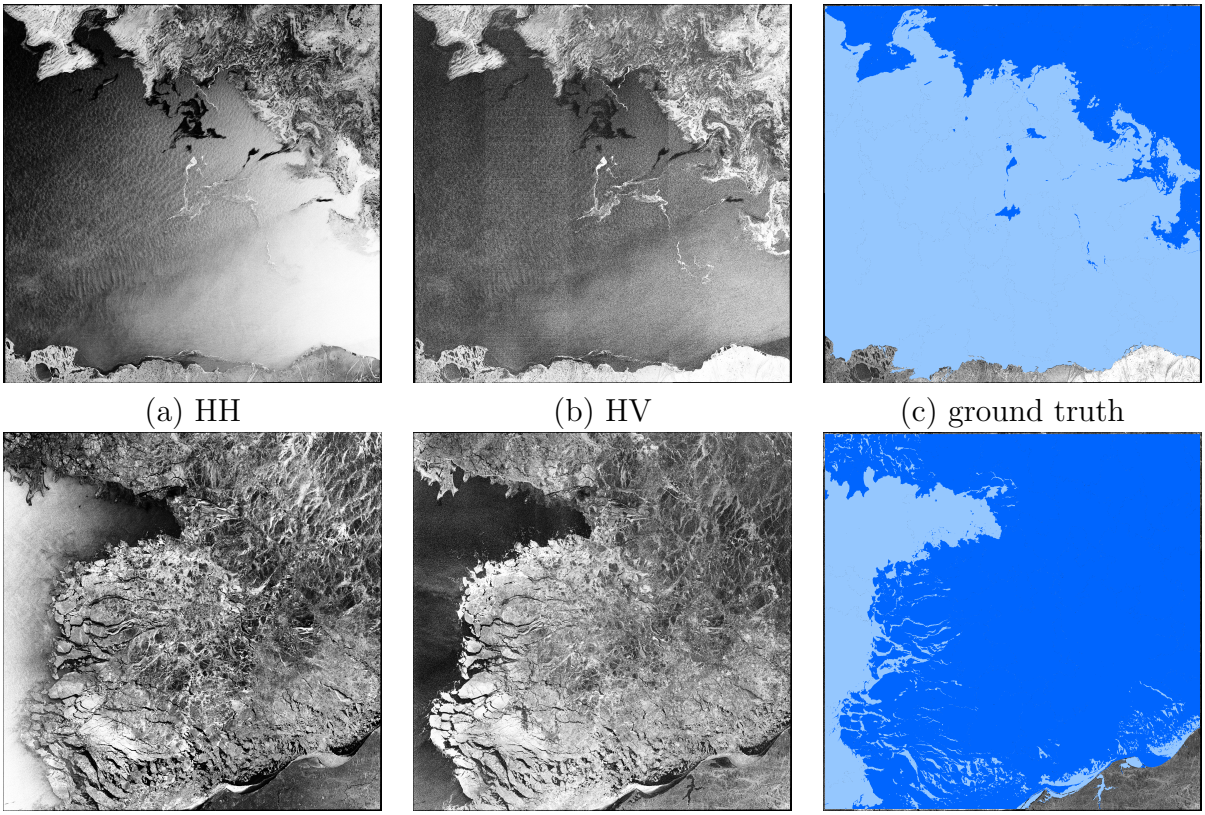
Table 3.3: Class names and the number of samples of the Kennedy Space Center dataset.

Class	Name	No. of Samples
1	Scrub	761
2	Willow swamp	243
3	Cabbage plam hammock	256
4	Cabbage plam/oak hammock	252
5	Slash pine	161
6	Oak/broadleaf hammock	229
7	Hardwood swamp	105
8	Graminoid marsh	431
9	Spartina marsh	520
10	Cattail marsh	404
11	Salt marsh	419
12	Mud flats	503
13	Water	927

3.2 SAR datasets

The SAR dataset used in this thesis, specifically in Chapter 7, was obtained from the C-band RADARSAT-2 SAR satellite over the Beaufort Sea and Chukchi Sea areas from May to December in the year 2010, and has been used in the previous publication [84]. They were captured in the ScanSAR Wide mode, which is mostly used for operational monitoring of sea ice at CIS [18]. Dual polarizations (HH and HV) are provided in the ScanSAR Wide mode. The spatial resolution of images is 50m, and the image sizes are around 10 000 × 10 000 pixels. The test images were acquired from both ascending and descending passes, with an incidence angle ranging from 20° to 49°.

In the pre-processing step, the log-transformed original images were down-sampled using 4 × 4 block averaging to reduce unnecessary computational cost [84]. Each test image has a size of about 2500 × 2500 pixels, as shown in Fig. 3.2. Even though the down-sampled images yield coarser classification results, the results are still far more detailed



(a) HH (b) HV (c) ground truth

(d) HH (e) HV (f) ground truth

Figure 3.2: Examples of Images in the SAR dataset. The first row corresponds to the data captured on October 3, 2010 and the related ground truth. The second row corresponds to the data captured on November 14, 2010 and the related ground truth.

than that provided by human interpretation [84]. The vector-based ground truth for the test images was made by an experienced ice analyst, as shown in Fig. 3.2 (c) and (f). To make a better evaluation of the algorithms, accurate pixelwise ground truth for each image has been created based on the vector-based ground truth for both training and validation.

Chapter 4

Joint feature extraction and classification using ensemble localized manifold learning

4.1 Introduction

General feature extraction methods for hyperspectral imagery have been reviewed in Section 2.2.2. In recent years, there has been increasing attention to the problem of limited training samples [115]. For LDA, a large number of training samples are usually required to determine the within-class and between-class scatter matrices. Kuo and Chang [78] found that NWFE with regularization outperforms LDA with regularization for the small-sample-size problem. Also, some supervised feature extraction methods have been extended into semi-supervised methods by incorporating the unlabeled training samples. Liao et al. [95] proposed a framework to combine LDA and an unsupervised local linear feature extraction (LLFE) method. Shi et al. [131] proposed a semi-supervised graph-based feature extraction method, and selected the unlabeled samples based on the segmentation result.

The purpose of the above methods is to learn an embedded subspace by a single linear transformation matrix. However, the structure of hyperspectral imaging data is typically so complex that the use of a single linear manifold may result in the loss of useful information, especially when we wish the dimensions of extracted features to be low. A traditional approach to capture nonlinear manifolds of high-dimensional data is to use the "kernel trick", such as kernel PCA [127] and generalized discriminant analysis [4]. However,

kernel methods usually have high computational complexity which is cubic in the size of training samples [16].

An alternative method to capture nonlinear manifolds is to use local manifolds or metrics by adapting to local sample points in the feature space. It is noted that all the geometric terms such as "local", "localized", and "distance" in this chapter refer to the feature space. An early attempt is the discriminant adaptive nearest neighbor classification algorithm (DANNC) [53]. This algorithm first uses local LDA to learn metrics for computing neighborhoods, and then adapts local neighborhoods based on local decision boundaries by a neighborhood-based classifier. However, the local metrics are learned independently, and thus the method is prone to overfitting [151]. A more recent approach is the multi-metric version of the large margin nearest neighbor method (LMNN-MM) [154] which first uses k-means to split training data into disjoint clusters, and then learns a Mahalanobis distance metric for each cluster. Although the number of local metrics is reduced from the number of training samples in DANNC to the number of clusters, overfitting is still unavoidable because the metrics are learned from separate parts of training samples independently. Moreover, such local methods will cause a discontinuity of the metrics near the k-means decision boundary. There are also approaches where the local manifolds of the training samples are represented by the weighted linear combination of multiple manifolds [75, 151], but a large number of parameters need to be estimated.

In the last decade, ensemble learning has been widely used to improve classification performance [12, 63, 176]. Combining multiple weak classifiers into a classifier ensemble can reduce the variance by a single classifier and thus prevent overfitting. All of the aforementioned issues related to manifold learning, along with the benefits of ensemble learning, motivates the proposed joint feature extraction and classification method in this chapter. Unlike traditional approaches that use feature extraction as a preliminary step followed by fitting a classifier on the extracted features, here we first learn multiple localized manifolds, and then fit multiple classifiers on features projected onto different localized manifolds, and finally combine the classifiers to provide the unified classification decision in order to reduce the bias caused by localization.

The rest of the chapter is organized as follows. Section 4.2 summarizes the proposed algorithm. In Section 4.3, the implementation details of localized manifolds are covered and the localization for two feature extraction methods, including a supervised method (L-NWFE) and a semi-supervised method (L-SELD) based on a novel weighting scheme are shown. In Section 4.4, the performance of the algorithm is evaluated by comparing with those based on a single global manifold using three standard hyperspectral datasets. Section 4.5 concludes the chapter by summarizing the above sections.

4.2 Overview of ensemble localized manifold learning algorithm

This section overviews the ensemble localized manifold learning (ELML), a joint feature extraction and classification method that captures the nonlinear structure using multiple linear manifolds. First, the feature space is partitioned into K clusters using a data clustering algorithm. Note that such a cluster is independent of the classes of training samples, so one cluster could have training samples of multiple classes. Then, localized manifolds are learned, each focused on one cluster. After K manifolds are learned, K sets of features are extracted using the manifolds. Afterwards, multiple base classifiers are trained on each set of features. All the classifiers are finally combined into a final classifier ensemble to improve classification performance. Fig. 4.1 shows a simulated example showing why multiple manifolds are better than a single manifold.

Algorithm 1 shows a high level algorithm for ELML. In Step 1, we use k-means to partition the data into K clusters due to its simplicity and fast convergence. Step 3 implements the localized manifolds and is the most important step in the ELML algorithm. We will focus on this step in the Section 4.3, and propose a weighting scheme to learn the manifolds for each cluster. In Step 4, the 1-nearest neighbor (1-NN) classifier is used as the base classifier. 1-NN is a very simple non-linear classifier with no tuning parameters. It is guaranteed to converge to an error rate less than twice the Bayes error when the number of samples approaches infinity [54]. NN-based classifiers are capable of capturing the variations of features, so they have been widely used to test the performance of feature extraction methods and achieve comparable performance to more complicated classifiers such as SVM [95, 165]. In the ELML algorithm, this property can help generate more diversity between the base classifiers. In Step 6, standard majority voting is used to aggregate the predictions by all the base classifiers.

4.3 Learning localized manifolds

From the theory of ensemble learning, the individual learners should have small bias and be diverse from each other [54]. Different from random sampling methods such as bootstrap sampling [35] and random subspace [60] commonly used in ensemble methods, the diversity of base classifiers is achieved by learning localized features in ELML. The key is thereby to learn the localized manifolds. If the manifold is learned only on a local cluster of data, it might be incapable of reflecting global data structure, and the bias will be thus increased.

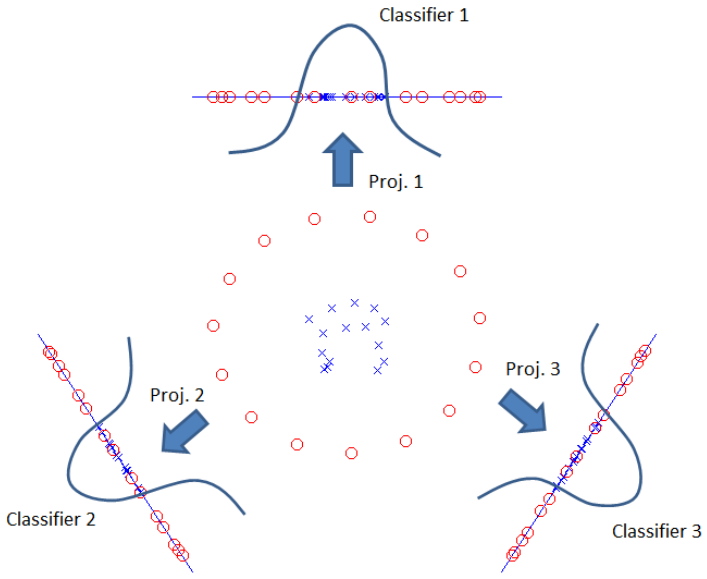


Figure 4.1: A simulated example showing why multiple manifolds are better than a single manifold. The red "o" and blue "x" represent two classes in two-dimension feature space. It is obvious that no linear dimension reduction method could project all the samples to one-dimension space to achieve projected separability. However, we can project them to three one-dimension manifolds, and fit three non-linear classifiers to each set of projected samples. Even though in each base classifier, some of the "o" are misclassified as "x", these misclassified samples can be finally corrected by majority voting.

Moreover, there might be insufficient training samples to learn a manifold in the local part. Therefore, in this chapter, each localized manifold is learned from all the training samples using a localization weighting scheme.

In this section, two existing localized methods are presented, including a supervised method (NWFE [79]) and a semi-supervised method (SELD [95]). These techniques are modified in this thesis by modifying them to be able to learn localized manifolds. Our new techniques are called localized NWFE (referred to as L-NWFE) and localized SELD (referred to as L-SELD). Similar to LDA, both methods seek a transformation so as to minimize the within-class scatter matrix and maximize the between-class scatter matrix:

$$W = \arg \max_{W \in \mathcal{R}^{d \times r}} \left[tr \left\{ (W^T S^{(w)} W)^{-1} W^T S^{(b)} W \right\} \right] \tag{4.1}$$

Algorithm 1 The ELML algorithm

Input: training data X, number of clusters K;
Output: A classifier ensemble $H(X)$;

1: Partition X into K clusters $\mathcal{C}_k$ $(k = 1, ..., K)$;
2: **for** $k = 1$ to K: **do**
3: Generate a manifold $\mathcal{M}_k$ that focuses on Cluster k: $\mathcal{M}_k \leftarrow \{X, \mathcal{C}_k\}$ $(k = 1, ..., K)$;
4: Generate a classifier h_k on data projected in the manifold: $h_k \leftarrow \mathcal{M}_k(X)$;
5: **end for**
6: Combine all the classifiers into the final classifier ensemble: $H(X) = \cup \{h_k(X)\}$ $(k = 1, ..., K)$.

where $S^{(w)}$ and $S^{(b)}$ are the within-class scatter and the between-class scatter in the original feature space, and d and r are the number of features in the original space and the new space respectively.

This optimization can be solved by eigen-decomposition, i.e., the extracted r features are the r eigenvectors associated with the largest r eigenvalues of $(S^{(w)})^{-1}S^{(b)}$. Therefore, the key of this category of algorithms is to define $S^{(w)}$ and $S^{(b)}$. In both proposed methods, new $S^{(w)}$ and $S^{(b)}$ are defined using a novel weighting scheme to incorporate localization, where sample points closer to the local clusters are assigned larger weights. The details of these two methods, L-NWFE and L-SELD, are described in the rest of this section.

4.3.1 Supervised localized feature extraction

Nonparametric weighted feature extraction

NWFE is a nonparametric feature extraction method based on scatter matrices. It is modified from the standard LDA by replacing the class mean with the local mean for each sample point. Moreover, the scatter matrices are weighted based on the Euclidean distances from sample points to their local means. More details of the original NWFE algorithm are provided by Kuo and Landgrebe [79].

Given L classes, the nonparametric between-class scatter matrix is defined as provided by Kuo and Landgrebe [79]:

$$S^{(b)} = \sum_{i=1}^{L} P_i \sum_{\substack{j=1 \\ j \neq i}}^{L} \sum_{l=1}^{N_i} \frac{\lambda_l^{i,j}}{N_i} (x_l^i - M_j(x_l^i)) \cdot (x_l^i - M_j(x_l^i))^T \tag{4.2}$$

$$S^{(w)} = \sum_{i=1}^{L} P_i \sum_{l=1}^{N_i} \frac{\lambda_l^{i,i}}{N_i} (x_l^i - M_i(x_l^i)) \cdot (x_l^i - M_i(x_l^i))^T \tag{4.3}$$

where x_l^i is a feature vector, N_i is the number of samples in class i, P_i is the prior probability of class i, and the nonparametric local weight $\lambda_l^{i,j}$ is defined as:

$$\lambda_l^{i,j} = \frac{dist(x_l^i, M_j(x_l^i))^{-1}}{\sum_{t=1}^{N_i} dist(x_t^i, M_j(x_t^i))^{-1}} \tag{4.4}$$

where $dist(\cdot, \cdot)$ is the Euclidean distance between two points. The local mean $M_j(x_l^i)$ of a sample x_l^i is the weighted mean of N_j sample points in class j

$$M_j(x_l^i) = \sum_{m=1}^{N_j} w_{lm}^{i,j} x_m^j \tag{4.5}$$

$$w_{lm}^{i,j} = \frac{dist(x_l^i, x_m^j)^{-1}}{\sum_{t=1}^{N_j} dist(x_l^i, x_t^j)^{-1}} \tag{4.6}$$

Localized nonparametric weighted feature extraction

To implement localization, A localized weight $\gamma_l^{i,k}$ is introduced for sample x_l^i on cluster $\mathcal{C}_k$, denoted as

$$\gamma_l^{i,k} = \exp\left\{ -\frac{dist^2(x_l^i, \mathcal{C}_k)}{\sigma^2} \right\} \tag{4.7}$$

where $dist(x_l^i, \mathcal{C}_k)$ denotes the Euclidean distance from x_l^i to cluster $\mathcal{C}_k$, and σ is a smoothness parameter, controlling the sensitivity of distance. When x_l^i is in $\mathcal{C}_k$, $dist(x_l^i, \mathcal{C}_k) = 0$, otherwise it is the shortest distance between x_l^i to any point that belongs to $\mathcal{C}_k$.

Therefore, the localized within-class and between-class scatter matrix are formulated as

$$S_k^{(b)} = \sum_{i=1}^{L} P_i \sum_{\substack{j=1 \\ j \neq i}}^{L} \sum_{l=1}^{N_i} \frac{\gamma_l^{i,k} \lambda_l^{i,j}}{N_i} (x_l^i - M_j(x_l^i)) \cdot (x_l^i - M_j(x_l^i))^T \tag{4.8}$$

$$S_k^{(w)} = \sum_{i=1}^{L} P_i \sum_{l=1}^{N_i} \frac{\gamma_l^{i,k} \lambda_l^{i,i}}{N_i} (x_l^i - M_i(x_l^i)) \cdot (x_l^i - M_i(x_l^i))^T \tag{4.9}$$

The localized weight σ is used to determine the degree of localization for learning manifolds. When $\sigma \to 0$, the manifold $\mathcal{M}_k$ is learned only from samples in $\mathcal{C}_k$, which is the strategy used in the LMNN-MM method [154]. When $\sigma \to +\infty$, the method is reduced to a single manifold learning algorithm.

4.3.2 Semi-supervised localized feature extraction

Semi-supervised local discriminant analysis

Semi-supervised local discriminant analysis (SELD) [95] is a category of semi-supervised feature extraction algorithms that linearly combines LDA and a LLFE method, such as LPP [57], neighborhood preserving embedding [56] (NPE), and linear local tangent space alignment (LLTSA) [175]. All these LLFE methods are designed to preserve the local neighborhood structure of data in the low-dimension feature space. LLFE can be expressed in a unified way:

$$W_{LLFE} = \underset{W}{\operatorname{argmax}} \frac{W^T X \overline{C} X^T W}{W^T X \underline{C} X^T W} \tag{4.10}$$

where $\overline{C}$ and $\underline{C}$ are both n×n matrices.

Here we only implement LPP [57], for which $\overline{C} = D$ and $\underline{C} = L$, where D is a n×n diagonal weight matrix, i.e., $D_{ii} = \sum_j A_{ji}$, and L is the Laplacian matrix, i.e., $L = D - A$, where A is the adjacency matrix. The implementation of NPE and LLTSA can be found in the original papers [95]. The only change is to use different $\overline{C}$ and $\underline{C}$.

To combine LDA with LLFE, the optimization problem of LDA is first reformulated into a similar form as LLFE [95]:

$$W_{LDA} = \underset{W}{\operatorname{argmax}} \frac{W^T X P X^T W}{W^T X (\overline{I} - P) X^T W} \tag{4.11}$$

where $\overline{I} = \begin{bmatrix} I_{n\times n} & 0 \\ 0 & 0 \end{bmatrix}$, $I_{n\times n}$ is an $n \times n$ identity matrix, $P = \begin{bmatrix} P_{n\times n} & 0 \\ 0 & 0 \end{bmatrix}$, and $P_{n\times n}$ is defined as:

$$P_{n\times n} = \begin{bmatrix} P^{(1)} & 0 & \cdots & 0 \\ 0 & P^{(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & P^{(L)}, \end{bmatrix} \tag{4.12}$$

and $P^{(l)}$ is the $N_l \times N_l$ matrix with all the elements equal to $\frac{1}{N_l}$.

Then, the optimization function of SELD is defined by linearly combining the covariance matrices of (4.10) and (4.11)

$$\begin{aligned} W_{SELD} &= \underset{W}{\operatorname{argmax}} \frac{W^T X \overline{A} X^T W}{W^T X \underline{A} X^T W} \\ &= \underset{W \in \mathcal{R}^{d\times r}}{\operatorname{argmax}} \frac{W^T X (P + \overline{C}) X^T W}{W^T X (\overline{I} - P + \underline{C}) X^T W} \end{aligned} \tag{4.13}$$

Localized semi-supervised local discriminant analysis

To localize SELD, we use a localized weight matrix $U^{(k)}$ related to cluster k with the same size of $\overline{S}_{SELD}$ and $\underline{S}_{SELD}$, whose elements $U^{(k)}_{i,j}$ representing the localized weight between two sample points x_i and x_j is defined as:

$$U^{(k)}_{i,j} = \exp\left\{ -\frac{dist^2(x_i, \mathcal{C}_k) + dist^2(x_j, \mathcal{C}_k)}{2\sigma^2} \right\} \tag{4.14}$$

where σ is a smoothness parameter as in Eq. (4.14).

Thus, the optimization problem of L-SELD is:

$$W^k_{L-SELD} = \underset{W}{\operatorname{argmax}} \frac{W^T X (\overline{A} \circ U^{(k)}) X^T W}{W^T X (\underline{A} \circ U^{(k)}) X^T W} \tag{4.15}$$

where $(\cdot \circ \cdot)$ is the Hadamard product that computes the element-wise product of two matrices.

4.4 Experiments

4.4.1 Experimental setup

The hyperspctral datasets for testing are described in Section 3.1. The classification performance using 1-NN classifiers on the original features and on extracted features by eight methods (i.e., PCA, LPP, LDA, NWFE, LFDA, SELD, L-NWFE, and L-SELD) are compared. For SELD and L-SELD, 1000 unlabeled samples are randomly selected for training. The single-manifold methods are applied to a single classifier and the multi-manifold methods are applied to multiple classifiers. For all the multi-manifold methods, the number of clusters in the k-means algorithm is fixed to 10. It has been tested that there is little improvement by setting the number of clusters to be greater than 10. The smoothness parameter σ is selected by grid search ($\sigma = 0, 2^0/10, 2^1/10, ..., 2^5/10$) and five-fold cross-validation using training samples. The number of extracted features ranges from 1 to 15. Two test cases are used in the experiment, in which there are 15 and 30 labeled training samples for each class respectively.

4.4.2 Experimental results and analysis

The experimental results by different feature extraction methods are in Fig. 4.2. LPP which is unsupervised and only dependent on local information is incapable of preserving separability of the original data. Even though supervised methods such as LDA and LFDA perform very well when there are sufficient training samples, they perform even worse than PCA when there are only 15 labeled samples per class. Though NWFE requires the estimation of the covariance matrix using labeled samples as well, its performance can be still maintained in the small-sample-size situation, which is consistent with the previous literature [78]. By using unlabeled training samples, SELD performs significantly better than both LDA and LPP. Also, both L-NWFE and L-SELD outperform NWFE and SELD respectively, especially when the number of extracted features is low. The highest OA achieved by each feature extraction methods with number of extracted features (maximum 15) is shown in Table 4.1. L-SELD achieves highest OA for both 15-sample and 30-sample cases in the University of Pavia and Kennedy Space Center datasets, and L-NWFE achieves highest OA for both cases in the Salinas dataset.

Finally, we test the sensitivity of the smoothness parameter σ. The relationship between σ and OA is shown in Fig. 4.3. All the OA reach the highest when σ is equal to some value between 0 and 3, and decrease slowly afterwards. Specifically, the classification performance

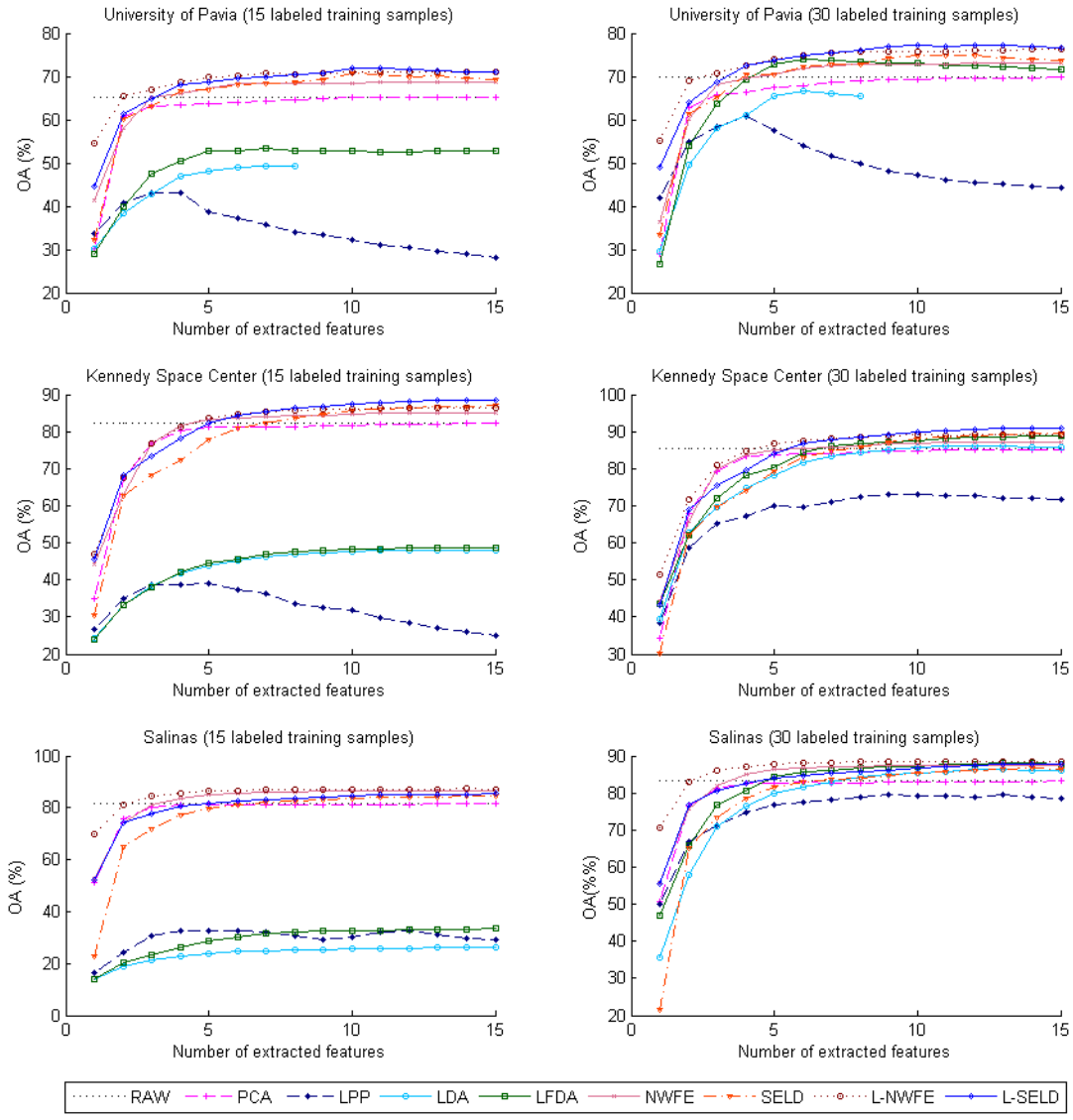


Figure 4.2: Classification accuracy (vertical axis) using 1-NN on different numbers of extracted features (horizontal axis) on three datasets for different number of labeled training samples. The dotted horizontal line tagged "RAW" is the classification accuracy using all the original features (103 the for University of Pavia dataset, 176 for the Kennedy Space Center dataset, and 200 for the Salinas dataset).

Table 4.1: Highest classification accuracy achieved for three datasets using 15 and 30 labeled training samples (with the corresponding number of extracted feature dimensions). "PU", "Salinas" and "KSC" represents the University of Pavia dataset, the Salinas dataset, and the Kennedy Space Center dataset respectively.

Methods	PU-15	KSC-15	Salinas-15	PU-30	KSC-30	Salinas-30
RAW	65.28 (103)	82.30 (176)	81.54 (200)	69.90 (103)	85.48 (176)	83.26 (200)
PCA	65.20 (15)	82.18 (14)	81.48 (15)	69.74 (15)	85.23 (15)	83.18 (15)
LPP	43.16 (3)	39.20 (5)	32.58 (4)	60.62 (4)	73.11 (10)	79.41 (9)
LDA	49.22 (7)	48.12 (12)	25.99 (14)	66.62 (6)	86.03 (13)	86.30 (13)
LFDA	53.46 (7)	48.66 (15)	33.43 (15)	74.03 (6)	88.81 (14)	87.98 (13)
NWFE	68.68 (15)	84.99 (15)	86.47 (12)	73.06 (15)	87.13 (15)	87.35 (13)
SELD	70.77 (10)	86.99 (15)	84.38 (15)	74.93 (10)	89.56 (15)	86.60 (14)
L-NWFE	71.15 (15)	86.43 (15)	**87.28** (14)	76.32 (15)	89.15 (14)	**88.52** (15)
L-SELD	**71.83** (11)	**88.57** (15)	85.31 (15)	**77.24** (12)	**91.05** (14)	87.90 (14)

of L-NWFE is relatively robust to the variation of σ, and the optimal performance is achieved when σ is close to 0. For L-SELD, OA reaches a peak around $\sigma = 1$ for all datasets. When σ becomes smaller, the manifolds are too focused on local structure and OA tends to decrease.

4.5 Summary

In this chapter, a joint feature extraction and classification method called ELML is proposed for hyperspectral imagery. Considering the complexity and nonlinearity of high-dimensional data structure, multiple linear localized manifolds are learned from the data. Then, multiple sets of features are extracted using these manifolds, and a classifier ensemble is trained on the features to obtain the final result. To implement ELML, L-NWFE and L-SELD are modified from NWFE and SELD using a localization weighting scheme in order to learn the localized manifolds, and the 1-NN classifier is used for classification. Experiments show that both L-NWFE and L-SELD compare favorably with respect to the referenced classification approaches when there are limited labeled training samples on three hyperspectral datasets.

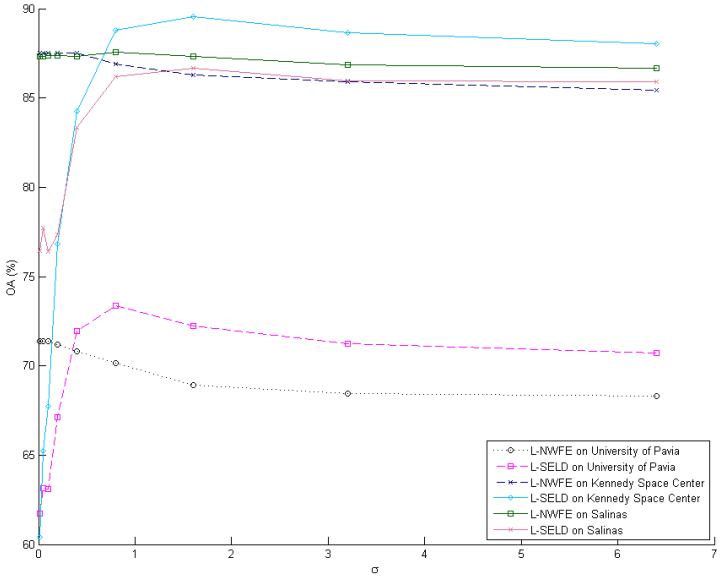


Figure 4.3: Classification accuracy (OA%) as a function of σ for both L-NWFE and L-SELD using 15 labeled samples per class and 15 extracted features on three datasets.

Chapter 5

Classification and segmentation using enhanced ensemble learning and conditional random fields

5.1 Introduction

This chapter investigates the performance of supervised classification methods in the situation of limited labeled training data, and especially focuses on ensemble methods. Ensemble methods have been successfully applied for hyperspectral image classification, such as random forests [50] and AdaBoost [74]. Compared to other classifiers, dimension reduction is usually unnecessary for ensemble methods because they deal fairly well with high-dimensional data [164]. In recent years, ensemble methods have been shown in particular to achieve high classification performance when the number of training samples is limited. Waske et al. [153] demonstrated that classifier ensembles using support vector machines and random feature selection can significantly improve classification performance. Yang et al. [163] proposed a dynamic subspace method for hyperspectral image classification which achieves better classification accuracy than the random subspace method. Xu et al. [162] investigated different classifiers for marine oil spill identification, and found that bagging-based methods significantly outperform other classification methods.

A recent trend of hyperspectral image classification research is the prevalence of spectral-spatial classification methods that incorporate spatial context to improve classification performance [115]. It has been demonstrated that global random field

methods are better than local filtering methods [126]. The most commonly-used random field model for remote sensing imagery is MRF [115]. Previous literature showed that the classification performance can be improved by combining MRF with a unary classifier that can generate probabilistic outputs [93, 101, 142]. In recent years, a discriminative random field model, i.e., the CRF model which can incorporate edge information has been used for remote sensing imagery [173, 177, 178].

This chapter addresses the training sample inadequacy issue by proposing a novel spectral-spatial hyperspectral image classification approach based on an enhanced ensemble classifier and the CRF technique. First, the proposed multiclass boosted rotation forest (MBRF) algorithm that integrates rotation forest and AdaBoost is used to obtain pixelwise estimation. The motivation is based on bias-variance analysis, i.e., the two ensemble classifiers have relative advantages in terms of decreasing model bias and model variance. Therefore, the combination of the two, as conducted in the proposed enhanced ensemble classifier, is able to take advantage of merits of both classifiers, especially in this small training sample size context. Second, to model the spatial contextual information in hyperspectral imagery, the proposed algorithm is incorporated into the CRF framework, serving as the unary term in the CRF objective function. Experiments on benchmark hyperspectral images demonstrate that the proposed MBRF algorithm is capable of outperforming other referenced state-of-the-art supervised classifiers, and is better able to aid the CRF approach for spectral-spatial classification of hyperspectral image, especially when the number of training samples is small.

The rest of this chapter is organized as follows. Section 5.2 introduces the bias-variance tradeoff for classification, and discusses the relative advantages and inadequacies of two mainstream ensemble methods, i.e., boosting and bagging from the perspective of the bias-variance trade-off, which motivates the proposed enhanced ensemble method. Section 5.3 introduces the proposed framework, including the details of the MBRF algorithm, its ability to approximate posterior probability estimates, and its incorporation into CRF to achieve spectral-spatial classification. Section 5.4 shows the comparison of the proposed method and several referenced state-of-the-art supervised classification methods on three hyperspectral datasets. Section 5.5 concludes the chapter by summarizing the above sections.

5.2 Bias-variance tradeoff for classification

Ensemble methods are computational techniques that combine a large number of base classifiers for improved prediction [10]. They have been widely used for supervised

classification due to their flexibility, ease of implementation, and outstanding performance.

The benefit of ensemble methods can be explained from a bias-variance decomposition perspective, which was originally proposed by Geman et al. for a regression model of squared loss [41]. Domingos [31] provided a unified bias-variance decomposition which can be applied to any loss function. The predicted loss $E_{D,y}[L(y,h)]$ for a given loss function L can be decomposed into intrinsic noise, bias and variance:

$$\begin{aligned} &E_{\mathcal{D},y}[L(y,h)] \\ &= c_1 E_y[L(y,h_*)] + L(h_*,h_m) + c_2 E_{\mathcal{D}}[L(h_m,h)] \\ &= c_1 N(x) + B(x) + c_2 V(x) \end{aligned} \tag{5.1}$$

where $\mathcal{D}$ is the training set, x is the example, y is the true value, h is the prediction, h_* is the optimal prediction that minimizes $E_y[L(y,h_*)]$, h_m is the main prediction [31] which is the mean of the predictions under squared loss, c_1 and c_2 are constants, and $N(x)$, $B(x)$, and $V(x)$ represent noise, bias, and variance respectively.

Since the noise is irreducible [54], we only consider the bias and the variance. The bias describes the error of the classifier in expectation, and the variance reflects the sensitivity of the classifier to variations in the training samples. For squared loss, $c_1 = c_2 = 1$, so both bias and variance increase the predicted loss. However, for zero-one loss used in classification problems, it has been demonstrated that c_2 is negative for biased examples, and therefore there is a much higher tolerance for variance [31, 101].

Ideally, we wish the classifiers to have both low bias and low variance. However, there is usually a tradeoff between bias and variance [54]. When the complexity of a classifier goes up, the bias tends to decrease while the variance will increase. Ensemble methods can largely reduce the variance by majority voting of the results by base classifiers, without affecting the bias or even reducing it [39]. Therefore, weak learners that are sensitive to small changes in data are selected as base classifiers, such as decision trees and perceptron [54].

Two mainstream ensemble approaches are boosting [38] and bagging [10]. The idea of boosting methods is to learn classifiers iteratively by adjusting the distribution of training samples based on the classification error, and predict labels by weighted majority voting. Both bias and variance can be reduced by boosting. Empirical experiments show that boosting is not easy to overfit [40], but when the sample size is small, boosting tends to have large variance and thus cause overfitting [99]. One of the most popular boosting methods is the AdaBoost algorithm [38]. It can be viewed as a forward stagewise additive modeling algorithm to minimize the exponential loss function.

The standard bagging is designed to reduce model variance by performing majority voting among base classifiers that are trained on bootstrap subsets of training samples. Compared to boosting, the bagging technique is more effective at reducing model variance, but the model bias is unchanged. In recent years, there are also some variants of the bagging method that have become popular. The random forest algorithm [11] is a method that combines bagging and random subspace method [60]. A reduction in the variance can be achieved by reducing the correlation between the trees, at the expense of a slight bias increase. Another typical method is the rotation forest algorithm [120], which performs feature transformation with some randomness on the data for each base classifier, and also combine the results by majority voting. Previous empirical experiments show that rotation forests have smallest variance compared to other ensemble methods such as boosting and random forests. Also, unlike bagging and random forests, the rotation can reduce bias even though the reduction of bias is not as significant as boosting [119].

Motivated by the above bias-variance analysis, an ensemble method integrating rotation forests and AdaBoost is proposed in this chapter so that both bias and variance can be further reduced by taking advantage of both methods. A similar approach to the proposed method is the RotBoost algorithm [172], but it is not developed for SSS context. Moreover, RotBoost adopts standard AdaBoost algorithm which requires the training error of the base classifiers to be less than 1/2, which is too strict for a multiclass classification problem. Instead, we use here a multiclass AdaBoost algorithm that is more tolerant about the training error. We also show that the proposed method allows the posterior probability to be naturally obtained without requiring the base classifiers to estimate probabilities. The posterior probability is used with the CRF framework to incorporate spatial context information. The implementation details will be introduced in the next section.

5.3 Proposed framework

In this section, we propose a two-stage framework for hyperspectral image classification with a limited number of training samples. The first stage is to perform ensemble learning to obtain posterior probability of class labels for pixels in hyperspectral image using only spectral information. The MBRF algorithm that combines rotation forests and AdaBoost is proposed for improving label prediction based on the motivation in Section 5.2. Using a multiclass AdaBoost algorithm, the posterior probability can be naturally generated without requiring the base classifiers to output probability estimates. In the second stage, based on the posterior probability, the proposed MBRF classifier is incorporated into the CRF framework, in order to simultaneously incorporate both spectral and spatial

information in hyperspectral imagery.

5.3.1 Multiclass boosted rotation forest

The MBRF method is a bagging-based method that combines multiple classifiers which are independent from each other. Instead of performing bootstrap sampling, the data is first perturbed by performing feature transformation with randomness, and then the classifier is trained by adaptive boosting. Therefore, the individual classifier for MBRF is not a single base classifier, but a boosted ensemble of base classifiers which is called meta-base classifier (MBC). Any rotation-variant classifier, i.e., the decision boundary will be changed by rotating the feature space of the data, can be used as base classifiers. The posterior probabilities by an MBC can be naturally approximated, and the probabilities by all the MBCs are finally combined. The flow chart of the MBRF algorithm is shown in Fig. 5.1.

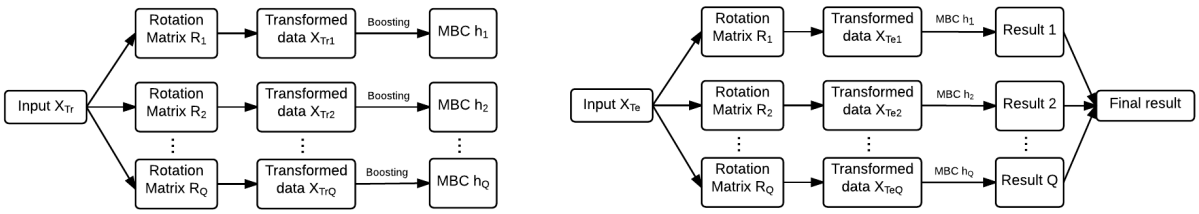


Figure 5.1: Flow charts of the proposed MBRF algorithm. The left is training stage and the right is test stage. Q rotation matrices and Q classifiers are learned based on training data X_{Tr} in the training stage. They are used to estimate the posterior probabilities of test data X_{Te} in the test stage.

To train a classifier h_i, the first step is to perturb the original data by multiplying by a rotation matrix. The procedure of calculating the rotation matrix is shown in Alg. 2. The original feature set is first randomly divided into Q subsets, and a random number of classes are eliminated to increase the randomness of the rotation matrix. Then a bootstrap sample of 75% sample size is selected [120]. Afterwards, a feature extraction method is performed on the bootstrap sample without reducing the dimensions. Empirical experiments show that PCA is the feature extraction method that can achieve best classification performance [77, 161]. The coefficients by PCA obtained for each subset are incorporated into a rotation matrix:

$$R = \begin{bmatrix} c_1^{(1)}, c_1^{(2)}, \cdots, c_1^{(M_1)} & \cdots & [0] \\ [0] & & \\ \vdots & \ddots & \vdots \\ [0] & \cdots & c_Q^{(1)}, c_Q^{(2)}, \cdots, c_Q^{(M_Q)} \end{bmatrix} \tag{5.2}$$

where Q is the number of subsets, M_i is the number of variables in each subset i ($i = 1...Q$), and $c_i^{(M_1)}, \cdots, c_i^{(M_i)}$ are $M_i \times 1$ coefficient vectors of principal components obtained from the bootstrap samples with variables in the i^{th} subset.

The columns in R are rearranged according to the order of the original feature set to obtain the final rotation matrix R^a.

Algorithm 2 Calculating rotation matrix R^a

1: Split the feature set $\mathbf{F}$ into Q subsets of features: F_i, $(i = 1, ..., Q)$;
2: **for** $i = 1...Q$ **do**
3: Let X_i be the dataset X for features in F_i;
4: Remove a random subset of classes from X_i;
5: Select a bootstrap sample of 75% sample size from X_i to form a new sample set X_i';
6: Apply PCA on X_i' to obtain the coefficients c_i^j, $(j = 1, ..., M_i)$;
7: **end for**
8: Construct R with the obtained coefficients using Eq. (5.2);
9: Output R^a by rearranging the columns of R by matching the order of features in F.

The second step is to perform AdaBoost on the rotated data. The original AdaBoost algorithm [38] is for binary classification. For multiclass problems, it may easily fail when the training error by base classifiers is greater than 1/2. One way to apply AdaBoost to a multi-class problem is to use one-versus-rest or one-versus-one strategies to decompose into multiple binary classification problems [179], such as AdaBoost.MH [125] and AdaBoost.M2 [124]. A disadvantage of these methods is the posterior probability cannot be directly generated. In the proposed method, we use a multiclass AdaBoost algorithm called SAMME [180]. It can be considered as forward stagewise additive modeling using a multiclass exponential loss function, which has the same statistical explanation as the original AdaBoost algorithm. Given K classes, it only requires the error rate by a base classifier to be less than $1 - 1/K$ rather than 1/2 which is very rigid for multiclass classification. We can also see that SAMME will reduce to AdaBoost when $K = 2$.

At the beginning, the training data are sampled from the training set $\mathcal{D}$ in the uniform distribution, i.e.,$\mathfrak{D}_1(\mathbf{x}) = 1/n$, where n is the number of training samples. Then, a base classifier h_t is trained on the sampled data, and the error ϵ^t is calculated:

$$\epsilon^t = p_{\mathbf{x_t} \sim \mathfrak{D}_t}(h_t(\mathbf{x}) \neq y) \tag{5.3}$$

where $\mathbf{x}$ represents the training samples and y represents their true labels.

The sampling distribution is updated in the way that misclassified samples are assigned larger weights:

$$\mathfrak{D}_{t+1}(\mathbf{x}) = \frac{1}{Z_t}\mathfrak{D}_t(\mathbf{x}) \cdot exp\{\ \alpha^t \mathbb{I}(h_t(\mathbf{x}) \neq y)\ \} \tag{5.4}$$

where Z_t is a normalization factor, $\mathbb{I}(\cdot)$ is the indicator function, and α_t is the model parameter based on the training error:

$$\alpha^t = log\frac{1-\epsilon^t}{\epsilon^t} + log(K-1) \tag{5.5}$$

For a new sample $\mathbf{x}$, the posterior probability $P(y = k \mid x)$ for each MBC can be approximated:

$$P(y = k \mid x) = \frac{e^{\frac{1}{K-1}} f_k^*(x)}{e^{\frac{1}{K-1}} f_1^*(x) + \cdots + e^{\frac{1}{K-1}} f_K^*(x)} \tag{5.6}$$

where

$$f_k^*(x) = \sum_{t=1}^{T} \alpha_t \cdot \delta(h_t(x), k) \tag{5.7}$$

and

$$\delta(h_t(x), k) = \begin{cases} 1 & h_t(x) = k \\ -\frac{1}{K-1} & \text{otherwise} \end{cases} \tag{5.8}$$

After the posterior probabilities for each MBC are obtained by SAMME, the final probability estimates are calculated by averaging over all the boosted classifiers. The

MBRF algorithm is described in Algorithm 3. Eq. (5.6) provides a way to obtain class probability with good theoretical meaning, so that it is not necessary for a single base classifier to generate probability estimates.

Algorithm 3 The MBRF algorithm

for $q = 1$ to Q: **do**
  Calculate the rotation matrix R_q^a as shown in Alg. 2;
  Perturb the data by multiplying the rotation matrix: $\mathbf{x_q} = \mathbf{x} \cdot R_q^a$;
  Initialize the weight distribution $\mathfrak{D}_{s1}(\mathbf{x_q}) = 1/n$;
  for $t = 1$ to T: **do**
    Train a classifier h_{qt} from the training set $\mathcal{D}$ under distribution $\mathfrak{D}_{qt}$: $h_{qt} = \mathfrak{L}(D, \mathfrak{D}_{qt})$;
    Update the sampling distribution $\mathfrak{D}_{q,t+1}$ using Eq. (5.4);
  end for
  Output conditional probability $P_q(y = k \mid \mathbf{x_q})$ using Eq. (5.6);
end for
Output final probability estimates by averaging:

$$P(y = k \mid \mathbf{x}) = \frac{1}{Q} \sum_{q=1}^{Q} P_q(y = k \mid \mathbf{x_q}).$$

5.3.2 Conditional random fields

The traditional MRF model is formulated in a probabilistic generative framework modeling the joint probability of the image and its labels [42, 92]. It assumes that a set of random variables have a Markovian property, which means the random variables are only dependent on their neighborhood. According to Bayes rule, the posterior probability is modeled as $P(\mathbf{y} \mid \mathbf{x}) \propto P(\mathbf{x} \mid \mathbf{y})P(\mathbf{y})$, where $\mathbf{y}$ is the labels and $\mathbf{x}$ is the observations. $P(\mathbf{y})$ is modeled as a Gibbs distribution. $P(\mathbf{x} \mid \mathbf{y})$ can be represented as a factorized form if we assume the conditional probability $P(x_i \mid y_i)$ is independent:

$$P(\mathbf{x} \mid \mathbf{y}) = \prod_i P(x_i \mid y_i) \tag{5.9}$$

Contrary to MRF, CRF [81] discriminatively models the posterior probability directly so that the rigid conditional independence assumption can be relaxed:

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} exp \left\{ - \sum_{c \in C} \psi_c(\mathbf{y_c}, \mathbf{x}) \right\} \tag{5.10}$$

where $Z(\mathbf{x}) = \sum_{\mathbf{y}} exp \{ - \sum_{c \in C} \psi_c(\mathbf{y_c}, \mathbf{x}) \}$ is the partition function and ψ_c is a potential defined on clique c.

For simplification, only unary and pairwise clique potentials are usually considered. Eq. (5.10) can be rewritten as

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} exp \{ - [\sum_{i \in S} \phi_i(y_i, \mathbf{x}) + \sum_{i \in S} \sum_{j \in \eta_i} \xi_{ij}(y_i, y_j, \mathbf{x})] \} \tag{5.11}$$

where $\phi_i(\cdot)$ and $\xi_{ij}(\cdot)$ are unary and pairwise clique potentials respectively, η_i is the set of neighbors of site i, and S is the set of integers indicating the discrete rectangular lattice.

In the proposed method, an 8-connected CRF is used. The unary and pairwise clique potentials can be defined as arbitrary domain-specific local discriminative classifiers [177]. In previous remote sensing literature, the unary potentials have been defined as posterior probabilities by various discriminative classifiers such as multinomial logistic regression [173, 176] and support vector machines [142, 178]. In the proposed method, we use the probability estimates by the proposed MBRF method as shown in the previous section. Thus, the unary potential can be defined as

$$\phi_i(y_i, \mathbf{x}) = \sum_{k=1}^{K} \delta(y_i = k) \{ - log P(y_i = k \mid \mathbf{x_i}) \} \tag{5.12}$$

where $P(y_i = k \mid \mathbf{x_i})$ is calculated using Eq. (5.6).

For the pairwise potentials, the standard MRF model only allows the contextual information of the labels to be used (i.e., the standard Potts model [116]), while both the labels and the observed data can be formulated in the CRF model as $\xi_{ij}(y_i, y_j, \mathbf{x})$ in Eq. (5.11). We note that pairwise connected terms will tend to have different labels at discontinuities in image structure. As a result, we use a generalized Potts model as the discontinuity preserving smoothness constraint:

$$\xi_{ij}(y_i, y_j, \mathbf{x}) = \begin{cases} \beta exp\{-\alpha(I_i^e + I_j^e)/2\} & y_i \neq y_j \\ 0 & \text{otherwise} \end{cases} \tag{5.13}$$

where i and j are indices of two neighboring pixels, I^e is the edge image obtained using the Gaussian derivative per band then per pixel maximum over all the bands, $\alpha = 1/(0.25 \cdot T_{otsu})$, T_{otsu} is the Otsu Threshold of the edge image I^e which will adapt the edge strength based on global edge strength of the image [133], and β is a weight parameter representing the degree of smoothness.

In practice, the optimum β is usually selected using cross validation [178]. In the inference step, the optimal labeling is assigned by maximizing the posterior probability in Eq. (5.11), i.e., to solve the energy minimization problem below:

$$\underset{y}{\operatorname{argmin}} \sum_{i \in S} \phi_i(y_i, \mathbf{x}) + \sum_{i \in S} \sum_{j \in \eta_i} \xi_{ij}(y_i, y_j, \mathbf{x}) \tag{5.14}$$

In previous literature, loopy belief propagation (LBP) [103] has been used to solve this combinatorial optimization problem [177, 178], but recently graph-cut based methods have become popular. For binary labeling problem, the graph-cut method can find the global optimum [48]. Boykov et al. [9] developed an efficient graph-cut based algorithm, i.e., α-expansion and α-β-swap algorithms, which are able to find an approximate solution to the multiclass labeling problem. The α-expansion algorithm has been demonstrated to outperform other state-of-the-art energy minimization methods on benchmark problems [141]. Furthermore, it has been proved [9] that a local minimum within a known factor of the global minimum can be found using α-expansion. As a result we use the α-expansion algorithm to solve the energy minimization problem.

5.4 Experiments

In this section, experiments are conducted for testing the performance of the proposed MBRF method and the combination with CRF. First, MBRF is compared with several state-of-the-art supervised pixelwise classification methods, including SVM, random forests (RF), SAMME, and rotation forests (RoF). SVM is one of the most commonly-used classifier for remote sensing imagery. RF and SAMME are advanced versions of bagging and standard AdaBoost respectively. Then, the combination of MBRF and CRF without using edge strength (MBRF-CRF-NE) and with edge strength (MBRF-CRF-E) is tested with two recently proposed spectral-spatial methods: SVMMRF-E [142] and MLR-CRF-E [173]. It is noted that SVMMRF-E is actually a CRF-based method because it uses posterior probabilities by a discriminative classifier and uses edge strength in the pairwise potentials.

The hyperspctral datasets for testing are described in Section 3.1. The main objectives of this section are testing and evaluating the performance of the aforementioned pixelwise and spectral-spatial classification methods in different numbers of training samples for the three datasets.

5.4.1 Experimental setup

For SVM, the radial basis function (RBF) is used as the kernel function, and the optimum parameters, i.e., the regularization parameter C and the bandwidth parameter of the RBF kernel γ are found using 5-fold cross-validation. The number of trees is set to 500. The number of variables randomly selected at each split is set by default, i.e., the square root of the total number of variables. For SAMME, the number of base classifiers are set to 100. For RoF and MBRF, the number of classes eliminated from the original data to calculate the rotation matrix is fixed to 3. In previous literature, the number of trees in the rotation forest algorithm is usually set to 10 [120,161]. Considering that the SSS problem might lead to slow convergence, the number of trees in rotation forests is set to 50. For MBRF, the number of MBCs is set to 30, and the number of trees in an MBC is set to 20. Decision tree classifier is used as the base classifier for all the ensemble methods. Due to the incapability of generating posterior probabilities directly by SVM, pairwise coupling [158] used in [142] is also adopted in the experiment.

We use the same setting for the number of training samples as [68]. Different numbers of randomly selected training samples per class (3, 5, 10, and 15) are used for testing. The overall accuracy (OA), average accuracy (AA), and kappa coefficient are calculated to evaluate the classification performance. We also notice that the classification performance is very sensitive to the selection of training samples when the number of training samples is limited, so all the methods are tested for 50 times using different randomly selected training samples, and the mean is used for all the statistics. For the weight parameter β in the CRF model, the traditional cross-validation is usually incapable of selecting the optimal parameter because the number of training samples is limited. Therefore, we conduct multiple tests using different β $(2^0, 2^1, 2^2, ..., 2^8)$ and report the highest test accuracy. To investigate the usefulness of the edge penalty information used in the smoothness constraint, we make a comparative study between the CRF model using edge penalty in Eq. 6.9 and without using edge penalty (i.e., $\alpha \to +\infty$).

5.4.2 Experiments: University of Pavia dataset

The first dataset for testing was acquired by the ROSIS sensor in University of Pavia, Italy. The classification result of the University of Pavia dataset is shown in Table 5.1. It is observed that MBRF achieves best classification performance for different number of labeled training samples. When there are only three samples per class, MBRF generally achieves about 5.5% OA higher than SVM, 9.2% higher than RF, 16.0% higher than SAMME, and 2.4% higher than RoF. The classification accuracy increases more than ten percent on average by combining with CRF. The classification accuracy of each class in the case of 10 training samples per class is shown in Table 5.2. MBRF has highest classification accuracy in five out of nine classes and has the highest average accuracy. The confusion matrix using MBRF-CRF-E with 10 random-selected samples per class in one test is shown in Table 5.3.

Among the spectral-spatial methods, MBRF-CRF-E achieves highest classification accuracy in all the cases. Also, we can see that CRF with edge penalty is significantly better than that without edge penalty, with an improvement of 5.3% classification accuracy on average. Fig. 5.2 shows the segmentation result using MBRF and CRF with different number of training samples in one test. The object boundaries are well-delineated due to the high spatial resolution and the urban scene of the dataset, which can help extract a clean edge map for the CRF model. Using edge penalty, neighboring pixels with strong edges can be prevented to be assigned the same label. Meanwhile, higher weight parameter can be selected so that pixels with high within-class variation can be smoothed out in the labeling.

5.4.3 Experiments: Salinas dataset

The second dataset for testing was acquired by AVIRIS Salinas dataset over Salinas Valley, California. The classification result of the Salinas dataset is shown in Table 5.4. The overall accuracy achieved by MBRF is 1.5% higher than SVM, 3.8% higher than RF, 10.0% higher than SAMME, and 1.7% higher than RoF on average.

The classification results by different methods using 10 samples per class in one test is shown in Fig. 5.3. Due to the low separability between the Vinyard-untrained class and the Grapes-untrained class, it is not enough to achieve satisfactory classification for these two classes by using only spectral information. In this case, the CRF model which serves as a smooth labeling method can help improve classification significantly. MBRF-CRF-E increase OA by 10.7% in all the cases compared to the pixelwise MBRF method, and

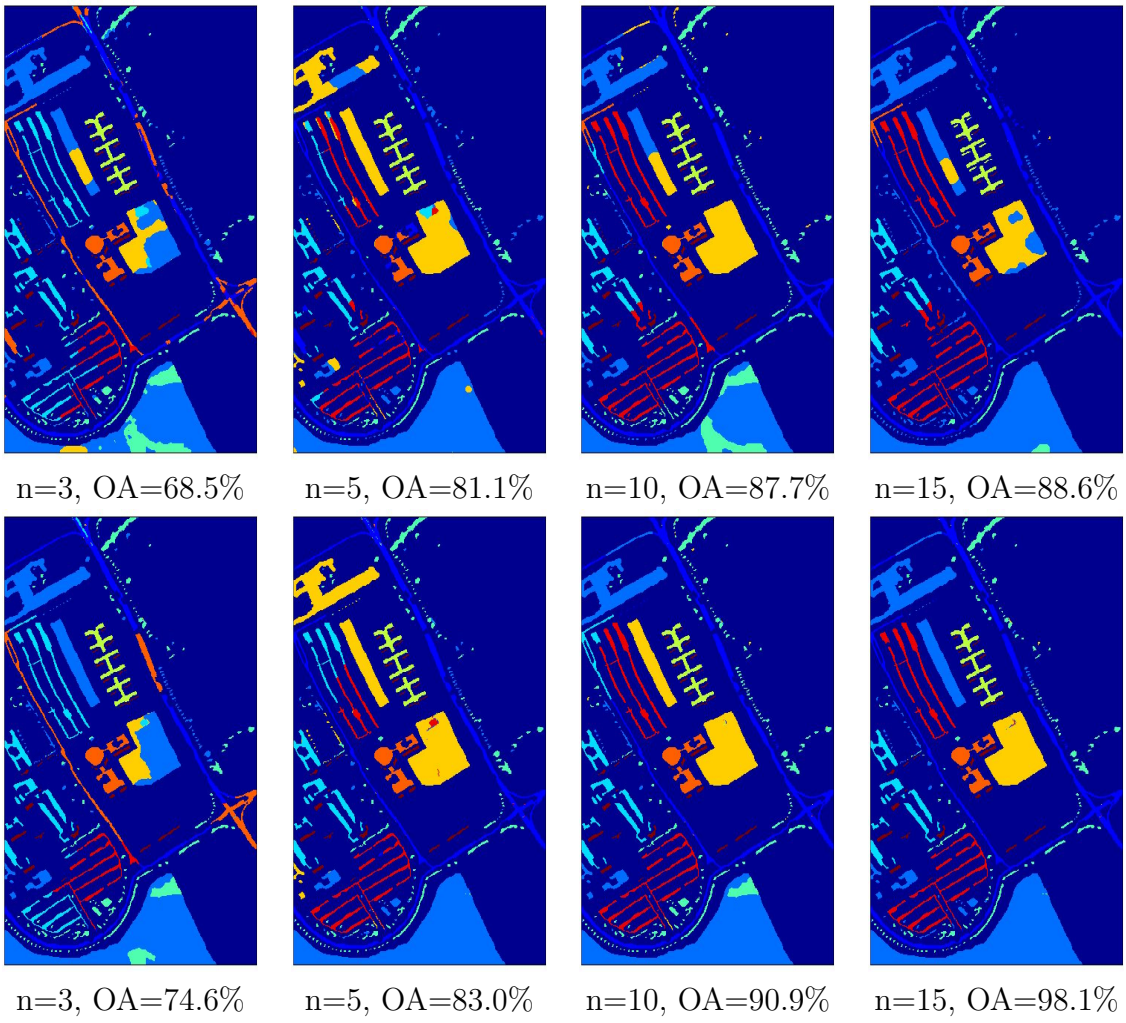


Figure 5.2: Segmentation map by MBRF-CRF-NE (top) and MBRF-CRF-E (bottom) in different number of training samples (n per class) with optimal weight parameter in the University of Pavia dataset. Results in the same column are based on the same probability outputs by MBRF.

Table 5.1: Overall accuracy (OA%) and average accuracy (AA%) , and kappa coefficient ([κ]) of different number of training samples by different methods for the University of Pavia dataset. The first five columns are results of pixelwise classifiers, and the last four columns are results refined by random fields. Bold identifies the highest OA among all the methods.

Classifier	Labeled samples per class			
	3	5	10	15
SVM	56.6 (68.0) [0.47]	60.0 (71.4) [0.51]	68.9 (77.3) [0.61]	73.8 (80.7) [0.67]
RF	56.7 (68.1) [0.47]	59.8 (70.6) [0.51]	62.4 (73.7) [0.54]	65.5 (75.6) [0.57]
SAMME	42.6 (51.4) [0.32]	53.2 (62.9) [0.43]	58.6 (69.3) [0.49]	62.8 (72.7) [0.54]
RoF	59.2 (67.9) [0.50]	64.1 (73.9) [0.56]	71.9 (79.3) [0.65]	76.5 (82.5) [0.70]
MBRF	**61.6** (71.9) [0.53]	**67.6** (76.2) [0.60]	**73.7** (80.7) [0.67]	**78.3** (84.0) [0.72]
SVMMRF-E	52.2 (61.3) [0.43]	66.0 (71.5) [0.58]	83.9 (86.1) [0.79]	90.8 (92.4) [0.88]
MLR-CRF-E	71.2 (76.2) [0.64]	79.6 (82.6) [0.74]	88.1 (88.5) [0.85]	91.7 (92.3) [0.89]
MBRF-CRF-NE	69.1 (75.9) [0.61]	78.2 (82.7) [0.72]	85.6 (88.6) [0.82]	90.2 (92.5) [0.87]
MBRF-CRF-E	**75.2** (80.3) [0.69]	**83.7** (86.9) [0.79]	**90.9** (92.9) [0.88]	**94.6** (95.9) [0.93]

achieves higher OA than other spectral-spatial methods. Instead, MBRFCRF-NE does not consider the edge between the Vinyard-untrained class and the Grapes-untrained class, and only relies on the unary classifier which is incapable of separating these two classes. The confusion matrix using MBRF-CRF-E with 10 random-selected samples per class in one test is shown in Table 5.5.

5.4.4 Experiments: Kennedy Space Center dataset

The Kennedy Space Center dataset was acquired by the AVIRIS instrument over the Kennedy Space Center, Florida on March 23, 1996 [50]. The classification result is shown in Table 5.6. Compared to the two datasets, this dataset is relatively easy because the pixelwise classification can achieve over 90% accuracy by MBRF using only 15 training

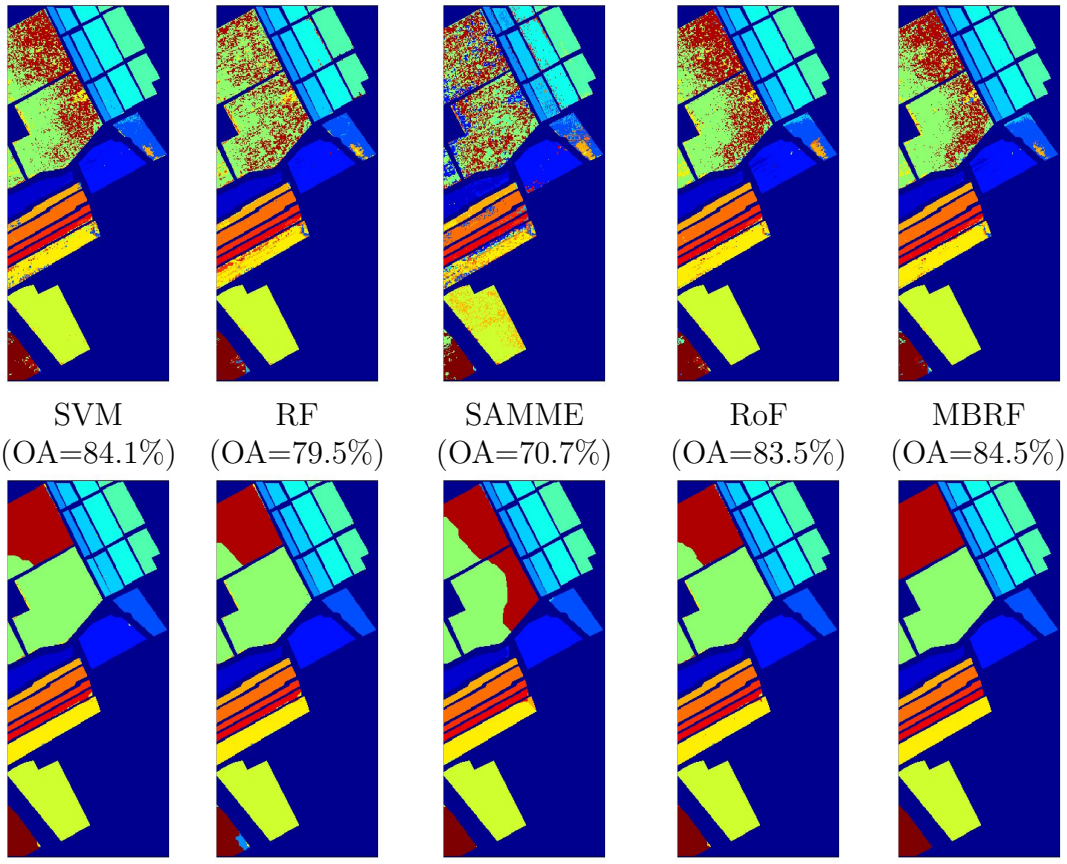


Figure 5.3: Classification maps by different methods using 10 labeled training samples per class on the Salinas dataset in one test.

Table 5.2: Class accuracy (%) achieved by different pixelwise classifiers for the University of Pavia dataset in the case of 10 training samples per class. "C1", "C2", ...,"C9" represent the nine classes orderly in the legend of Fig. 3.1. Bold identifies the highest accuracy among all the methods.

Classifier	C1	C2	C3	C4	C5	C6	C7	C8	C9	AA
SVM	65.9	64.3	58.5	90.5	99.1	58.4	85.8	73.2	**99.9**	77.3
RF	65.0	53.4	52.3	87.1	99.0	50.1	82.7	70.6	**99.9**	73.4
SAMME	62.2	49.7	50.0	81.2	97.1	52.6	77.1	66.4	91.4	69.8
RoF	67.6	67.3	61.5	91.0	99.0	66.9	83.8	**76.0**	99.8	79.2
MBRF	**70.4**	**70.0**	67.4	91.9	99.4	**68.7**	**87.2**	75.5	**99.9**	**81.2**

Table 5.3: Confusion matrix using the MBRF-CRF-E algorithm for the University of Pavia dataset with 10 randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.1.

	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	3614	0	6	0	2	2998	1	0	0
C2	0	12515	0	772	0	5352	0	0	0
C3	0	0	2087	0	0	2	0	0	0
C4	77	175	0	2375	1	414	0	7	5
C5	0	0	0	0	1327	7	0	1	0
C6	0	3	1	0	2	5012	0	1	0
C7	1	0	0	0	0	0	1319	0	0
C8	6	0	0	0	0	120	0	3546	0
C9	0	0	0	0	0	0	17	0	920

samples for each class. Considering the classification accuracy for all the cases, MBRF averagely outperforms SVM by 2.8%, RF by 9.2%, SAMME by 13.2%, and RoF by 2.7%.

After combining with CRF, the highest overall accuracy using 15 training samples per class can reach 98% by MBRF-CRF-NE. There is no improvement of using the edge penalty in the CRF model. Compared to the first two datasets, this dataset has relatively low spatial resolution. Also, there is not much strong edge information in the rural area of the image where most of the training areas are located. As shown in Fig. 5.4, both methods achieve similar classification accuracy inside the training areas. However MBRF-CRF-NE tends to oversmooth over the image, while MBRF-CRF-E is better at preserving details, especially in small regions with strong boundary information. The confusion matrix using MBRF-CRF-E with 10 random-selected samples per class in one test is shown in Table

Table 5.4: Overall accuracy (OA%) and average accuracy (AA%) , and kappa coefficient ([κ]) of different number of training samples by different methods for the Salinas dataset. The first five columns are results of pixelwise classifiers, and the last four columns are results refined by random fields. Bold identifies the highest OA among all the methods.

Classifier	Labeled samples per class			
	3	5	10	15
SVM	76.9 (84.4) [0.74]	80.3 (87.9) [0.78]	83.3 (90.6) [0.82]	85.0 (91.9) [0.83]
RF	75.6 (83.5) [0.73]	77.9 (85.9) [0.76]	80.8 (88.3) [0.79]	82.1 (89.5) [0.80]
SAMME	64.2 (71.4) [0.61]	71.2 (77.9) [0.68]	76.2 (83.7) [0.74]	79.9 (87.3) [0.78]
RoF	76.3 (83.4) [0.74]	80.0 (87.9) [0.78]	83.5 (90.6) [0.82]	84.8 (91.7) [0.83]
MBRF	**79.8** (87.3) [0.78]	**81.2** (88.8) [0.79]	**84.6** (91.3) [0.83]	**86.1** (92.6) [0.85]
SVMMRF-E	78.1 (82.9) [0.76]	85.0 (91.3) [0.83]	91.4 (95.1) [0.90]	95.6 (97.1) [0.95]
MLRCRF-E	84.6 (90.6) [0.83]	**90.2** (93.4) [0.89]	94.4 (96.3) [0.94]	97.1 (98.1) [0.97]
MBRFCRF-NE	87.0 (92.4) [0.86]	87.0 (92.7) [0.86]	92.2 (93.6) [0.91]	95.6 (97.5) [0.95]
MBRFCRF-E	**90.5** (94.3) [0.89]	89.4 (91.8) [0.88]	**96.2** (96.0) [0.96]	**98.1** (98.6) [0.98]

5.7.

5.4.5 Summary of classification results and sensitivity analysis

Observed from the pixelwise classification results (Table 5.1, Table 5.4, and Table 5.6), the overall classification accuracy by SAMME is not satisfactory because it is prone to overfitting when there are limited training samples. RF is less prone to overfitting because it reduces model variance, however, it cannot reduce bias of the decision tree classifiers. SVM is slightly better than RF, especially in the cases of $n = 10$ and $n = 15$. However, SVM is sensitive to the selection of model parameters, which are often determined by cross validation. When the number of training samples is limited, the parameters that achieve best cross-validation performance are less likely to be the optimal parameters. This could

Table 5.5: Confusion matrix using the MBRF-CRF-E algorithm for the Salinas dataset with 10 randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.2 in order.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16
C1	1999	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C2	0	3716	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C3	0	0	1966	0	0	0	0	0	0	0	0	0	0	0	0	0
C4	0	0	2	1382	0	0	0	0	0	0	0	0	0	0	0	0
C5	0	0	4	30	2632	0	0	0	0	1	1	0	0	0	0	0
C6	0	10	0	0	0	3939	0	0	0	0	0	0	0	0	0	0
C7	0	0	0	0	3	0	3554	9	0	0	0	0	0	1	1	1
C8	0	0	0	0	0	0	0	11094	0	163	1	0	0	2	1	0
C9	0	0	0	0	0	0	0	0	6193	0	0	0	0	0	0	0
C10	0	0	0	3	0	0	0	0	47	3204	0	0	0	12	0	2
C11	0	0	0	3	0	0	0	0	0	0	0	1055	0	0	0	0
C12	0	0	0	0	0	0	0	0	0	0	0	1917	0	0	0	0
C13	0	0	0	0	0	0	0	0	0	0	0	892	0	14	0	0
C14	0	0	0	0	0	0	0	1	0	29	0	0	0	1030	0	0
C15	0	0	3	32	1	0	0	2	0	2	0	0	0	0	7218	0
C16	0	0	0	0	1	0	0	1	0	13	0	0	0	0	0	1782

be improved using unsupervised heuristics or semi-supervised heuristics. The rotation forest classifier which reduces both bias and variance perform well very in SSS problems. It achieves higher overall classification accuracy than other comparing methods except MBRF in most of the cases. MBRF achieves the highest OA, AA, and kappa coefficient for all the cases, and it gains additional improvement of 2.5% classification accuracy over RoF. This is because the combination of RoF with SAMME can further reduce model bias. Moreover, the high variance and overfitting drawback of boosting in SSS problem turns into a benefit because it can de-correlate the MBCs from each other, and thus increase diversity.

From the spectral-spatial classification results, we observe that OA achieved by SVMMRF-E is not satisfactory in the cases of $n = 5$ and $n = 10$, which is even worse than the pixelwise SVM result. The reason might be that the pairwise coupling method [158] fails to generate satisfactory posterior probabilities when there are insufficient training samples. Instead, MLR is a classifier that learns posterior probabilities discriminatively, and results show that the OA achieves by MLR-CRF-E is achieved better than that by SVMMRF-E. Similarly, MBRF allows the natural approximation of posteriors. We can see that the highest classification accuracy is achieved by either MBRF-CRF-E or MBRF-CRF-NE in all the datasets. Also, MBRF-CRF-E achieves higher classification methods than MBRF-CRF-NE in the University of Pavia dataset which has high spatial resolution and strong edge strength.

Compared to the rotation forest classifier, there is one more parameter to determine in the MBRF method, i.e., the number of trees T for the embedded SAMME algorithm.

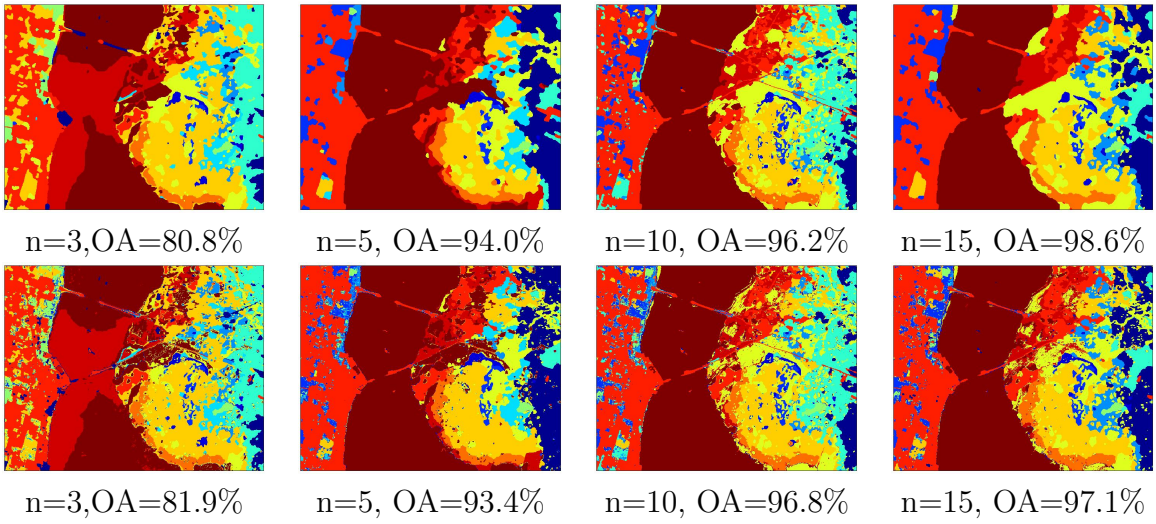


Figure 5.4: Segmentation map by MBRF-CRF-NE (top) and MBRF-CRF-E (bottom), in different number of training samples (n per class) with optimal weight parameter β in the Kennedy Space Center dataset. We can see that details are more preserved in the results using edge penalty. Results in the same column are based on the same probability outputs by MBRF.

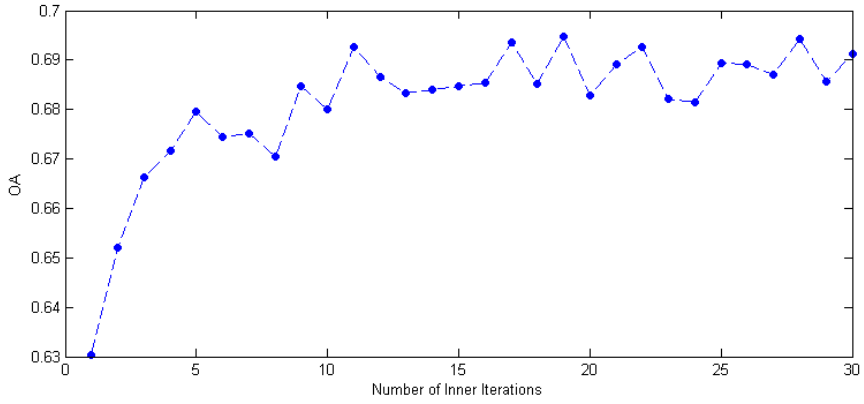


Figure 5.5: Overall classification accuracy (OA%) as a function of the number of inner iterations T in MBRF for the University of Pavia dataset using different randomly-selected training samples (5 per class).

Table 5.6: Overall accuracy (OA%) and average accuracy (AA%) , and kappa coefficient ([κ]) of different number of training samples by different methods for the Kennedy Space Center dataset. The first five columns are results of pixelwise classifiers, and the last four columns are results refined by random fields. Bold identifies the highest OA among all the methods.

Classifier	Labeled samples per class			
	3	5	10	15
SVM	73.1 (67.5) [0.70]	79.1 (74.2) [0.77]	86.0 (82.2) [0.84]	88.4 (85.0) [0.87]
RF	68.5 (62.2) [0.65]	73.5 (68.0) [0.70]	78.1 (73.6) [0.76]	80.9 (76.5) [0.79]
SAMME	56.0 (51.2) [0.51]	69.2 (64.5) [0.66]	77.9 (73.7) [0.75]	81.9 (77.8) [0.80]
RoF	73.2 (67.6) [0.70]	80.0 (75.3) [0.78]	85.8 (81.8) [0.84]	88.2 (84.4) [0.87]
MBRF	**78.0** (72.7) [0.76]	**82.7** (78.2) [0.81]	**87.5** (83.6) [0.86]	**89.8** (86.2) [0.89]
SVMMRF-E	68.1 (62.0) [0.65]	82.8 (78.8) [0.81]	92.7 (91.0) [0.92]	95.2 (94.2) [0.95]
MLR-CRF-E	84.3 (81.3) [0.82]	89.2 (87.0) [0.88]	94.4 (93.0) [0.94]	96.3 (95.2) [0.96]
MBRF-CRF-NE	**87.8** (84.5) [0.86]	**91.9** (89.3) [0.91]	**96.2** (94.7) [0.96]	**98.0** (96.7) [0.98]
MBRF-CRF-E	86.8 (84.4) [0.85]	90.9 (88.4) [0.90]	95.3 (94.0) [0.95]	97.3 (96.2) [0.97]

When $T = 1$, MBRF is reduced to RoF. Fig. 5.5 shows the overall classification accuracy as the number of trees increases for the University of Pavia dataset. The number of outer iterations Q is fixed to 20, the same as the previous experimental setting. As shown in Fig. 5.5, the classification performance is improved as T increases, and the accuracy becomes stable when T is greater than 15.

For the computation time, MBRF requires to train T base classifiers in an outer iteration compared to RoF, but the computational cost is still low considering the small training sample size, and the testing speed is very fast if the decision tree classifier is used. Compared to other ensemble methods such as bagging and boosting, MBRF might be slower due to the innate PCA step. The computational complexity of PCA is $O(D^3)$ if the eigendecomposition of the $D \times D$ covariance matrix is performed using a power method [148]. But in the MBRF, D is only a small subset of features, which is set to three

Table 5.7: Confusion matrix using the MBRF-CRF-E algorithm for the Kennedy Space Center dataset with 10 randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.3 in order.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
C1	727	0	0	0	0	24	0	0	0	0	0	0	0
C2	0	226	0	1	0	4	0	1	1	0	0	0	0
C3	0	0	240	1	0	0	0	4	1	0	0	0	0
C4	1	0	47	104	7	78	0	4	0	0	0	1	0
C5	0	0	0	61	85	2	0	3	0	0	0	0	0
C6	0	0	0	7	0	212	0	0	0	0	0	0	0
C7	0	0	0	0	0	0	95	0	0	0	0	0	0
C8	0	0	0	3	0	0	0	416	0	0	0	1	1
C9	0	0	0	0	0	0	0	12	498	0	0	0	0
C10	0	0	0	0	0	0	0	0	0	393	0	1	0
C11	0	0	0	0	0	0	0	1	0	0	408	0	0
C12	1	2	0	0	0	0	0	5	0	2	4	443	36
C13	0	0	0	0	0	0	0	0	0	0	0	0	917

in the experiment. Therefore, the PCA does not increase computational cost very much.

Finally, we test the sensitivity of the weight parameter. The test accuracy related to different number of training samples for all the datasets is shown in Fig. 5.6. It is observed that the weight parameter for CRF with edge penalty is less sensitive than that without edge penalty. Also, the optimal weight parameter varies for different number of training samples. Based on the observations, it slightly increases when there are more training samples. One explanation is that the spatial context is only helpful when the probability estimates are reliable. If a majority of pixels in a region are misclassified, random fields or any other type of smooth labeling methods tend to make the result worse.

5.5 Summary

A spectral-spatial classification method was proposed in this chapter to deal with the situation when there are limited labeled training samples available. It is based on a novel ensemble method combining rotation forests and multiclass AdaBoost. The classification performance can be enhanced by combining both methods because the rotation forest algorithm reduces model variance while the AdaBoost algorithm reduces model bias. Also, we showed that the posterior probability can be naturally approximated by the

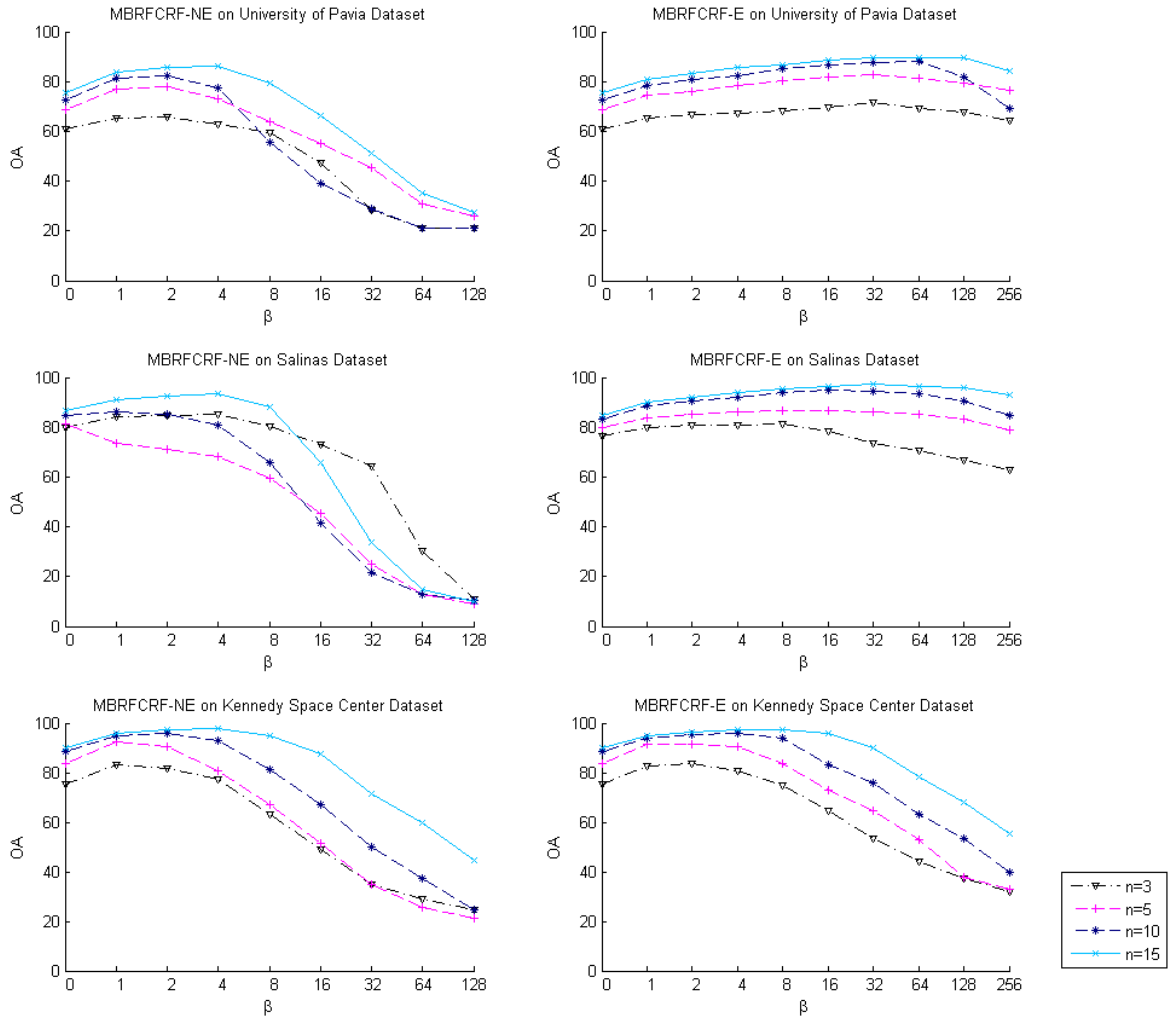


Figure 5.6: Overall classification accuracy (OA%) for different weight parameter (β) and different number of training samples (n per class) in three datasets. The results in the left column are created by CRF without edge penalty, and the results in the right column are created by CRF with edge penalty.

proposed MBRF method and incorporated into the CRF framework. Experimental results showed that MBRF as well as its combination with CRF outperforms other state-of-the-art supervised classification methods when the number of labeled training samples is limited.

Chapter 6

ST-IRGS: a semi-supervised self-learning system for segmentation and classification of hyperspectral imagery

6.1 Introduction

In recent years, semi-supervised classification methods have been developed for hyperspectral image classification, which improve the performance of classifiers by incorporating unlabeled training samples [14, 15, 21, 33, 89, 152]. Most of these semi-supervised methods use a regularized term which encourages unlabeled samples that are close to each other in the feature space to be assigned the same label. Due to the regularization, the decision boundary can be adjusted to cross regions in low density. However, empirical experiments have shown that semi-supervised methods make no improvement or even detrimental impact on classification performance [181]. Moreover, many semi-supervised methods usually have high computational cost, especially when the size of unlabeled training samples is large. Further, the above semi-supervised approaches either do not consider spatial context, or use spatial context models as a post-processing step to refine the classification results, which make no improvement on parameter estimation for training models.

In this chapter, a region-based semi-supervised classification and segmentation algorithm called self-training iterative region merging using semantics (ST-IRGS) as

well as its application for hyperspectral imagery is presented. ST-IRGS is a semi-supervised extension of the IRGS algorithm [170] which is an unsupervised classification and segmentation segmentation algorithm that integrates hierarchical region merging in a spatial context model, and has been successfully applied to SAR sea ice imagery [169] and later to polarmetric SAR imagery [167]. To the authors' knowledge, this is the first algorithm to integrate CRF, region merging, and self-training into a single framework. Experimental results show that ST-IRGS outperforms state-of-the-art methods using limited labeled data for training.

Below is the description of the ST-IRGS algorithm:

1. ST-IRGS is formulated in the CRF framework, in which image features such as edge strength can be arbitrarily incorporated into the energy function.
2. ST-IRGS uses a multimodal Gaussian model to calculate the unary potentials, in which multiple Gaussian modalities are assumed for a single class. Bayesian information criterion (BIC) is used to estimate the number of mixtures in each class.
3. ST-IRGS inherits the iterative region merging of the original IRGS algorithm. The integration of iterative region merging with labeling for optimizing the CRF energy function can significantly reduce the number of nodes, so that the convergence rate can be speeded up and near-global optimum solutions can be achieved.
4. ST-IRGS integrates the self-training technique that is capable of iteratively expanding the training sample set and retraining the classifier. Pixels that are in the same regions of the original training samples are selected as new training samples, so that the edge constraint can be implicitly used for training sample selection.

The rest of this chapter is organized as follows. Section 6.2 reviews the IRGS algorithm that the proposed approach is based on. Section 6.3 describes the proposed ST-IRGS algorithm. Section 6.4 shows the performance of the proposed approach on benchmark hyperspectral image datasets. Section 6.5 concludes the chapter by summarizing the above sections.

6.2 Review of IRGS

This section provides a high-level description of the IRGS algorithm [170] to help understand the proposed ST-IRGS algorithm.

The goal of image classification is to find the optimal label field configuration given the image data. In the Bayesian framework, we aim to find a y that optimizes

$$\mathbf{y}^* = \underset{\mathbf{y}\in\mathbf{Y}}{\operatorname{argmax}}\, p(\mathbf{x} \mid \mathbf{y})P(\mathbf{y}) \tag{6.1}$$

The term $p(\mathbf{x} \mid \mathbf{y})$ can be represented as a factorized form if we assume that the conditional probability $P(x_i \mid y_i)$ is independent:

$$p(\mathbf{x} \mid \mathbf{y}) = \prod_i P(x_i \mid y_i) \tag{6.2}$$

The label field $\mathbf{Y}$ can be modeled as an MRF, assuming the Markovian property of the labels. For image labeling problems, usually only the pairwise clique potentials are considered [92].

In IRGS, GMM is used to calculate the unary potential energy. The multi-level logistic (MLL) model [30] is used to incorporate spatial context, penalizing regions that are neighbors but are assigned different labels. An edge penalty function is also incorporated into the pairwise potential energy. A prior model should theoretically not be dependent on the data. In practice, however, the additional edge penalty can help improve classification performance [170].

By taking the logarithm on both sides of Eq. (6.1) and incorporating the unary and pairwise potentials, the region-based energy function is formulated as [117, 170]:

$$\begin{aligned}\mathbf{Y}^r = \underset{\mathbf{y}_i^r, i=1,\ldots,N_r}{\operatorname{argmin}} \Bigg\{ \sum_i^{N_r} \sum_{s\in\Omega_i} \left[\frac{1}{2}\log(\left|\mathbf{\Sigma}_{y_i^r}\right|) + \frac{1}{2}(\mathbf{x}_s - \mu_{y_i^r})^T \mathbf{\Sigma}_{y_i^r}^{-1}(\mathbf{x}_s - \mu_{y_i^r}) \right] \\ + \beta \sum_i \sum_j \sum_{s\in\partial\Omega_i\cap\partial\Omega_j} g(\nabla_s) \Bigg\}\end{aligned} \tag{6.3}$$

where $\mu_{y_i^r}$ and $\mathbf{\Sigma}_{y_i^r}$ are the mean and variance related to class y_i^r, $g(\nabla_s)$ is the edge penalty function, and $\partial\Omega_i \cap \partial\Omega_j$ denotes the boundary between Ω_i and Ω_j.

A simulated annealing (SA) iterative procedure is used to solve the combinatorial optimization problem in IRGS. In each iteration, each region is scanned in a random order, and the label field configuration that minimizes the energy function in Eq. (6.3) is

chosen. Different from traditional pixelwise MRF approaches, IRGS also performs iterative region merging during the SA iterations. This process significantly reduces the number of regions and avoids being trapped in a local minimum. Here, a region is assumed to be homogeneous and unimodal, so the merging criterion can be derived from Eq. (6.3) by assuming each region has a unique label. The region merging starts from the result by an over-segmentation algorithm such as watershed [149], and a region adjacency graph (RAG) is constructed. In each iteration, neighboring regions are merged if the merging can reduce the global energy function. For two neighboring regions Ω_i and Ω_j, the merging criterion is derived as:

$$\delta E_{ij} = (N_i + N_j)\log(|\mathbf{\Sigma}_{ij}|) - N_i \log(|\mathbf{\Sigma}_i|) - N_j \log(|\mathbf{\Sigma}_j|) - \beta \sum_{s \in \partial\Omega_i \cap \partial\Omega_j} g(\nabla_s) \tag{6.4}$$

where N_i and N_j are the number of pixels in Ω_i and Ω_j respectively.

Only regions that have the same class label are allowed to be merged. Once two regions get merged, related nodes and edges in the RAG will be also updated accordingly.

6.3 The ST-IRGS algorithm

For most of remote sensing image labeling methods, classification and segmentation are performed separately. Segmentation is either served as a preprocessing step to reduce computational cost, or a postprocessing step to refine classification results using spatial context. However, human interpretation is not performed in two separate steps. In fact, classification and segmentation can be mutually beneficial to each other. Classification obtains the probability of a pixel belonging to a class, which can help guide the segmentation. Also, segmentation provides region-level features that are more robust to noise, which can reciprocally improve the classifier.

The IRGS algorithm [169] has been used as a joint classification and segmentation approach by incorporating shape prior knowledge into the energy function, while it is still in a unsupervised manner because no training samples are used. Recently, IRGS has been extended to supervised classification by combining with a support vector machine (SVM) [84], but the SVM model is pre-trained and the probabilities remain unchanged during segmentation. In the rest of this section, the proposed ST-IRGS algorithm, which is a joint classification and segmentation approach based on the CRF model will be described, where the self-training technique is integrated to iteratively update the classifier.

6.3.1 Conditional random fields

CRF [81] has been introduced in Section 5.3.2 as well as its advantages over MRF. An example of the CRF model for image labeling is shown in Fig. 6.1. There are mainly two advantages of CRF over MRF that can benefit our approach. First, the unary potential at site s in CRF is a function of all the observation data $\mathbf{x}$ as well as the label y_s, while in MRF it is a function of $\mathbf{x}_s$ and y_s only. In our approach, the label of a pixel y_s is not determined only by the features at the same site $\mathbf{x}_s$, but by all the features in the same region as s. Second, in MRF, the pairwise potential for each pair s and t is independent of the observation, while in CRF it is a function of all the $\mathbf{x}_s$ as well as the labels y_s and y_t. Therefore, the edge strength can be naturally incorporated into the the pairwise potential in CRF.

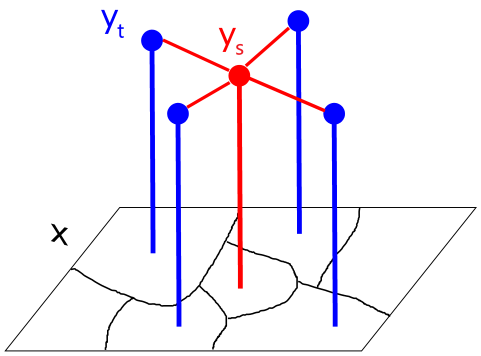


Figure 6.1: Example of a CRF for image labeling.

To summarize, there are mainly two advantages of CRF over MRF that can benefit our approach. First, the unary potential at site s in CRF is a function of all the observation data $\mathbf{x}$ as well as the label y_s, while in MRF it is a function of $\mathbf{x}_s$ and y_s only. In our approach, the label of a pixel y_s is not determined only by the features at the same site $\mathbf{x}_s$, but by all the features in the same region as s. Second, in MRF, the pairwise potential for each pair s and t is independent of the observation, while in CRF it is a function of all the $\mathbf{x}_s$ as well as the labels y_s and y_t. Therefore, the edge strength can be naturally incorporated into the the pairwise potential in CRF.

In Section 6.3.2, the details about the unary and pairwise potentials in the CRF model, as formulated in Eq. (5.11) are shown.

6.3.2 Defining unary and pairwise potentials

For the unary potentials, we simply use the Gaussian classifier extended from the original IRGS algorithm to generate probability estimates, but our approach can also be adaptable to other supervised or semi-supervised probabilistic classifiers. The main advantages of a Gaussian classifier are its simplicity of estimating parameters and its meaningful probability estimates compared to the non-probabilistic classifiers. Given sufficient training samples, a Gaussian classifier works very well when the data fit a normal distribution.

The traditional Gaussian MLC [98] assumes each class to be a single Gaussian distribution. In practice, however, a class often includes multiple materials that have different spectral signatures, in which case a single Gaussian distribution is incapable of modeling the whole class. Therefore, we assume that each class is a linear mixture of multiple Gaussian distributions, which we call "multi-modal Gaussian MLC" (MGMLC):

$$p(\mathbf{x_s}|y_s) = \sum_{k=1}^{\mathcal{C}_{y_s}} \alpha_k \mathbb{N}(\mathbf{x_s}, \mu_\mathbf{k}, \mathbf{\Sigma_k}) \tag{6.5}$$

where $\mathcal{C}_{y_s}$ is the number of clusters in the class y_s, and

$$\mathbb{N}(\mathbf{x_s}, \mu_\mathbf{k}, \mathbf{\Sigma_k}) = \frac{1}{(2\pi)^{d/2}\,|\mathbf{\Sigma_k}|^{1/2}} exp\,\{\, -\frac{1}{2}(\mathbf{x_s} - \mu_\mathbf{k})^T \mathbf{\Sigma_k}^{-1}(\mathbf{x_s} - \mu_\mathbf{k})\,\} \tag{6.6}$$

MGMLC combined with dimension reduction techniques has been found to outperform SVM for hyperspectral image data [94]. But different from MLC, the model parameters for each Gaussian distribution cannot be directly obtained from training samples. Expectation maximization (EM) is often used to estimate the parameters by maximizing the expected log-likelihood function. Also, the number of mixtures for each class is unknown. There are many methods to estimate the number of clusters. We simply use BIC [128] that has been demonstrated to provide an accurate estimation on the number of clusters for hyperspectral data [94]. Given the complete log-likelihood $\hat{L}$ with optimal model parameters optimized by MGMLC, BIC can be calculated without extra cost [54]:

$$\text{BIC} = -2\log(\hat{L}) + d(\log N_s) \tag{6.7}$$

where N_s is the number of samples and d is the number of features.

To incorporate Eq. (6.5) into the CRF model, we define the unary potential and use Bayes' theorem:

$$\begin{aligned}\phi_s(y_s, \mathbf{x}) &= -\log\{p(y_s \mid \mathbf{x}_s)\} \\ &= -\log\left\{\frac{p(\mathbf{x}_s \mid y_s)P(y_s)}{p(\mathbf{x}_s)}\right\}\end{aligned} \tag{6.8}$$

Note that $p(\mathbf{x}_s)$ is independent of $\mathbf{Y}$ and thus can be ignored in the optimization.

For the pairwise potential, we only consider first-order interactions, i.e., two nodes only have non-zero weights when they are neighbors. Here, the pairwise potential is based on a function of the gradient structure distribution. The Otsu's threshold [108] is used to determine a threshold that maximizes inter-class separability between two gradient clusters. So we define

$$\xi_{st}(y_s, y_t, \mathbf{x}) = \begin{cases} \beta g(\nabla_s) & y_s \neq y_t \\ 0 & \text{otherwise} \end{cases} \tag{6.9}$$

$$g(\nabla_s) = exp\{-\alpha \left\|\mathbf{x}_s - \mathbf{x}_t\right\|_\infty^2 / K(\tau)^2\} \tag{6.10}$$

where $\|\cdot\|_\infty^2$ is the infinite norm that returns the maximum absolute value for all the feature bands in the image, $\alpha = 1/(0.25 \cdot T_{\text{Otsu}})$, where T_{Otsu} is the Otsu threshold of the edge image which adapts the edge strength based on global edge strength of the image, and $K(\tau)$ is monotonically increasing with the iteration number τ [170]:

$$K(\tau) = \begin{cases} f(2\tau), & 1 \leq \tau \leq 50 \\ 1.1K(\tau - 1), & \tau > 50 \end{cases} \tag{6.11}$$

where $f(i)$ takes a value in $[0, 1]$ indicating that i percentage of the site pairs have weaker edge strength than $f(i)$. Such a penalty function is called "graduated increased edge penalty" that has been experimentally demonstrated to be satisfactory [170].

By incorporating Eq. (6.8) and Eq. (6.9) into Eq. (5.11) and cumulating the potentials related to the same region, we have the region-based energy optimization function for the CRF model:

$$\begin{aligned}\mathbf{Y}_{\text{CRF}}^r = \underset{\mathbf{y}_i^r, i=1,\ldots,N_r}{\text{argmin}} \Bigg\{ &\sum_i^{N_r} \sum_{s \in \Omega_i} -\log\{p(\mathbf{x}_s \mid y_i^r)P(y_i^r)\} \\ &+\beta \sum_i \sum_j \sum_{s \in \partial\Omega_i \cap \partial\Omega_j} g(\nabla_s) \Bigg\}\end{aligned} \tag{6.12}$$

where $P(y_i^r)$ is the class prior, and $p(\mathbf{x}_s \mid y_i^r)$ is calculated from Eq. (6.5).

Even though the energy function in Eq. (6.12) is very similar to (6.3) except the replacement of GMM by the supervised GMGLC classifier, the definition of CRF provides the theoretical legality of using edge strength in the pairwise potential of (6.12). In fact, the optimization of Eq. (6.12) is the same as the optimization of the pixelwise CRF model with region equivalence constraints. If the label of a pixel is changed by Gibbs sampling, all the other pixels in the same region should change to the same labels as well. Therefore, we can consider the Gibbs sampling is performed on each region in the RAG as a unit.

6.3.3 Joint region merging and self-training

In our approach, the same assumption of regions as that in the IRGS algorithm described in Section 6.2 is used, so the merging criterion in the proposed approach is the same as Eq. (6.4). The only difference is that we further restrict the necessary condition for merging. After MGMLC is performed on the training data, each region is assigned an additional "cluster label" to one of the Gaussian mixtures in the class that region is labeled. In the region merging step, two adjacent regions are allowed to be merged when they are not only labeled as the same class, but also as the same cluster. Such a criterion can help protect the homogeneity of regions so that they can be modeled as Gaussian distribution with single modality. Moreover, if a region contains training samples, the risk of incorporating samples that are incorrectly labeled into the training set could be reduced.

Followed by region merging, the self-training step is performed in order to expand the training set and retrain the classifier iteratively. Self-training (also called "self-learning" or "self-teaching") is a wrapper algorithm that has been commonly used for semi-supervised learning [181]. In the self-training framework, we first train a classifier using a small amount of labeled data, then use it to classify the unlabeled data. Later we select the most confident unlabeled points with their predicted labels and move them into the training set. The classifier is retrained multiple times. However, such a procedure is largely dependent on the performance of the classifier. In the early iterations, there is an insufficient number of training samples, and thus the classifier is very likely to obtain inferior results, which might in turn make the results iteratively worse.

In the remote sensing context, most image classification tasks select training samples and perform classification on the same image, and pixels around a training sample have high probability of belonging to the same class as that training sample. However, such location information of training samples was rarely used in previous literature. Dópido et al. [33] recently used the spatial locations of the initial labeled training samples to

select unlabeled samples. Only pixels that are around the labeled training samples can be selected, based on the assumption that nearby pixels are likely to belong to the same class. This is safer than selecting unlabeled samples only depending on their estimated class probabilities. Nevertheless, the self-training algorithm by Dópido et al. [33] is a naive model without consideration of either edge strength or spatial context.

In our approach, the region merging and self-training procedures are both concatenated into the spatial context model. The region merging technique enables the integration of a safer strategy of self-training, i.e., only pixels that belong to the same region as a labeled sample are labeled and incorporated into the training samples. Such a strategy not only uses spatial constraint, but also takes the edge strength into account in an implicit manner because edge strength has been incorporated into the merging criterion. Further, the spatial context model can label the outliers according to their surrounding pixels. Such outliers can be considered as obstacles, which are "cleaned up" by the spatial context model to boost region merging, and subsequently enables the further expansion of more training samples. Lastly, the self-training naturally stops when the regions stop merging, so extra stopping criteria for self-training are not required.

6.3.4 Selection of weight parameter

The weight parameter β in Eq. (6.3) controls the smoothness by leveraging the data term and spatial term in the CRF model. It is very challenging to select the weight parameter automatically, and thus it was predetermined and fixed during the optimization process in many random field based approaches [90, 142, 160]. Nevertheless, some approaches that modify the parameter adaptively have been shown to outperform those using a fixed parameter. Deng and Clausi [29] used a gradually increasing weighting scheme during SA. In IRGS [170], the parameter is based on class separability which is decreasing as spatial context is incorporated during the merging process. For supervised classification, the weight parameter can be selected by exhaustive grid search. To avoid the high computational cost resulted from grid search, Serpico and Moser [129] proposed an automatic supervised procedure for optimizing the weight parameters based on the Ho-Kashyap algorithm. However, when the number of labeled training samples is limited, grid search or other automatic selection methods might not be feasible. Here, we adopt the parameter selection method in IRGS to a supervised context.

In IRGS, the weight parameter is defined based on class separability [117]:

$$\hat{\beta} = C_1 \frac{J_{\min}/C_2}{1 + J_{\min}/C_2} \beta_0 \tag{6.13}$$

where β_0 is the ratio of the current total class boundary length over the image size [117], C_1 and C_2 are constants, and $J_{\min} = \min_{ij} J_{ij}$ is the minimum pairwise class separability.

In ST-IRGS, the Fisher's criterion is extended to multiple Gaussian modalities:

$$J_{ij} = \sum_{s \in c\{i\}} \sum_{t \in c\{j\}} p(y_i)p(y_j)Tr(S_{W_{st}}^{-1} S_{B_{st}}) \tag{6.14}$$

$$S_{W_{st}} = \frac{N_s}{N_s + N_t}\mathbf{\Sigma}_s + \frac{N_t}{N_s + N_t}\mathbf{\Sigma}_t \tag{6.15}$$

$$S_{B_{st}} = (\mathbf{u}_s - \mathbf{u}_t)(\mathbf{u}_s - \mathbf{u}_t)^T \tag{6.16}$$

where $c\{i\}$ and $c\{j\}$ represent the sets of modalities belonging to class i and j respectively.

J_{ij} is updated using the estimated GMM parameters updated in each iteration. In the original IRGS algorithm, J is relatively large at the beginning of the iterations because the feature model dominates the energy function, and is decreasing as segmentation proceeds due to the spatial context that gradually plays an important role [170]. In the ST-IRGS algorithm, however, the training samples are incapable of representing the true class at the beginning and result in poor classification accuracy, so the class separability is relative low. As self-training proceeds, the classifier is retrained and the model parameters are refined. Then, J tends to increase, and so does $\hat{\beta}$. This agrees with the intuition that the spatial context is only helpful when a majority of predictions by the unary classifier is correct.

6.3.5 Summary of the proposed algorithm

The summary of the ST-IRGS algorithm is shown in Algorithm 4. The watershed algorithm is first performed on the image to obtain an oversegmentation result, and pixels in the same region of the training samples are added into the training set. Then, the training of the MGMLC classifier, Gibbs sampling, and region merging are performed iteratively. The parameter β is updated in each iteration, and the edge penalty is also updated as a function of $K(\tau)$. After the maximum number of iterations is reached, the boundary sites are labeled using maximum a posteriori [117].

6.4 Experiments

In this section, experiments are conducted for testing the performance of the proposed ST-IRGS algorithm. The hyperspctral datasets for testing are described in Section 3.1.

Algorithm 4 The ST-IRGS algorithm

Input: Set C_1, C_2 in Eq. (6.14);
Set $\tau_{\max}$ and let $\tau = 0$;
Set initial temperature T_0 and K_0;
Obtain the edge map of the image using Eq. (6.10).

Output: A predicted class map.

1: Obtain an initial over-segmentation result by watershed transform and construct an initial RAG;
2: Update $\mathcal{S}_{Tr}$ and $\mathcal{D}$. For a pixel s, add it to $\mathcal{S}_{Tr}$ and its corresponding features and estimated label $(\mathbf{x}_s, y_s)$ into $\mathcal{D}$ if $\exists t$, $t \in \mathcal{S}_{Tr}$ and s, t belong to the same region;
3: Estimate the number of mixtures for each class using Eq. (6.7) on the current $\mathcal{D}$;
4: Retrain the classifier and calculate the class probabilities for each pixel using Eq. (6.5);
5: Update β using Eq. (6.13), let $\tau = \tau + 1$, update $T(\tau)$ and $K(\tau)$, and update edge penalty in Eq. (6.10);
6: Perform Gibbs sampling on the current RAG to find a suboptimal solution for Eq. (6.12). Each RAG vertex is visited once randomly. If $\tau = \tau_{\max}$, go to Step 9.
7: Calculate ∂E_{ij} using Eq. (6.3) for each pair of adjacent regions in the current RAG, if Ω_i and Ω_j have the same cluster label;
8: If the minimum ∂E_{ij} is negative, merge Ω_i and Ω_j, and then update the RAG and Ω correspondingly. If merging happens, go to Step 7; Otherwise go to Step 2;
9: Label region boundary sites on the class boundaries to one of the classes its neighboring regions belong based on maximum a posteriori.

The experimental analysis is mainly focused on the first two datasets whose ground truth areas are large enough to verify the correctness of labels as the training set is expanded by self-training. The experimental setup is described in Section 6.4.1, and the experiments are shown in Section 6.4.2.

6.4.1 Experimental setup

The ST-IRGS algorithm is implemented in Microsoft Visual C++ 2010. The same parameters are used for both datasets. C_1 and C_2 in Eq. (6.13) are set to 5 and 1 respectively based on the experiment. T_0 is set to 1, and $T(\tau_{\max}) = T_0 \cdot 0.98^{\tau-1}$. $\tau_{\max}$ is set to 100. The maximum number of Gaussian mixtures in each class is set to 5. If the number of training samples in a class is less than 200, the number of mixtures is automatically set

to 1 to avoid insufficient training samples. In contrast, if the number of available training samples is too large, we perform Monte Carlo sampling to reduce computational costs for estimating GMM parameters, e.g., sample a pixel with a probability p:

$$p = \begin{cases} 1, & N_i < N_{\max} \\ \frac{N_{\max}}{N_i}, & \text{otherwise} \end{cases} \tag{6.17}$$

where N_i is the number of pixels available for the training set, and $N_{\max}$ is a threshold of maximum sampled pixels. In the experiment, λ is set to 0.01, and $N_{\max}$ is set to 5000.

To avoid excessive merging in a single step which might cause class imbalance of the expanded training set, we define the maximum number of merges in one iteration, which is set to 300. In each iteration, the merging is processed from the most negative ∂E_{ij} in an ascending order. To reduce computation time, the model parameters are not re-estimated unless $N_i > \lambda N_i^{\text{old}}$ for class i, where N_i is the number of pixels available for the training set that, N_i^{old} is the number of pixels related to the old classifier, and λ is a constant that is set to 1.01 in the experiment.

As a preprocessing step, PCA is first applied to reduce the dimensions to 10. We use three algorithms that are also based on Gaussian models for comparison: MGMLC, GMRF, and ST-IRGS-S. GMRF combines MGMLC with the standard MRF model, and ST-IRGS-S is a simple version of ST-IRGS by assuming a single Gaussian distribution (or called "uni-modal") for each class. All the methods are region-based methods that perform the watershed algorithm first, and use pixels that are in the same watershed region as the initial set for training. At the end, one watershed or higher-scale region is assigned a unique class label. The result by MGMLC serves as an initialization for the other three methods. At the end, we also compare the results by ST-IRGS to results by state-of-the-art methods published in recent years using the same datasets and limited training data.

In the experiment, all the tests are repeated ten times using different randomly-selected labeled training samples, five per class for both images. Below are the results and analysis.

6.4.2 Experimental results and analysis

Experiments: University of Pavia dataset

The classification results are shown in Table 6.1. The confusion matrix using the ST-IRGS in one test is shown in Table 6.2. MGMLC followed by a watershed algorithm only achieves an OA of 71.8% by averaging ten runs. In Fig. 6.2 (a), even though the watershed algorithm

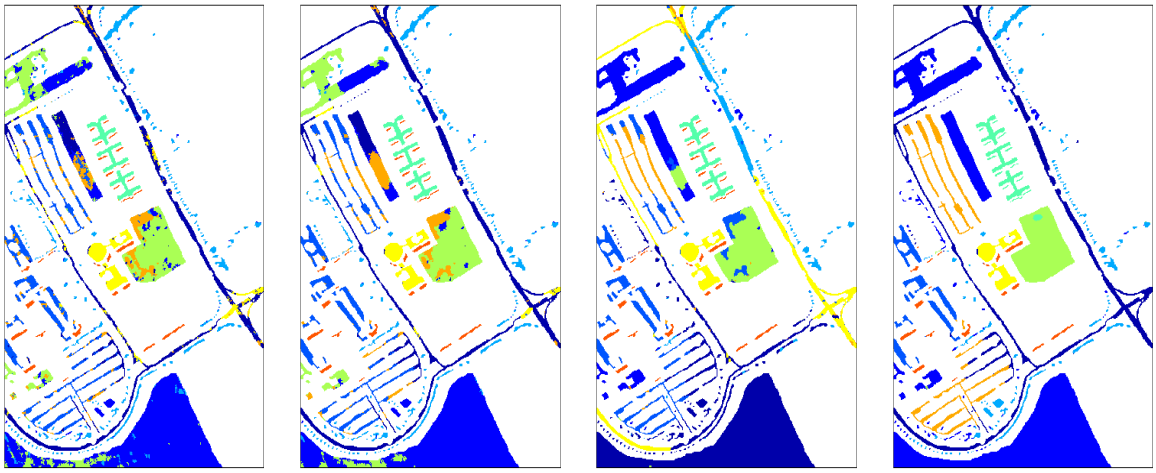
Table 6.1: Class accuracy, overall accuracy (OA%) and average accuracy (AA%) in percentage, and kappa coefficient ([κ]) using different region-based methods for the University of Pavia dataset with five labeled samples per class (45 samples in total).

	MGMLC	GMRF	ST-IRGS-S	ST-IRGS
Asphalt	80.9±11.3	87.8±7.5	41.3±48.5	**98.1**±2.6
Meadows	65.2±12.7	67.5±13.2	40.0±41.8	**85.0**±15.1
Gravel	74.2±15.4	78.9±15.1	**92.2**±10.5	91.8±11.4
Trees	**97.2**±2.3	97.2±2.9	83.3±11.3	87.9±10.7
Painted metal sheets(1345)	99.6±0.2	**99.7**±0.2	97.4±7.9	96.6±10.6
Bare Soil	72.1±12.5	77.5±13.1	88.3±8.5	**95.7**±4.4
Bitumen	85.1±24.8	86.1±27.6	**99.7**±0.3	98.9±0.4
Self-Blocking Bricks	44.8±24.5	55.8±28.8	26.2±19.6	**83.2**±27.5
Shadows	**99.0**±0.9	98.8±0.9	93.2±16.6	86.5±7.2
OA	71.8±5.4	75.7±5.0	55.0±19.9	**89.5**±6.3
AA	79.8±3.9	83.3±4.9	73.5±7.5	**91.5**±3.6
κ	0.80±0.04	0.83±0.05	0.74±0.08	**0.92**±0.04

can reduce the noise more or less, the labels are still noisy due to the large within-class variance of the same class and insufficiency of training samples. This is evidently shown in the bare soil area in the middle of the image, and the meadows area at the bottom. Also, the filament-shape self-blocking bricks among the trees are largely misclassified as gravels, yielding to only 44.8% classification accuracy.

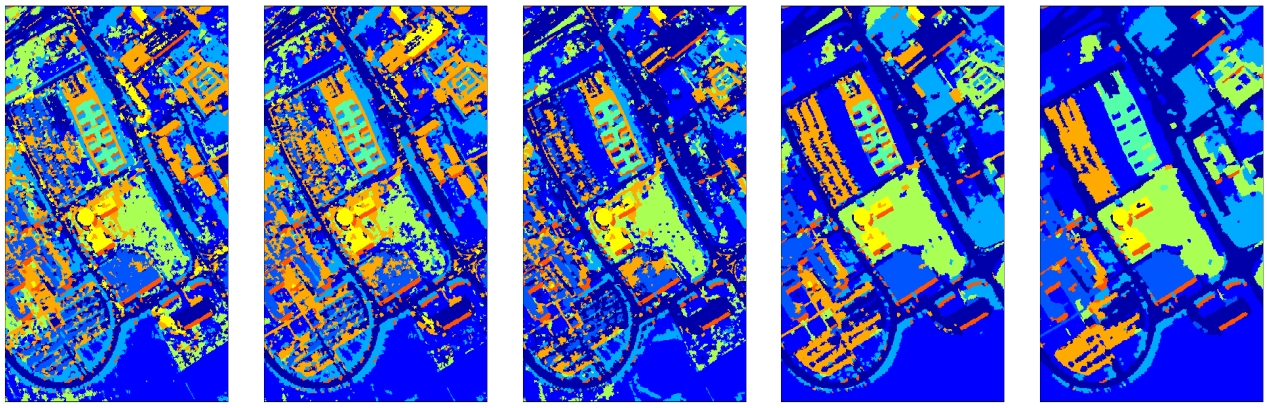
In Fig. 6.2 (b), the isolated misclassified labels are largely removed due to the spatial context using MRF model, and the final OA is improved by 3.9% compared to MGMLC. However, the errors could be hardly corrected if a majority of labels are misclassified in a region. For example, a large area of the meadows on the top left is still misclassified as shadows or bare soil. Compounding the errors, the wrong labels tend to accumulate and dominate the area, resulting in even poorer classification accuracy. This is the reason why the ST-IRGS-S achieves unsatisfactory classification accuracy. In Fig. 6.2 (c), the whole meadow area at the bottom is misclassified as asphalt. This indicates that the uni-modal Gaussian distribution is incapable of modeling some classes in this dataset, which is more significant as the training set becomes larger. The misclassified pixels might be selected for new training samples to train a new classifier or guide the region merging in the future iterations, and thus might make things worse.

The ST-IRGS algorithm achieves higher classification performance than the above three methods on this dataset. The classification performance by ST-IRGS improves stably as the number of iterations increases, and converges in about 90 iterations, while the classification



(a) MGMLC (b) GMRF (c) ST-IRGS-S (d) ST-IRGS

Figure 6.2: Classification results by different methods for the University of Pavia image



(a) $\tau = 1$ (b) $\tau = 10$ (c) $\tau = 20$ (d) $\tau = 50$ (e) $\tau = 100$

Figure 6.3: Classification results in different iterations by ST-IRGS for the University of Pavia image using five labeled training samples per class.

Table 6.2: Confusion matrix using the ST-IRGS algorithm for the University of Pavia dataset with five randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.1 in order.

	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	5985	2	0	561	0	0	0	33	0
C2	0	15590	1	1070	0	939	0	0	999
C3	9	0	2040	0	0	0	0	0	0
C4	56	170	0	2659	0	27	0	2	100
C5	0	0	1	0	1294	0	0	0	0
C6	0	1337	0	88	0	3539	0	15	0
C7	15	0	0	0	0	0	1265	0	0
C8	4	0	0	1165	0	0	0	2463	0
C9	0	0	0	24	17	0	0	0	856

accuracy by ST-IRGS-S drops significantly after a few iterations, as shown in Fig. 6.4. as shown in Fig. 6.4. The intermediate results by ST-IRGS are shown in Fig. 6.3. The result in Fig. 6.3 (a) is the same as that by MGMLC. Its corresponding training set is shown in Fig. 6.5 (a). The training set gradually expands as the number of iterations increases, so more and more training samples are collected to improve the classifier. Meanwhile, the incorporation of spatial context smooths the labeling and thus helps the expansion of the training set. Comparing Fig. 6.3 (d) with (a), (b), and (c), the labels are smoother and more accurate. For example, the class of self-blocking bricks which achieves very low accuracy in the early iterations are, for the most part, correctly classified in Fig. 6.3 (d). Even though the area of this land cover is in a filament shape, it can still be expanded by ST-IRGS. Moreover, large misclassified regions in the meadow areas can also be corrected owing to the expansion of the training set. At the end of iterations, the number of regions is reduced to about 2000, which is approximately 1/6 of the original number of regions, so that the chance of converging to a suboptimal solution can be lowered.

Experiments: Salinas dataset

The classification results for the Salinas dataset are shown in Table 6.3. The confusion matrix using the ST-IRGS in one test is shown in Table 6.4. Most classes have good separability and thus achieve satisfactory classification accuracy even by using MGMLC. One exception is the separation between the grapes-untrained class and the vinyard-untrained class. In Fig. 6.7, most of the grapes-untrained area is misclassified into the

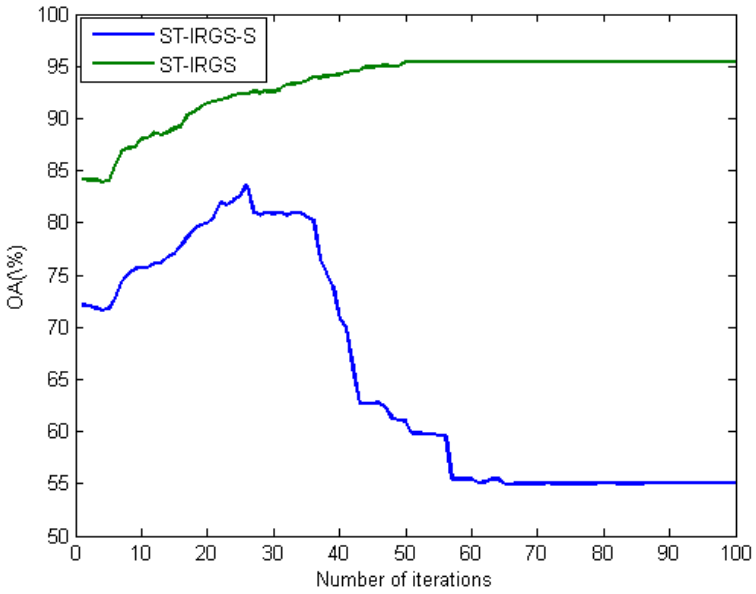
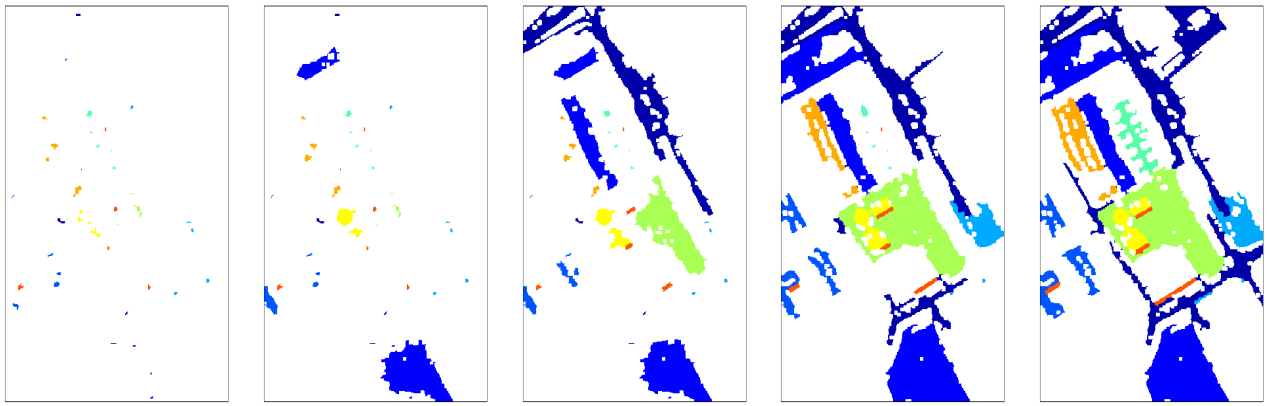


Figure 6.4: Overall accuracy by ST-IRGS and ST-IRGS-S during the iterations for the University of Pavia image.



(a) $\tau = 1$ (b) $\tau = 10$ (c) $\tau = 20$ (d) $\tau = 50$ (e) $\tau = 100$

Figure 6.5: Training set in different iterations by ST-IRGS for the University of Pavia image using five labeled training samples per class.

Table 6.3: Class accuracy, overall accuracy (OA%) and average accuracy (AA%) in percentage, and kappa coefficient ($[\kappa]$) using different region-based methods for the Salinas dataset with five labeled samples per class (80 samples in total).

	MGMLC	GMRF	ST-IRGS-S	ST-IRGS
Brocoli-green-weeds-1	97.4±3.4	99.0±1.9	**100.0**±0.0	**100.0**±0.0
Brocoli-green-weeds-2	99.6±0.6	99.3±1.1	**100.0**±0.0	99.9±0.2
Fallow	99.5±0.9	96.0±8.2	99.6±1.1	**99.9**±0.2
Fallow-rough-plow	99.9±0.2	100.0±0.1	**100.0**±0.0	99.9±0.1
Fallow-smooth	80.3±16.7	84.2±17.0	97.6±0.1	**98.6**±0.2
Stubble	99.8±0.2	99.9±0.1	**99.9**±0.0	99.9±0.2
Celery	99.5±0.7	99.6±0.7	99.9±0.3	**99.9**±0.1
Grapes-untrained	55.8±22.8	59.0±30.4	92.7±3.4	**93.1**±3.3
Soil-vinyard-develop	98.5±1.5	96.9±3.6	**99.9**±0.1	99.5±0.5
Corn-senesced-green-weeds	87.1±10.1	**88.7**±10.7	84.5±13.9	84.8±12.1
Lettuce-romaine-4wk	89.5±6.3	90.0±6.0	**94.4**±3.6	88.2±31.2
Lettuce-romaine-5wk	88.5±12.3	91.0±11.5	**100.0**±0.0	100.0±0.0
Lettuce-romaine-6wk	96.8±2.9	**98.0**±1.5	97.9±1.5	97.0±4.8
Lettuce-romaine-7wk	93.3±5.6	**93.8**±5.1	81.7±29.5	87.7±18.9
Vinyard-untrained	74.5±13.8	81.5±13.5	76.8±17.8	**89.2**±13.4
Vinyard-vertical-trellis	94.9±3.1	95.9±2.7	**98.6**±1.4	97.4±3.4
OA	84.2±4.3	86.0±5.8	93.7±2.6	**95.4**±2.5
AA	90.9±2.3	92.0±2.4	95.2±2.3	**95.9**±2.9
κ	0.91±0.02	0.92±0.02	0.95±0.02	**0.96**±0.03

vinyard-untrained class. Such misclassification is difficult to be corrected by simple random fields or other smooth labeling methods. Therefore, the OA is only improved by 1.8% using MRF. With self-training, ST-IRGS-S increases an OA by 7.7% over GMRF after using the self-training technique. ST-IRGS also shows a stable improvement from the MGMLC initialization, and is slightly better than ST-IRGS-S, as shown in Fig. 6.6. ST-IRGS converges in about 50 iterations, and achieves the highest classification accuracy of 95.4%.

In Fig. 6.8, even though the grapes-untrained class and the vinyard-untrained class have very low separability, the training set related to these two classes is still expanding in a correct way. There are mainly two reasons. First, as the number of training samples increases, the estimated model parameters are more close to the parameters for the true classes, and thus the classification error rate is reduced. Second, the edge strength is relatively strong between the two areas, which can prevent assigning the same label to regions on both sides, or merging the regions across the boundary. Fig. 6.9 and Fig. 6.6

Table 6.4: Confusion matrix using the ST-IRGS algorithm for the Salinas dataset with five randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.2 in order.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16
C1	1959	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C2	0	3652	0	0	0	0	0	0	0	24	0	0	0	0	0	0
C3	0	0	1926	0	0	0	0	0	0	0	0	0	0	0	0	0
C4	0	0	0	796	3	0	0	0	0	0	0	0	0	0	545	0
C5	0	0	0	33	2593	1	0	0	0	0	0	0	0	0	1	0
C6	0	0	0	0	0	3908	1	0	0	0	0	0	0	0	0	0
C7	0	0	0	0	0	0	3528	0	0	0	0	0	0	0	1	0
C8	0	0	0	0	0	0	0	10272	0	145	0	0	0	0	804	0
C9	0	0	0	0	0	0	0	0	6153	0	0	0	0	0	0	0
C10	0	4	0	0	0	0	0	0	47	3177	0	0	0	0	0	0
C11	0	2	0	0	0	0	0	0	0	0	1016	0	0	0	0	0
C12	0	0	0	0	0	0	0	0	0	0	239	1638	0	0	0	0
C13	0	0	0	0	0	0	0	0	0	0	0	0	850	16	0	0
C14	0	0	0	0	0	0	0	0	0	35	0	0	0	985	0	0
C15	0	0	0	0	0	0	0	1828	0	4	0	0	0	0	5386	0
C16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1757

show the gradual refining of the labeling and improvement of classification accuracy as the number of iterations increases.

In Fig. 6.8 (e), most pixels in the grapes-untrained class and vinyard-untrained classes can be expanded to the training set with the correct labels. There is an unexpanded region in the right of the vinyard-untrained class, which is finally misclassified as grapes-untrained because its characteristics are more close to that of the grapes-untrained class. This can be probably avoided by adding a training sample into the region. Even though, such a misclassification is more easily corrected than pixel-based methods because one can simply change the label of the whole region in the post-processing step.

Experiments: Kennedy Space Center dataset

The classification results on the Kennedy Space Center dataset are shown in Table 6.5. The confusion matrix using the ST-IRGS in one test is shown in Table 6.6. Compared to the first two datasets, the ground truth only covers very small areas compared to the image size, so the test accuracy might not reflect the true accuracy of the whole image. Nevertheless, ST-IRGS achieves highest classification accuracy among all the methods. Though ST-IRGS only outperforms GMRF by 3.4% OA, we can observe in Fig. 6.10 that a large part of water areas is misclassified. Such a misclassification does not affect the classification accuracy because there is no ground truth of these areas. Also, the details of some small regions in Fig. 6.10 (d) can be preserved despite the edge strength is weaker than that in the first two images.

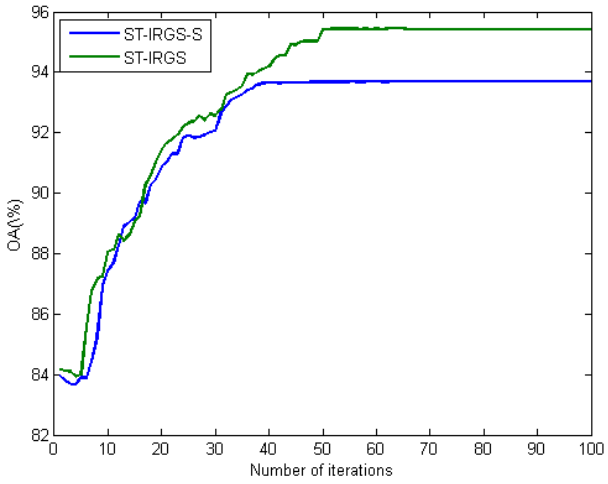
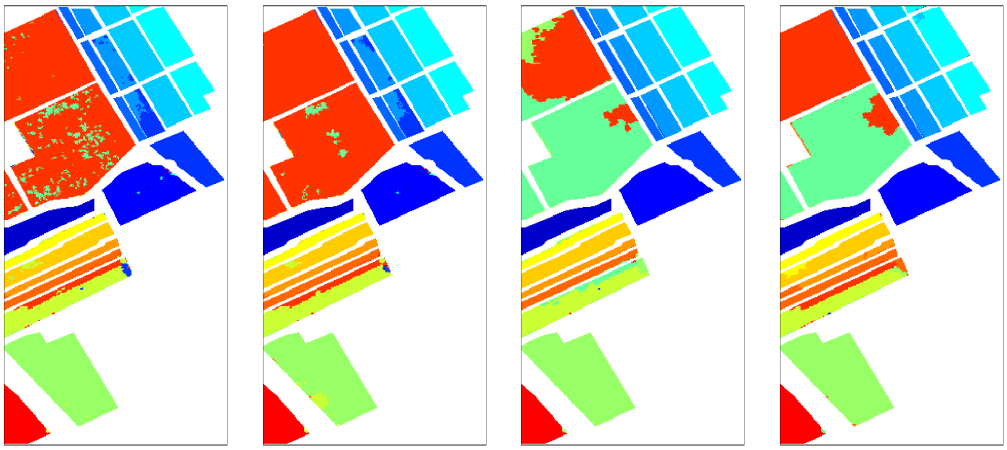
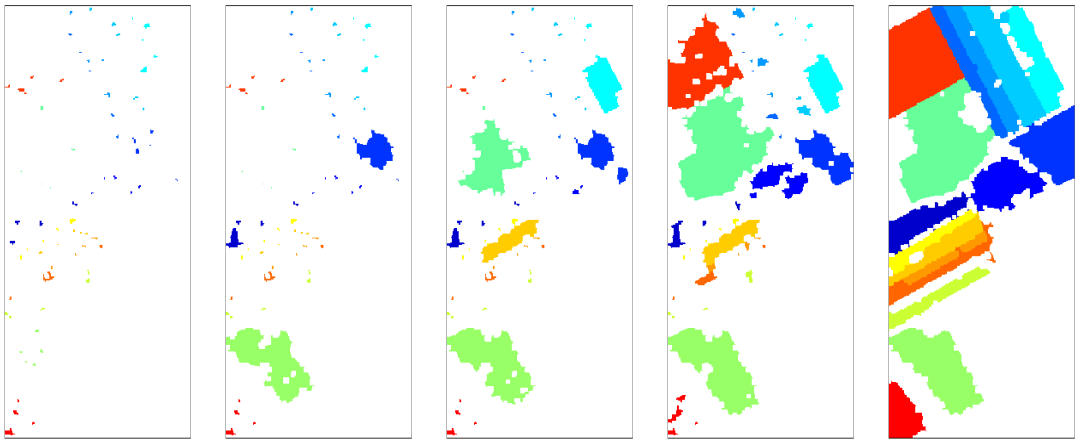


Figure 6.6: Overall accuracy by ST-IRGS and ST-IRGS-S during the iterations for the Salinas image.



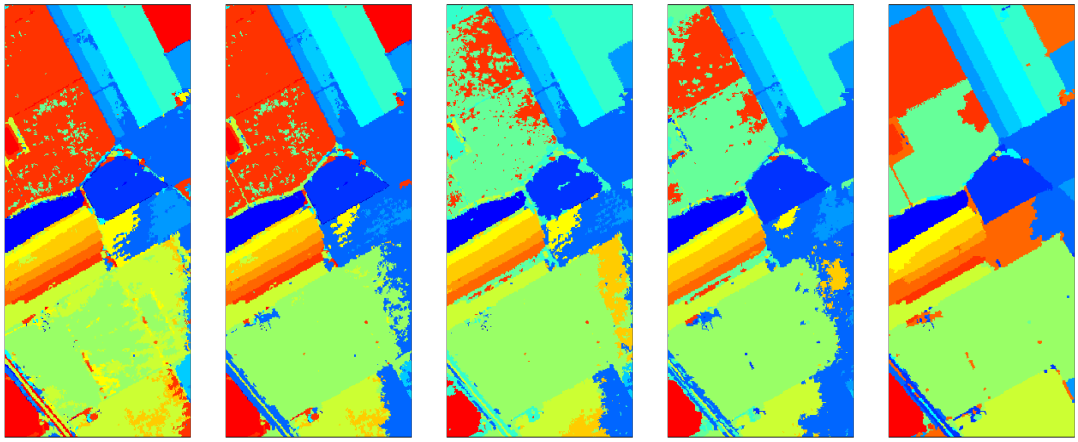
(a) MGMLC (b) GMRF (c) ST-IRGS-S (d) ST-IRGS

Figure 6.7: Classification results by different methods for the Salinas image



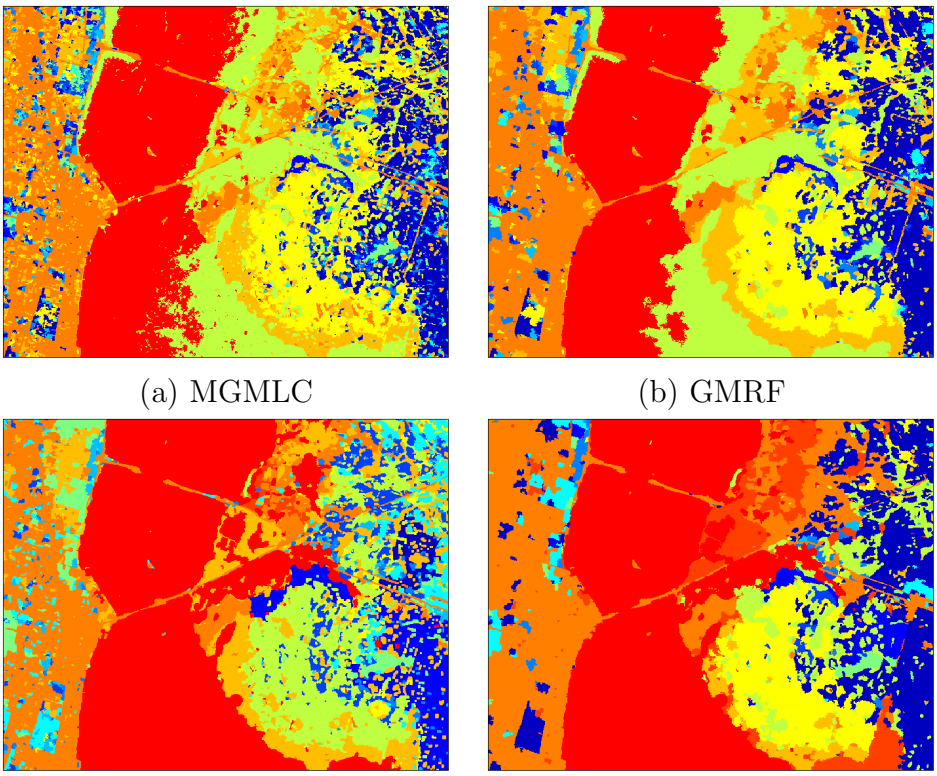
(a) $\tau = 1$ (b) $\tau = 5$ (c) $\tau = 10$ (d) $\tau = 20$ (e) $\tau = 50$

Figure 6.8: Training set in different iterations by ST-IRGS for the Salinas image using five labeled training samples per class.



(a) $\tau = 1$ (b) $\tau = 5$ (c) $\tau = 10$ (d) $\tau = 20$ (e) $\tau = 50$

Figure 6.9: Classification results in different iterations by ST-IRGS for the Salinas image using five labeled training samples per class.



(a) MGMLC (b) GMRF

(c) ST-IRGS-S (d) ST-IRGS

Figure 6.10: Classification results by different methods for the Kennedy Space Center image

Table 6.5: Class accuracy, overall accuracy (OA%) and average accuracy (AA%) in percentage, and kappa coefficient ([κ]) using different region-based methods for the Kennedy Space Center dataset with five labeled samples per class (65 samples in total).

	MGMLC	GMRF	ST-IRGS-S	ST-IRGS
Scrub	89.1±3.3	**95.7**±1.7	62.3±31.3	93.7±10.6
Willow swamp	91.3±2.8	**97.0**±1.7	60.6±27.2	85.9±24.5
Cabbage palm hammock	77.1±8.8	**86.9**±8.6	81.0±30.0	70.1±14.6
Cabbage palm//oak hammock	61.8±8.0	73.6±12.6	73.6±10.5	**73.7**±14.3
Slash pine	83.7±8.8	87.0±7.5	**88.6**±8.2	85.1±7.8
Oak//broadleaf hammock	52.2±8.1	72.0±14.0	**80.6**±17.6	69.4±12.3
Hardwood swamp	84.9±9.4	99.6±0.8	**100.0**±0.0	**100.0**±0.0
Graminoid marsh	65.0±13.9	68.3±18.8	**83.7**±17.9	81.5±18.5
Spartina marsh	83.7±13.0	**90.7**±9.1	80.0±42.2	87.6±28.2
Cattail marsh	88.0±8.9	**97.7**±3.2	92.4±11.0	90.5±11.6
Salt marsh	88.2±2.8	89.5±4.0	90.5±5.1	**92.7**±6.4
Mud flats	45.6±31.0	50.9±36.5	61.3±32.9	**92.1**±7.8
Water	99.2±1.5	98.2±5.7	95.4±9.7	**100.0**±0.0
OA	80.7±2.6	86.2±4.4	79.9±14.1	**89.4**±4.6
AA	77.7±2.0	85.2±3.6	80.8±11.9	**86.3**±4.8
κ	0.78±0.02	0.85±0.04	0.81±0.12	**0.86**±0.05

Results of ST-IRGS using different number of labeled training samples

The result of ST-IRGS using different number of labeled training samples (5, 10, 15, and 20) is shown in Fig. 6.11. As the number of labeled training samples increases, the OA by ST-IRGS is improved on all the three datasets. But there is little improvement after the number of labeled training samples per class is greater than 15. For the Salinas dataset, there is only 1.4% improvement when using 20 labeled samples per class compared to using 5 samples. In practice, there is a trade-off between spending some time to label a few more training samples and sacrificing several percent of classification accuracy. Also, different classes may require different number of labeled training samples depending on their complexity.

Comparison of ST-IRGS and other methods

The comparison of the ST-IRGS algorithm, the MBRF-CRF algorithm in Chapter 5, and other state-of-the-art methods recently published for limited training samples problem is

Table 6.6: Confusion matrix using the ST-IRGS algorithm for the Kennedy Space Center dataset with five randomly-selected labeled samples per class. The number n at i^{th} row and j^{th} column means n pixels that belong to class i are misclassified into class j. "C1", "C2", ... represent the corresponding classes in Table 3.3 in order.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
C1	697	0	0	0	0	0	0	0	0	14	0	0	0
C2	0	155	0	0	0	0	37	1	0	0	0	0	0
C3	29	0	171	0	0	0	0	0	6	0	0	0	0
C4	33	0	41	121	1	0	0	3	0	0	0	3	0
C5	0	0	0	3	107	0	0	0	1	0	0	0	0
C6	52	21	0	0	0	106	0	0	0	0	0	0	0
C7	0	0	0	0	0	0	55	0	0	0	0	0	0
C8	0	0	0	0	0	0	0	220	161	0	0	0	0
C9	0	0	0	0	0	0	0	0	470	0	0	0	0
C10	0	0	0	0	0	0	0	0	196	141	0	17	0
C11	0	0	0	0	0	0	0	0	0	0	369	0	0
C12	0	0	0	0	0	0	0	0	0	0	0	453	0
C13	0	0	0	0	0	0	0	0	0	0	0	0	877

shown in Table 6.7, using the same hyperspectral datasets. We replicated the highest OA that those methods can achieve in the publications. Jia et al. [68] used a supervised approach, and others used semi-supervised approaches. Also, all methods but Dópido et al. [33] used spatial context information from either texture features or the random field model. The results show that our ST-IRGS algorithm achieves highest OA for both datasets. For the University of Pavia dataset, ST-IRGS achieves 5.4% higher than the method by Dópido et al. [33] that used 10 labeled samples for each class, in spite of the possibility of further improvement of their method by incorporating spatial context information. For the Salinas dataset, ST-IRGS achieves 8.2% higher OA than the method by Li et al. [89], and 6.0% higher OA than MBRF-CRF-E. For the Kennedy Space Center dataset, ST-IRGS achieves 8.2% higher OA than the method based on sparse representation proposed by Haq et al. [123] which does not consider spatial context. MBRF-CRF-NE with the optimal weight parameter based on the test accuracy achieves 2.5% higher OA than ST-IRGS.

6.5 Summary

A semi-supervised joint classification and segmentation algorithm named ST-IRGS was presented and applied to hyperspectral image classification. This algorithm is based on

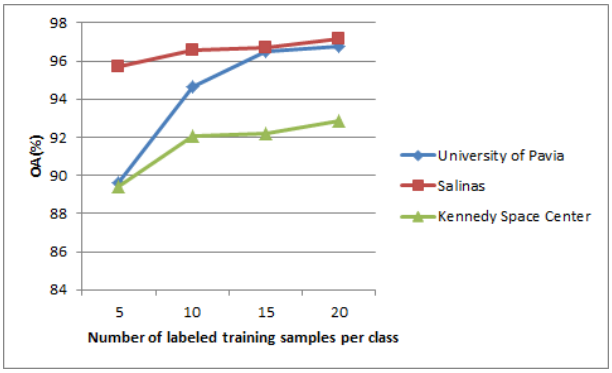


Figure 6.11: Classification results by ST-IRGS using different labeled training samples.

the CRF model, and is capable of combining semi-supervised classification, region merging, and spatial context model into a single framework. The multimodal Gaussian model is used as the unary potential in the CRF model, and the edge penalty function is incorporated into the pairwise potential. The self-training technique is adopted to incorporate unlabeled samples. The selection of new samples is only from regions that contain the original training samples, which is more reliable than just based on the predictions of the unlabeled samples. The information of intensity and edge strength can be implicitly used for the selection due to the region merging criterion. Experiments on three benchmark hyperspectral images indicate that the ST-IRGS algorithm outperforms other methods without self-training, and achieves reasonable classification accuracy when there are few labeled samples available.

Table 6.7: Comparison of MBRF-CRF, ST-IRGS, and state-of-the-art methods for limited labeled training samples using the same datasets. The highest OA achieved by other methods are replicated from the publications.

Methods	labels per class	supervision	spatial context	OA	AA	[κ]
University of Pavia dataset						
Jia et al. [68]	5	supervised	Gabor texture	75.2	80.6	0.69
Wang et al. [152]	5	semi-supervised	Gabor texture	69.6	\	\
Dópido et al. [33]	10	semi-supervised	no	84.1	87.4	0.80
MBRF-CRF-E	5	supervised	CRF	83.7	86.9	0.79
ST-IRGS	5	semi-supervised	region merging + CRF	**89.5**	**91.5**	**0.92**
Salinas dataset						
Jia et al. [68]	5	supervised	Gabor texture	84.1	87.2	0.82
Wang et al. [152]	5	semi-supervised	Gabor texture	86.8	\	\
Li et al. [89]	5	semi-supervised	MRF	87.2	\	\
MBRF-CRF-E	5	supervised	CRF	89.4	91.8	0.88
ST-IRGS	5	semi-supervised	region merging + CRF	**95.4**	**95.9**	**0.96**
Kennedy Space Center dataset						
Haq et al. [123]	5	supervised	no	81.2	\	\
MBRF-CRF-NE	5	supervised	CRF	**91.9**	**89.3**	**0.91**
ST-IRGS	5	semi-supervised	region merging + CRF	89.4	86.3	0.86

Chapter 7

Extension of ST-IRGS for automated interpretation of SAR sea ice imagery using multiple image features

7.1 Introduction

The operational mapping of sea ice is beneficial for several important purposes, including ship navigation, weather forecasting, and environmental science [169]. Among many satellite sensors, SAR sensors are very suitable for sea ice mapping because of the all-weather and all-day imaging capability. Also, the RADARSAT-2 satellite makes dual-band polarization on large scale available, so that more information can be provided for the operational sea ice mapping.

In the Canadian Ice Service (CIS) [37], the interpretation of sea ice imagery is currently performed manually by ice analysts everyday. An example of the manual ice chart is shown in Fig. 7.1 [37]. The manually-outlined irregular polygons are "egg codes" [37] defined by World Meteorological Organization, which record the ice conditions such as the ice concentration in the polygons estimated by the ice experts based on their experiences. The ice chart provides interpretation in a large scale, but lacks because the ice-water boundaries are coarsely outlined, and ice types are only estimated by fraction. Also, the result may not be consistent for different ice analysts [102]. By comparison, automated interpretation approaches have the potential of generating high-volume pixel-level classification maps with no inter-operator bias.

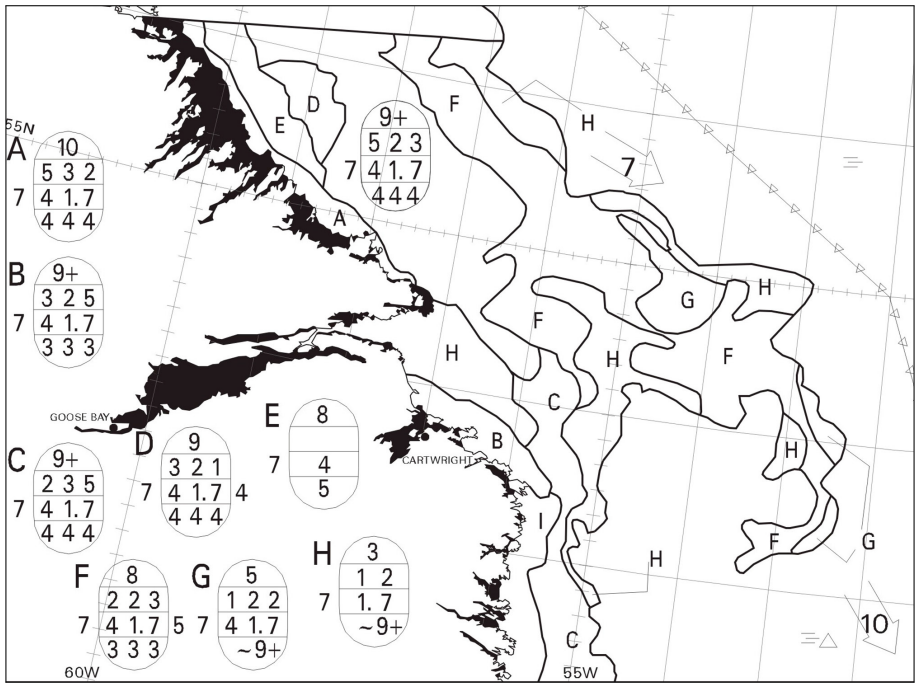


Figure 7.1: An example of an ice chart [37] and the egg codes, recording the total ice concentration, the partial ice concentration, the stage of ice development, and the form of ice in the polygon.

The automated interpretation of SAR sea ice imagery has been studied for over a decade [29,73,80,84,136,169]. However, there is no known algorithm that generates labeled maps that can be used operationally for this purpose. There are mainly two reasons. First, the backscatter of ice and open water in the SAR imagery are affected by many factors such as incidence angle, snow condition, and wind speed, resulting in tremendous within-class variability across scenes and even within a scene. Also, SAR imagery is corrupted by speckle noise [83], which degrades the image details and further decreases class separability. As a result, traditional pixelwise classification methods [73, 80] are incapable of achieving accurate classification. Second, traditional supervised classification methods require sufficient labeled samples for training. However, the ground-truthing requires the expertise and experience of interpreting sea ice, a time-consuming and tedious task considering the large size of SAR images. Otherwise, if only a small number of pixels are labeled for training, there may be insufficient training samples for characterizing the true data distribution which leads to inferior classification results.

In this chapter, ST-IRGS, which has been described in Chapter 6, is applied to SAR sea ice imagery. Compared to the hyperspectral images for testing in the early chapters, SAR images have larger image size, which results in high computational complexity and slow convergence of some labeling models such as MRF. Therefore, grouping pixels into regions can help decrease computation complexity and avoid suboptimal solutions [170]. Also, region-based methods enable the extraction of high-level features which are more resistent to noise. A common way is to first generate over-segmentation result, and then iteratively merge similar regions based on region information such as intensity and boundary [13, 91, 144, 166]. These methods are largely dependent on a good merging criterion. As described in Chapter 6, ST-IRGS is a region merging algorithm that inherits useful properties from IRGS [169]. Unlike traditional region merging methods, ST-IRGS is integrated with a CRF to iteratively reduce the number of nodes, and edge strength is used in both classification and region merging. The key feature of the ST-IRGS is an embedded self-training procedure. Compared to a similar approach [33], the ST-IRGS can iteratively expand the training candidate set owing to its region merging property, so that the abundant unlabeled samples can be explored even if they are not near the original training samples. Also, the correctness of the labels can be ensured by the region properties.

The rest of the chapter is structured as follows. Section 7.2 describes the proposed ST-IRGS algorithm in the context of the ice-water labeling problem. Experiments on a RADARSAT-2 SAR dual-polarization dataset are reported in Section 7.3. Section 7.4 summarizes the chapter.

7.2 Methodologies

7.2.1 Problem formulation

Given a few labeled pixels in an image, we aim to learn the optimal labeling configuration of all the pixels in the image. For a first-order CRF model, the posterior probabilities $P(\mathbf{y} \mid \mathbf{x})$ can be formulated as [59, 132]:

$$\log P(\mathbf{y} \mid \mathbf{x}, \theta) = \sum_{s} \phi_s(y_s, \mathbf{x}^{\phi}; \theta^{\phi}) + \beta \sum_{s} \sum_{t \in \eta_s} \xi_{st}(y_s, y_t, g_{st}(\mathbf{x}^{\xi}); \theta^{\xi}) - \log Z(\theta, \mathbf{x}) \tag{7.1}$$

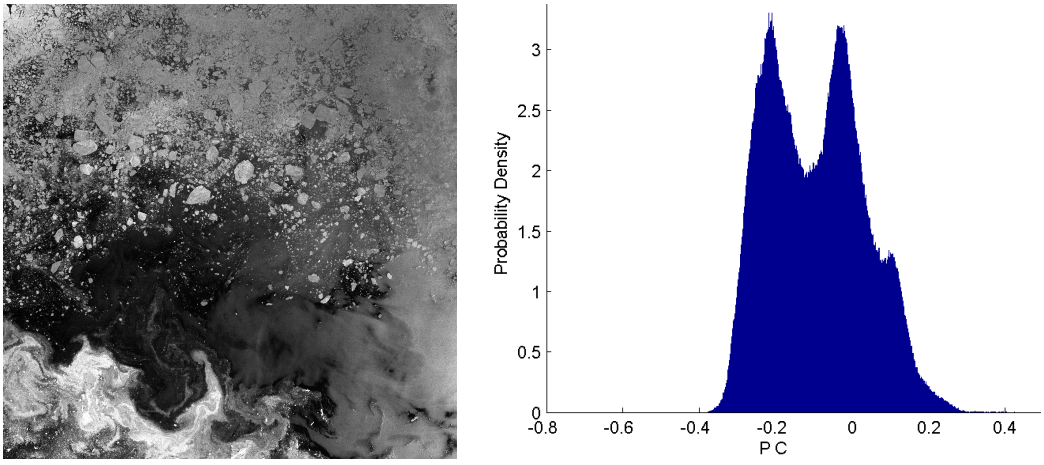
where $\phi_s(\cdot)$ and $\xi_{st}(\cdot)$ are unary and pairwise clique potentials respectively, s indexes nodes in a discrete rectangular lattice, η_s refers to the 4-connected neighbors of node s, $g_{st}(\mathbf{x}^{\xi})$ is the edge feature for two adjacent nodes s and t, $\mathbf{Z}(\theta, \mathbf{x})$ is a partition function, $\{\mathbf{x}^{\phi}, \mathbf{x}^{\xi}\} \subset \mathbf{x}$ are features for the potentials, $\mathbf{y}$ is the label configuration, β is a weight parameter between unary and pairwise potentials, and $\theta = \left\{\theta^{\phi}, \theta^{\xi}\right\}$ are model parameters.

Compared to Eq. (5.11), different features are used for the unary and pairwise potentials. We use a combination of backscatter and texture features for $\mathbf{x}^{\phi}$ in the unary potentials. In the previous literature, GLCM parameters have been demonstrated to be effective in distinguishing different ice types and open water [24]. In our approach, we adopt a total of 28 features including the mean, standard deviation, and GLCM measures in different window sizes extracted for ice-water classification using a RADARSAT-2 image dataset [84].

The unary potentials are defined using Gaussian models. For SAR sea ice imagery, a single Gaussian mode may be insufficient to model a class even within a scene. Fig. 7.2 shows an example that the principal component of the water class has multiple mixtures. Therefore, we use MGMLC that has been introduced in Chapter 6 as the unary classifier. The unary potentials are defined as:

$$\phi_s(y_s, \mathbf{x}^{\phi}; \alpha_k, \mu_{\mathbf{k}}, \mathbf{\Sigma}_{\mathbf{k}}, \mathcal{C}_{y_s}) = \log \left\{ P(y_s) \sum_{k=1}^{\mathcal{C}_{y_s}} \alpha_k \mathbb{N}(\mathbf{x}_s^{\phi}, \mu_{\mathbf{k}}, \mathbf{\Sigma}_{\mathbf{k}}) \right\} \tag{7.2}$$

where $P(y_s)$ is a class prior, $\mathcal{C}_{y_s}$ is the number of mixtures in the class, $\mu_{\mathbf{k}}$ and $\mathbf{\Sigma}_{\mathbf{k}}$ are the mean and variance of the k^{th} mixture, and α_k is the mixture prior in the class $\mathcal{C}_{y_s}$. Although other classifiers such as logistic regression [89] and SVM [178] can be used as the unary



(a) (b)

Figure 7.2: (a) RADARSAT-2 SAR imagery in HH polarization captured over Beaufort Sea on July 30, 2010. (b) Probability density function of the first principal component (PC) of the 28 features for the water class, which clearly shows multiple mixtures due to the incidence angle effect.

classifier, the multimodal Gaussian model performs very well for ice-water classification in the experiment given sufficient training samples, and the estimated mixture parameters can be used for region merging and determining the weight parameter later.

For $\mathbf{x}^{\xi}$ in the pairwise potentials, we use the same edge penalty function as the original IRGS algorithm [170]. However, we only use the backscatter in the HV polarization which is less sensitive to incidence angle effect compared to the HH polarization [84]. The edge feature is set to measure the gradient between neighboring pixels [170]. Thus, the pairwise potential is defined as:

$$\xi_{st}(y_s, y_t, \mathbf{x}) = \begin{cases} \beta g_{st}(\mathbf{x}_{\xi}) & y_s \neq y_t \\ 0 & \text{otherwise} \end{cases} \tag{7.3}$$

$$g_{st}(x^{\xi}) = exp\left[-\left(\frac{x_s^{\xi} - x_t^{\xi}}{K}\right)^2\right] \tag{7.4}$$

where β is a weight parameter, and K is a gradually increasing parameter that is calculated using Eq. (6.11).

Fig. 7.3 shows an overview of the ice-water labeling framework.

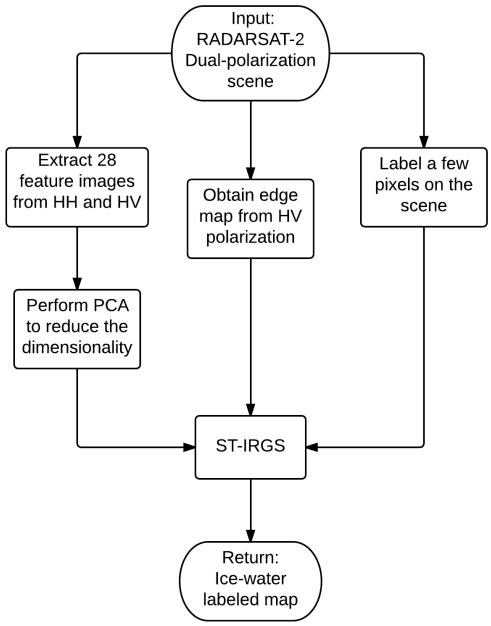


Figure 7.3: Overview of the ice-water labeling framework.

7.2.2 Problem solution using ST-IRGS

The solution to Eq. (7.1) is mostly the same as that in Chapter 6. An alternating procedure is used to perform both parameter estimation and inference, as shown in Fig. 7.4. In each iteration, the model parameters are first fixed, and the label configuration is optimized. Then, the model parameters are updated based on the current labels. In the parameter estimation step, the parameters related to the unary potentials are estimated independently. The number of mixtures is estimated using BIC [128], and then α_k, $\boldsymbol{\mu}_\mathbf{k}$, and $\boldsymbol{\Sigma}_\mathbf{k}$ are estimated using EM. After the parameters are estimated, Gibbs sampling [8] is used for inference. Each node is processed only once in one iteration.

Similar to the IRGS algorithm [170], we incorporate the hierarchical region growing procedure into the iterations to build a hierarchical data-adaptive structure to make the optimization more efficient. The watershed algorithm [149] is first performed to obtain the over-segmented regions. Then, similar neighboring pixels are iteratively merged into regions, so that the number of nodes can be significantly reduced, and thus the convergence

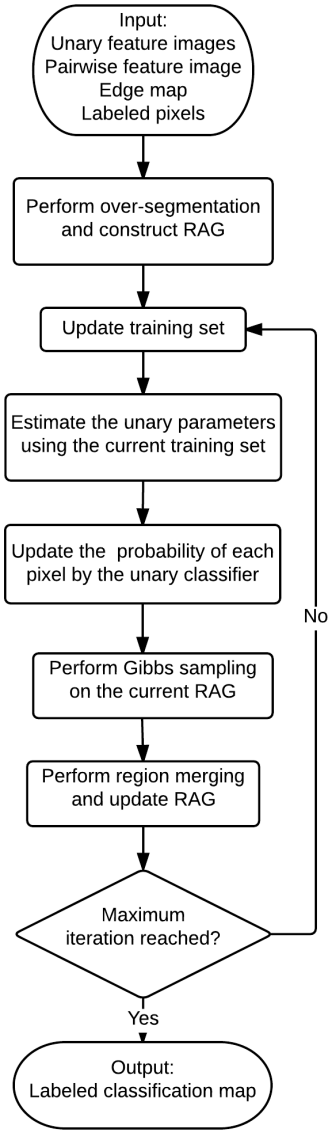


Figure 7.4: Flow chart of the ST-IRGS algorithm

rate can be increased. In our approach, we assume the distribution of HV backscatter in each region to be unimodal Gaussian distribution. Even if the distribution of features in a SAR image is not strictly Gaussian, the Gaussian models can still be used to approximate it [29, 169]. Dong et al. [32] compared segmentation results using Gaussian-MRF and Gamma-MRF, and showed that the Gaussian-MRF model is even more stable and reliable while the Gamma-MRF model mistakenly merges some small segments. The same merging criterion as the IRGS is used [169]:

$$\partial E_{ij} = \sum_{s \in \Omega_k} \log(\Sigma_k) - \sum_{s \in \Omega_i} \log(\Sigma_i) - \sum_{s \in \Omega_j} \log(\Sigma_j) - \beta \sum_{\langle s,t \rangle \in C; s \in \Omega_i, t \in \Omega_j} g_{st}(x^{\xi}) \tag{7.5}$$

where ∂E_{ij} is the energy difference between merging two regions i and j and not merging them, Ω_i and Ω_j are two neighboring regions for merging, Ω_k is the region after merged, C is the set of pixels in the discrete rectangular lattice, and Σ_i is the variance in the region i. The detailed derivation is found in a previous publication [170].

To preserve the single-mode property of the regions, each region is assigned a mixture label based on the previous estimated number of mixtures and Gaussian parameters. Only regions that are in both the same predicted class label and mixture label are allowed to be merged. In each iteration, the edges of adjacent nodes with negative ∂E_{ij} are put into a merging list in an ascending order to make the merging more efficient, and the corresponding region and edge information is updated after each merging. Also, each node is only allowed to be merged once in early iterations to avoid extremely-large regions that may result in imbalanced number of training samples. For SAR sea ice imagery and other remote sensing images that contain homogeneous regions, the number of nodes can be finally reduced to a very small number.

To address the problem of insufficient training samples, a self-training procedure is followed by region merging in each iteration. Self-training is a semi-supervised technique that iteratively retrains the classifier using the predictions that are confident [181]. However, traditional self-training methods are largely dependent on the performance of the classifier. If the classifier performs very badly, the predictions are not reliable and will degrade the classification performance if they are used for retraining. In our approach, we only select pixels which are in the same regions as the original training samples to be new training samples.

Another important issue is to determine the weight parameter β between the unary and pairwise potentials. The optimum weight parameter can be selected using a grid search, but it is time-consuming and unreliable when only limited training samples are available.

In the ST-IRGS, we use the scheme for adapting the weight parameter of the IRGS. The parameter β is updated based on class separability in each iteration:

$$\beta = C_1 \frac{J/C_2}{1 + J/C_2} \beta_0 \tag{7.6}$$

where β_0 is the ratio of the current total class boundary length over the image size [117], C_1 and C_2 are constants, and J is the separability of ice and water classes. See thesis by Yu for details [168].

To calculate the pairwise class separability J, we still use the Fisher's criterion, but we need to sum up the values of all the pairwise mixtures because each class has multiple mixtures:

$$J = \sum_{i \in \mathcal{C}_{\text{ice}}} \sum_{j \in \mathcal{C}_{\text{water}}} \left\{ \alpha_i \alpha_j \frac{|\mu_i - \mu_j|}{\Sigma_j^2 + \Sigma_j^2} \right\} \tag{7.7}$$

where μ_i and Σ_i are the mean and variance of the HV backscatter for pixels that belong to the mixture i.

In each iteration, the β and K in Eq. (7.3) and Eq. (7.4) are updated correspondingly until a maximum number of iterations $\tau_{\max}$ is reached.

7.3 Experiments

7.3.1 Experimental setup

The SAR datasets for testing are described in Section 3.2. Scenes that contain only ice or only open water are excluded, and 12 scenes, each of which contain both ice and open water are used for testing. The ST-IRGS algorithm is implemented in Microsoft Visual C++ 2010. C_1 and C_2 in Eq. (7.6) are set to 3 and 0.4 respectively, as suggested in the original IRGS algorithm [169]. We use the same weight parameter for all the test images considering its adaptability, even though the accuracy could be improved by carefully tuning the parameters for each scene. $\tau_{\max}$ is set to 100. The experiments show that there is no further change of the label configurations after 100 iterations. The setting of K in Eq. (7.4) is the same as the IRGS algorithm [170]. The maximum number of mixtures $\mathcal{C}_i$ for each class i is set to 5. When the number of training samples in a class is less than 200, a

single-mode Gaussian model is used in order to guarantee sufficient samples for estimating the model parameters. To reduce the computational cost, we randomly select 5000 samples for parameter estimation in the EM algorithm if the number of expanded training samples for a class exceeds 5000. Empirically the selection of more training samples does not show additional improvement. Also, the parameters are only updated when the available training samples for a class increase by 2%. In the first 30 iterations, each region is only allowed to be merged once to avoid class imbalance, as mentioned in Chapter 6.

At the beginning of the experiment, only ten pixels for each class are randomly selected for training samples. We use MGMLC, GMRF, and SVM-IRGS [84] for comparison. Both MGMLC and GMRF are closely related to the proposed ST-IRGS algorithm for comparison. MGMLC uses the pixels in the same watershed algorithm as the original training samples for parameter estimation, and it serves as the unary classifier for both GMRF and ST-IRGS. GMRF combines MGMLC with the standard MRF model, and the graph-cut algorithm [9] is used for inferencing the labels. The weight parameter of GMRF adopts the parameter with highest test accuracy in a set $\{2^0, 2^1, 2^2, ..., 2^8\}$. For all the methods, PCA is first applied to the 28 features and the first five principal components are used. SVM-IRGS [84] combines SVM and IRGS into a single framework, and uses the leave-one-out training scheme, which is fundamentally different from the ST-IRGS algorithm. However, it can still be used as a reference because it has been tested on the same dataset.

7.3.2 Experimental results and analysis

The classification result of the whole test dataset is shown in Table 7.1. Each image is tested for 10 times, and the mean and standard deviation of the OA for each scene are shown in the table. The ST-IRGS achieves about 92% classification accuracy, which is significantly higher than MGMLC and GMRF, and only a little lower than SVM-IRGS. Moreover, the variation of the OA due to random sampling is significantly reduced by the ST-IRGS. This is because as the training set is expanded, the classification performance becomes less sensitive to the initial training samples. Also, ST-IRGS only achieves slightly lower OA than SVM-IRGS which requires a large number of labeled samples from other scenes to train the SVM model.

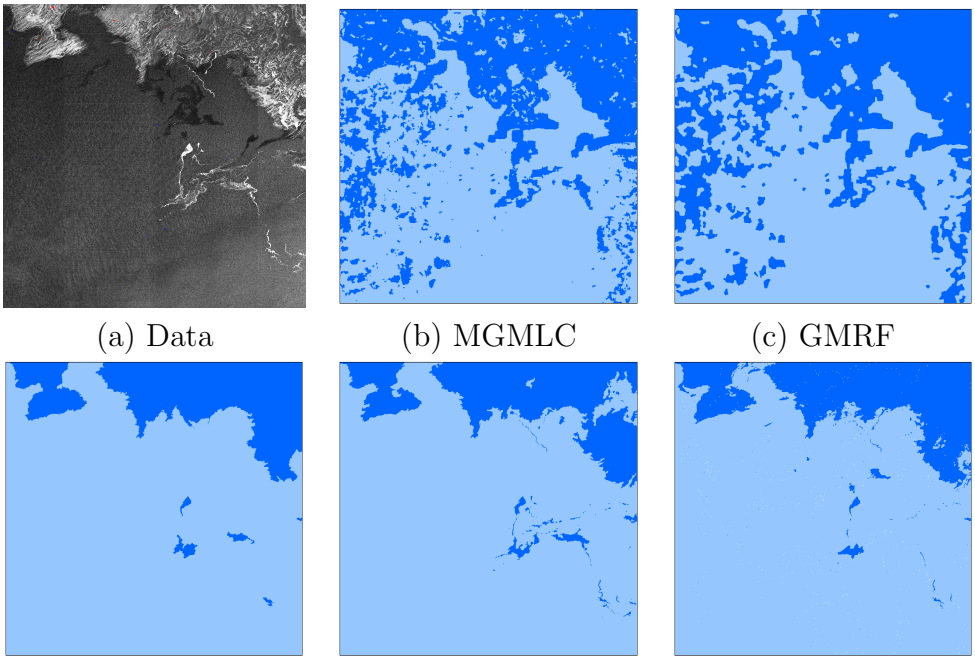
Fig. 7.5 shows the result on a RADARSAT-2 image scene captured over the Beaufort Sea on October 3, 2010. The ice usually starts to freeze in the Beaufort Sea in early October. The air temperature was -1.0°C on this date [36]. In Fig. 7.5 (a), different stages of ice growth including new ice, grey ice, and grey-white ice are observed. There is also

Table 7.1: Overall classification accuracy (OA%) and kappa coefficient for MGMLC, GMRF, ST-IRGS, and SVM-IRGS on a RADARSAT-2 SAR dual-polarization dataset. The results by SVM-IRGS are replicated from the previous publication [84].

Date	MGMLC	GMRF	ST-IRGS	*SVM-IRGS [84]*
20100524	82.92±9.89	84.27±13.79	97.19±1.73	***98.36***
20100623	94.84±3.90	92.69±7.75	**98.70**±0.16	*98.45*
20100629	79.41±13.61	79.98±15.17	**93.36**±1.47	*93.25*
20100712	70.10±10.01	75.24±10.17	85.30±5.01	***92.03***
20100721	78.96±8.13	83.45±6.67	89.39±1.09	***91.54***
20100730	69.35±5.14	69.18±5.78	84.77±2.36	***90.31***
20100807	73.28±7.82	78.50±4.20	85.71±1.39	***89.95***
20100909	78.19±16.18	83.64±9.48	86.60±1.52	***94.12***
20101003	75.01±13.08	79.92±10.23	**97.62**±0.26	*94.97*
20101021	75.37±17.12	88.43±6.45	**97.64**±0.31	*97.24*
20101027	74.73±10.76	79.75±6.37	93.71±0.65	***94.64***
20101114	78.19±6.45	82.45±9.10	**96.03**±0.57	*95.62*
Average	77.53±10.17	81.46±8.76	92.17±1.38	***94.21***

some multi-year ice in the north east. It had survived the summer and began to build up together with the first-year ice that is freezing. The wind speed at PRDA2 tower was 12.6 m/s [36], which results in the wind-roughened texture of the open water on the image.

Due to the high wind and the incidence angle effect, some areas of the open water have similar properties to ice in both intensity and texture. In Fig. 7.5 (b), the rough water area in the west and the open water in the near range in the south east are misclassified as ice. Conversely, some newly-formed ice that has similar backscatter to water is also misclassified. Moreover, class boundaries are not accurate because the edges of the texture images are blurred due to the large GLCM window size used. This is unavoidable because if a smaller texture window size is used, some macroscopic texture patterns might fail to be extracted. In Fig. 7.5 (c), GMRF can refine some of the labeling with the aid of the spatial context, but in some open water areas, there is even more misclassification due to the poor estimation of the unary classifier, as shown in the west of Fig. 7.5 (b). Also, there is no improvement on the correction of the class boundaries. In Fig. 7.5 (d), the classification result is significantly improved due to the expanded self-training set by the self-training technique. An average of 97.2% overall classification accuracy can be achieved in 10 tests using different randomly-selected samples. Also, the class boundaries are corrected after the incorporation of edge strength into the CRF energy function. SVM-IRGS in Fig. 7.5 (e) can capture the boundary between open water and small pieces of ice in the middle,



(a) Data (b) MGMLC (c) GMRF

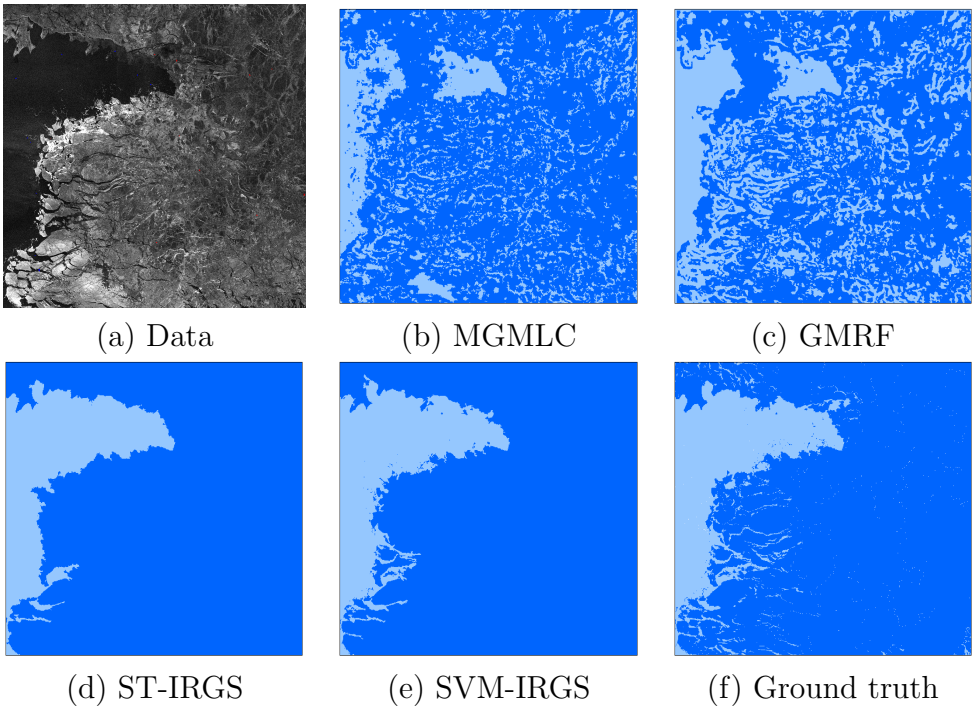
(d) ST-IRGS (e) SVM-IRGS (f) Ground truth

Figure 7.5: Classification results of the data captured on October 3, 2010. (a) shows the HV polarization of the data, with original training samples overlain on the image (dark blue: ice; light blue: open water). (b), (c), (d), and (e) are classification results of MGMLC, GMRF, ST-IRGS, and SVM-IRGS respectively. (f) is the ground truth.

but there is misclassification of ice into water in the north east.

Beyond the improvement, there are two obvious errors in Fig. 7.5 (d). First, the grease ice in the middle of the image is mostly misclassified. Even though a human interpreter is able to identify some grease ice by its difference from the surrounding open water, the feature space of grease ice is overlapped with that of open water in other areas of the scene. The correct labeling of the grease ice may be at the cost of misclassifying some open water, as shown in Fig. 7.5 (b) and (c). Also, the small pieces of ice in the south east are labeled as open water. Due to the texture window size, there is little difference between those small ice floes and the open water in the texture feature space. Also, the small regions tend to be considered as noise in a spatial context model. Nevertheless, neither of these misclassifications incurs significant operational issues [84].

Another result is shown in Fig. 7.6, using the scene captured over the Chukchi sea on November 14, 2010. The air temperature was -15.1°C. The wind speed measured at



(a) Data (b) MGMLC (c) GMRF

(d) ST-IRGS (e) SVM-IRGS (f) Ground truth

Figure 7.6: Classification results of the data captured over Chukchi sea on November 14, 2010. (a) shows the HV polarization of the data, with original training samples overlaid on the image (dark blue: ice; light blue: open water). (b), (c), (d), (e) are classification results of MGMLC, GMRF, ST-IRGS, and SVM-IRGS respectively. (f) is the ground truth.

PRDA2 tower was 6.7 m/s. In mid November, the ice coverage is increasing towards the south west in the Chukchi Sea, and new ice starts to form. In this image, there is a mixture of first-year ice, grey ice, and grey-white ice in the north east. Some ice area has very dark intensity, and is difficult to be distinguished from open water from in a small scale. As a result, MGMLC that is only based on small watershed regions is unable to achieve satisfactory classification result. GMRF can correct some misclassifications of open water into ice, but does not help improve the classification accuracy of ice. The ST-IRGS correctly classifies most of the pixels by expanding the training set, and achieves an OA of 96.0% over 10 tests. The SVM-IRGS achieves an OA of 95.6%, but it has more accurate ice/water boundary.

Fig. 7.7 shows the change of the OA during the iterations for the dataset. The OA fluctuates at the beginning of the iterations because even though the number of training samples increases, they are still incapable of characterizing the whole image. Once new

samples are incorporated, the classifier will change greatly. Such an intermediate result is not reliable. If the self-training is only based on this result without the region constraint, the subsequent result might be even worse. Instead, the ST-IRGS only trusts the predicted labels in the regions that contain the original training samples based on region merging, in order to ensure the correctness of the self-training samples. After 30 iterations, the OA starts to increase as more samples are added into the training set, and become stable both after 80 iterations.

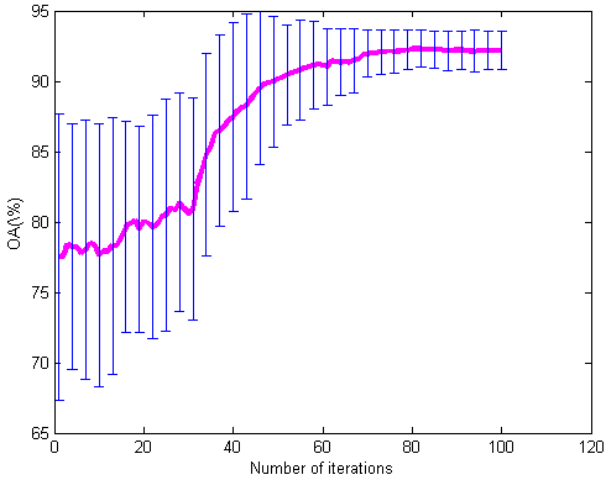


Figure 7.7: Overall classification accuracy by ST-IRGS during its iterations, averaged over all the test images.

In Fig. 7.8, the number of regions is gradually reduced from about 100,000 at the beginning to less than 400 at the end of the iterations. There are two advantages. First, reducing the number of regions makes the optimization more efficient and can help extract high-level features. Second, it is very convenient to correct the classification results manually by re-labeling the misclassified regions using the region maps in any previous iteration.

7.4 Summary

The ST-IRGS algorithm is capable of effectively distinguishing ice and open water in large-scale dual-polarization SAR images using a very small number of labeled samples. The multimodal Gaussian model is suitable for describing the class distributions considering the complexity of the SAR sea ice data. The inherent combination of self-training in

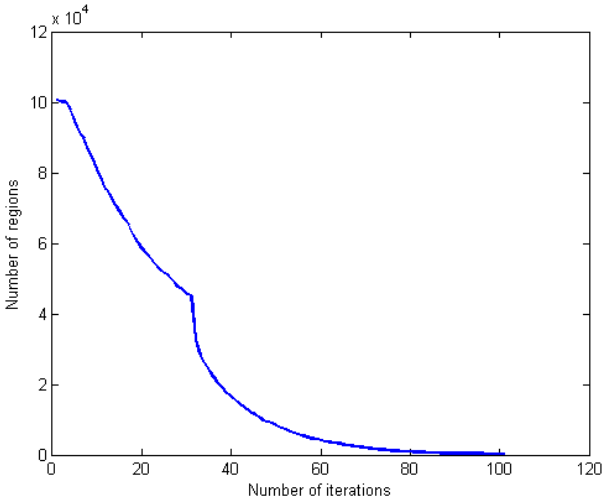


Figure 7.8: Number of regions by ST-IRGS during the iterations, averaged over all the test images.

the IRGS framework can iteratively and correctly expand the training set and improve the classifier. Robust classification results have been achieved on a RADARSAT-2 dual-polarization dataset captured over the Beaufort and Chukchi Sea areas.

Chapter 8

Conclusions

In this chapter, the summary and research contributions for the thesis are provided followed by a few suggestions for the directions of future research.

8.1 Summary

This thesis systematically studies automated interpretation of remote sensing imagery with limited labeled training data. Traditional automated interpretation techniques usually consist of three separate steps, which are sequentially feature extraction, classification, and segmentation. In this thesis, three methods are proposed, and each of them focuses on one of the three steps. A summary of the three methods is shown in Table 8.1.

8.2 Research contributions

The specific contributions in the four main chapters of this thesis are described as follows.

- In Chapter 4, we have demonstrated multiple localized manifolds outperform a single manifold for high-dimensional data with limited training samples. We proposed the ELML framework for joint feature extraction and classification based on localized manifold learning. Multiple linear manifolds emphasizing on different locations of the feature space are learned to characterize the input data, and a classification ensemble is then trained using the features extracted via the different manifolds.

Table 8.1: Summary of three proposed methods (ELML, MBRF-CRF, and ST-IRGS).

	ELML	MBRF-CRF	ST-IRGS
Feature extraction	**Localized manifolds**	PCA (no dimension reduction)	PCA
Classification	k-NN	**MBRF**	MGMLC
Segmentation	No	CRF	**CRF+region merging**
Supervision	Supervised/semi-supervised	Supervised	Semi-supervised

Such manifolds are localized in the feature space, and can overcome the challenges and limitations associated with learning a single global manifold for characterizing complex data structures. Two feature extraction methods, including a supervised method (NWFE) and a semi-supervised method (SELD), are modified into localized methods (L-NWFE and L-SELD). The effectiveness of the proposed algorithm is illustrated using six feature extraction methods including NWFE and SELD for comparison on hyperspectral datasets.

- In Chapter 5, we investigate the performance of different classifiers using limited labeled training data in a supervised manner. We focus on the ensemble learning techniques, which are particularly effective for high-dimensional data and the SSS problem. An enhanced ensemble method is developed, which combines the rotation forest algorithm and a multiclass AdaBoost algorithm. The benefit of the combination can be explained by the bias-variance analysis, especially in the situation of inadequate training samples and high dimensionality. MBRF can be considered as a joint feature extraction and classification method due to its inherent feature extraction step using PCA. But different from ELML introduced in Chapter 4, the dimensionality is not reduced. Further, MBRF innately produces posterior probabilities by the embedded multi-class boosting algorithm, which serve as the unary potentials of the CRF model to incorporate spatial context information. The effectiveness of the proposed MBRF and its combination with CRF is shown by comparing with several state-of-the-art classification methods on hyperspectral datasets.

- Chapter 6 and Chapter 7 propose the ST-IRGS algorithm, which is a semi-supervised joint classification and segmentation algorithm. The algorithm is derived from the CRF framework to incorporate spatial context. GMM is used to estimate the probabilities for the unary potentials, and the edge strength is naturally incorporated into the energy function due to the flexibility of CRF. Unlike traditional methods based on random fields, region merging is concatenated with the CRF inference to reduce the number of nodes iteratively, and the merging criterion is designed to minimize the global energy function. Moreover, the self-training technique is used,

which iteratively enlarges the training sample set and retrains the classifier. The selection of training samples is based on region merging, so that the risk of assigning wrong labels can be reduced. Chapter 7 extends the ST-IRGS algorithm to ice-water classification using dual-polarization RADARSAT-2 imagery. Compared to the hyperspectral imagery used in Chapter 6, the SAR imagery for testing in Chapter 7 has a larger size and more complex data properties. Therefore, texture information needs to be incorporated into the CRF model. The performance of the ST-IRGS algorithm has been tested on both hyperspectral and SAR datasets.

8.3 Suggestions of future research directions

The work motivates future research in the following directions.

- ELML in Chapter 4 is more like a framework than a specific algorithm. For example, the k-means algorithm can be replaced by any other clustering algorithm, and the base classifier (1-NN) can be replaced by any supervised or semi-supervised classifier. The feasibility of this framework has been shown in this thesis, while a comprehensive study of different clustering algorithms and classifiers can be further investigated. Also, spatial context can be further incorporated to improve classification accuracy using MRF or CRF as a post-processing step if 1-NN is replaced by other classifiers that can generate probability estimates;

- The main problem of using random field models for limited training samples is the difficulty of determining the weight parameter, because the result by grid search is not reliable. Therefore, it is worthwhile developing automated techniques for choosing the weight parameters. In ST-IRGS, we use an automated technique to determine the weight parameter based on class separability. In the previous literature, it can also be determined using the Ho-Kashyap algorithm [129]. Automated approaches for determining the weight parameter can be further investigated in future work.

- ST-IRGS in Chapter 5 & 6 uses a Gaussian model as the unary classifier, because it works well with ST-IRGS and is suitable for the test datasets in the experiments. For some data whose distribution cannot be approximated as a single or a multimodal Gaussian distribution, other classifiers such as SVM, MLR, or the proposed MBRF method might be more appropriate. If non-Gaussian classifiers are used, the merging criterion and the class separability measure need to be modified because in ST-IRGS they are both based on Gaussian parameters. In the operational use, it is helpful to

incorporate more classifiers into the software system, so that users can decide which one to be used. Also, it might be more helpful if the most appropriate classifier can be automatically selected based on the characteristics of the data.

References

[1] O. Aytekin, M. Koc, and I. Ulusoy, *Local primitive pattern for the classification of SAR images*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 4, 2431–2441.

[2] A. Baraldi and F. Pianigiani, *An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters*, IEEE Transactions on Geoscience and Remote Sensing **33** (1995), no. 2, 293–304.

[3] Basque University, *Hyperspectral remote sensing scenes*, 2015, [Online; accessed 13-May-2015].

[4] G. Baudat and F. Anouar, *Generalized discriminant analysis using a kernel approach*, Neural computation **12** (2000), no. 10, 2385–2404.

[5] M. Belkin and P. Niyogi, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural computation **15** (2003), no. 6, 1373–1396.

[6] D. Benboudjema and W. Pieczynski, *Unsupervised statistical segmentation of nonstationary images using triplet Markov fields*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **29** (2007), no. 8, 1367–1378.

[7] J. A. Benediktsson, M. Pesaresi, and K. Arnason, *Classification and feature extraction for remote sensing images from urban areas based on morphological transformations*, IEEE Transactions on Geoscience and Remote Sensing **41** (2003), no. 9, 1940–1949.

[8] C. M. Bishop, *Pattern recognition and machine learning*, springer New York, 2006.

[9] Y. Boykov, O. Veksler, and R. Zabih, *Fast approximate energy minimization via graph cuts*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **23** (2001), no. 11, 1222–1239.

[10] L. Breiman, *Bagging predictors*, Machine Learning **24** (1996), no. 2, 123–140.

[11] ———, *Random forests*, Machine Learning **45** (2001), no. 1, 5–32.

[12] G. J. Briem, J. A. Benediktsson, and J. R. Sveinsson, *Multiple classifiers applied to multisource remote sensing data*, IEEE Transactions on Geoscience and Remote Sensing **40** (2002), no. 10, 2291–2299.

[13] L. Bruzzone and L. Carlin, *A multilevel context-based system for classification of very high spatial resolution images*, IEEE Transactions on Geoscience and Remote Sensing **44** (2006), no. 9, 2587–2600.

[14] L. Bruzzone, M. Chi, and M. Marconcini, *A novel transductive SVM for semisupervised classification of remote-sensing images*, IEEE Transactions on Geoscience and Remote Sensing **44** (2006), no. 11, 3363–3373.

[15] G. Camps-Valls, T. Bandos Marsheva, and D. Zhou, *Semi-supervised graph-based hyperspectral image classification*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 10, 3044–3054.

[16] G. Camps-Valls and L. Bruzzone, *Kernel methods for remote sensing data analysis*, vol. 26, Wiley Online Library, 2009.

[17] G. Camps-Valls, D. Tuia, L. Bruzzone, and J. Atli Benediktsson, *Advances in hyperspectral image classification: Earth monitoring with statistical learning methods*, IEEE Signal Processing Magazine **31** (2014), no. 1, 45–54.

[18] Canadian Ice Service, *MANICE: Manual of standard procedures for observing and reporting ice conditions*, Meteorological Service of Canada (Ottawa, Canada), 2005.

[19] E. A. Carvalho, D. M. Ushizima, F. N. Medeiros, C. I. O. Martins, R. C. Marques, and I. Oliveira, *SAR imagery segmentation by statistical region growing and hierarchical merging*, Digital Signal Processing **20** (2010), no. 5, 1365–1378.

[20] C.-C. Chang and C.-J. Lin, *LIBSVM : a library for support vector machines*, (2013), 1–39.

[21] M. Chi and L. Bruzzone, *Semisupervised classification of hyperspectral images by SVMs optimized in the primal*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 6, 1870–1880.

[22] D. A. Clausi, *Comparison and fusion of co-occurrence, Gabor, and MRF texture features for classification of SAR sea ice imagery*, Atmosphere-Ocean **39** (2001), no. 3, 37–41.

[23] ———, *An analysis of co-occurrence texture statistics as a function of grey level quantization*, Canadian Journal of Remote Sensing **28** (2002), no. 1, 45–62.

[24] D. A. Clausi and B. Yue, *Comparing cooccurrence probabilities and Markov random fields for texture analysis of SAR sea ice imagery*, IEEE Transactions on Geoscience and Remote Sensing **42** (2004), no. 1, 215–228.

[25] C. Cortes and V. Vapnik, *Support-vector networks*, Machine learning **20** (1995), no. 3, 273–297.

[26] J. C. Curlander and R. N. McDonough, *Synthetic aperture radar*, John Wiley & Sons, 1991.

[27] B. Demir and S. Erturk, *Hyperspectral image classification using relevance vector machines*, IEEE Geoscience and Remote Sensing Letters **4** (2007), no. 4, 586–590.

[28] H. Deng and D. A. Clausi, *Unsupervised image segmentation using a simple MRF model with a new implementation scheme*, Pattern Recognition **37** (2004), no. 12, 2323–2335.

[29] H. Deng and D. A. Clausi, *Unsupervised segmentation of synthetic aperture radar sea ice imagery using a novel Markov random field model*, IEEE Transactions on Geoscience and Remote Sensing **43** (2005), no. 3, 528–538.

[30] H. Derin and H. Elliott, *Modeling and segmentation of noisy and textured images using Gibbs random fields*, IEEE Transactions on Pattern Analysis and Machine Intelligence (1987), no. 1, 39–55.

[31] P. Domingos, *A unified bias-variance decomposition*, Proceedings of International Conference on Machine Learning, 2000, pp. 231–238.

[32] Y. Dong, B. C. Forster, and a. K. Milne, *Comparison of radar image segmentation by Gaussian- and Gamma-Markov random field models*, International Journal of Remote Sensing **24** (2003), no. 4, 711–722.

[33] I. Dópido, J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias, and J. A. Benediktsson, *Semi-supervised self-learning for hyperspectral image classification*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 7, 4032–4044.

[34] Q. Du and J. E. Fowler, *Hyperspectral image compression using JPEG2000 and principal component analysis*, IEEE Geoscience and Remote Sensing Letters **4** (2007), no. 2, 201–205.

[35] B. Efron and B. Efron, *The jackknife, the bootstrap and other resampling plans*, vol. 38, SIAM, 1982.

[36] K. Ersahin, I. G. Cumming, and R. K. Ward, *Segmentation and classification of polarimetric SAR data using spectral graph partitioning*, IEEE Transactions on Geoscience and Remote Sensing (2010), 1–11.

[37] D. Fequest, *Manice: manual of standard procedures for observing and reporting ice conditions*, Environment Canada, 2005.

[38] Y. Freund and R. E. Schapire, *A desicion-theoretic generalization of on-line learning and an application to boosting*, Computational Learning Theory, Springer, 1995, pp. 23–37.

[39] ———, *Experiments with a new boosting algorithm*, Proceedings of International Conference on Machine Learning, vol. 96, 1996, pp. 148–156.

[40] J. Friedman, T. Hastie, and R. Tibshirani, *Additive logistic regression: a statistical view of boosting*, The Annals of Statistics **28** (2000), no. 2, 337–407.

[41] S. Geman, E. Bienenstock, and R. Doursat, *Neural networks and the bias/variance dilemma*, Neural Computation **4** (1992), no. 1, 1–58.

[42] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Transactions on Pattern Analysis and and Machine Intelligence (1984), no. 6, 721–741.

[43] P. Geurts, D. Ernst, and L. Wehenkel, *Extremely randomized trees*, Machine Learning **63** (2006), no. 1, 3–42.

[44] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, *Random forests for land cover classification*, Pattern Recognition Letters **27** (2006), no. 4, 294–300.

[45] J. Glaister, A. Wong, and D. A. Clausi, *Despeckling of synthetic aperture radar images using monte carlo texture likelihood sampling*, IEEE Transactions on Geoscience and Remote Sensing **52** (2014), no. 2, 1238–1248.

[46] M. Govender, K. Chetty, and H. Bulcock, *A review of hyperspectral remote sensing and its application in vegetation and water resource studies*, Water Sa **33** (2007), no. 2.

[47] A. A. Green, M. Berman, P. Switzer, and M. D. Craig, *A transformation for ordering multispectral data in terms of image quality with implications for noise removal*, IEEE Transactions on Geoscience and Remote Sensing **26** (1988), no. 1, 65–74.

[48] D. Greig, B. Porteous, and A. H. Seheult, *Exact maximum a posteriori estimation for binary images*, Journal of the Royal Statistical Society. Series B (Methodological) (1989), 271–279.

[49] T. Grimwood, *UCS satellite database*, Union of Concerned Scientists **31** (2011).

[50] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, *Investigation of the random forest framework for classification of hyperspectral data*, IEEE Transactions on Geoscience and Remote Sensing **43** (2005), no. 3, 492–501.

[51] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, *Textural features for image classification*, IEEE Transactions on Systems, Man and Cybernetics (1973), no. 6, 610–621.

[52] K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, *Hybrid image segmentation using watersheds and fast region merging*, IEEE Transactions on Image Processing **7** (1998), no. 12, 1684–1699.

[53] T. Hastie and R. Tibshirani, *Discriminant adaptive nearest neighbor classification*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **18** (1996), no. 6, 607–616.

[54] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani, *The elements of statistical learning*, vol. 2, Springer, 2009.

[55] D. Haverkamp, L. K. Soh, and C. Tsatsoulis, *A comprehensive, automated approach to determining sea ice thickness from SAR data*, IEEE Transactions on Geoscience and Remote Sensing **33** (1995), no. I, 46–57.

[56] X. He, D. Cai, S. Yan, and H.-J. Zhang, *Neighborhood preserving embedding*, Proceedings of International Conference on Computer Vision, vol. 2, pp. 1208–1213,.

[57] X. He and P. Niyogi, *Locality preserving projections*, Proceedings of Advances in Neural Information Processing Systems, 2004, pp. 153–160.

[58] X. He, R. S. Zemel, and A. Miguel, *Multiscale conditional random fields for image labeling*, Computer Vision and Pattern Recognition (Washington, DC, USA), 2004.

[59] X. He, R. S. Zemel, and M. Carreira-Perpindn, *Multiscale conditional random fields for image labeling*, CVPR (Washington, DC, USA.), vol. 2, 2004, pp. II–695.

[60] T. K. Ho, *The random subspace method for constructing decision forests*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **20** (1998), no. 8, 832–844.

[61] S. Hojjatoleslami and J. Kittler, *Region growing: a new approach*, IEEE Transactions on Image processing **7** (1998), no. 7, 1079–1084.

[62] Q. A. Holmes, D. R. Nuesch, and R. A. Shuchman, *Textural analysis and real-time classification sea-ice types using digital SAR data*, IEEE Transactions on Geoscience and Remote Sensing (1984), no. 2.

[63] X. Huang and L. Zhang, *An SVM ensemble approach combining spectral, structural, and semantic features for the classification of high-resolution remotely sensed imagery*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 1, 257–272.

[64] G. Hughes, *On the mean accuracy of statistical pattern recognizers*, IEEE Transactions on Information Theory **14** (1968), no. 1, 55–63.

[65] M. Imani and H. Ghassemian, *Band clustering-based feature extraction for classification of hyperspectral images using limited training samples*, IEEE Geoscience and Remote Sensing Letters **11** (2014), no. 8, 1325–1329.

[66] ———, *Feature extraction using attraction points for classification of hyperspectral images in a small sample size situation*, IEEE Geoscience and Remote Sensing Letters **11** (2014), no. 11, 1986–1990.

[67] Q. Jackson and D. A. Landgrebe, *An adaptive method for combined covariance estimation and classification*, IEEE Transactions on Geoscience and Remote Sensing **40** (2002), no. 5, 1082–1087.

[68] S. Jia, Z. Zhu, L. Shen, and Q. Li, *A two-stage feature selection framework for hyperspectral image classification using few labeled samples*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7** (2014), no. 4, 1023–1035.

[69] L. O. Jimenez and D. A. Landgrebe, *Hyperspectral data analysis and supervised feature reduction via projection pursuit*, IEEE Transactions on Geoscience and Remote Sensing **37** (1999), no. 6, 2653–2667.

[70] T. Joachims, *Transductive inference for text classification using support vector machines*, Proceedings of International Conference on Machine Learning (Bled, Slovenia), vol. 99, 1999, pp. 200–209.

[71] U. Kandaswamy, D. A. Adjeroh, and M. C. Lee, *Efficient texture analysis of SAR imagery*, IEEE Transactions on Geoscience and Remote Sensing **43** (2005), no. 9, 2075–2083.

[72] J. Karvonen, B. Cheng, T. Vihma, M. Arkett, and T. Carrieres, *A method for sea ice thickness and concentration analysis based on SAR data and a thermodynamic model*, The Cryosphere **6** (2012), no. 6, 1507–1526.

[73] J. A. Karvonen, *Baltic sea ice sar segmentation and classification using modified pulse-coupled neural networks*, IEEE Transactions on Geoscience and Remote Sensing **42** (2004), no. 7, 1566–1574.

[74] S. Kawaguchi and R. Nishii, *Hyperspectral image classification by bootstrap AdaBoost with random decision stumps*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 11, 3845–3851.

[75] T.-K. Kim and J. Kittler, *Locally linear discriminant analysis for multimodally distributed classes for face recognition with a single model image*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **27** (2005), no. 3, 318–327.

[76] S. Kumar, J. Ghosh, and M. M. Crawford, *Best-bases feature extraction algorithms for classification of hyperspectral data*, IEEE Transactions on Geoscience and Remote Sensing **39** (2001), no. 7, 1368–1379.

[77] L. I. Kuncheva and J. J. Rodríguez, *An experimental study on rotation forest ensembles*, Multiple Classifier Systems, Springer, 2007, pp. 459–468.

[78] B.-C. Kuo and K.-Y. Chang, *Feature extractions for small sample size classification problem*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 3, 756–764.

[79] B.-C. Kuo and D. A. Landgrebe, *Nonparametric weighted feature extraction for classification*, IEEE Transactions on Geoscience and Remote Sensing **42** (2004), no. 5, 1096–1105.

[80] R. Kwok, E. Rignot, B. Holt, and R. Onstott, *Identification of sea ice types in spaceborne synthetic aperture radar data*, Journal of Geophysical Research: Oceans (1978–2012) **97** (1992), no. C2, 2391–2402.

[81] J. Lafferty, A. McCallum, and F. C. Pereira, *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*, Proceedings of International Conference on Machine Learning, 2001, pp. 282–289.

[82] J.-S. Lee, *Digital image enhancement and noise filtering by use of local statistics*, Pattern Analysis and Machine Intelligence (1980), no. 2, 1978–1981.

[83] J.-S. Lee and E. Pottier, *Polarimetric radar imaging: from basics to applications*, CRC press, 2009.

[84] S. Leigh, Z. Wang, and D. A. Clausi, *Automated ice-water classification using dual polarization SAR satellite imagery*, IEEE Transactions on Geoscience and Remote Sensing **52** (2014), no. 9, 5529–5539.

[85] F. Li, D. A. Clausi, L. Wang, and L. Xu, *A semi-supervised approach for ice-water classification using dual-polarization SAR satellite imagery*, Proceedings of International Conference on Computer Vision and Pattern Recognition Workshops (Boston, USA), IEEE, 2015.

[86] F. Li, L. Xu, P. Siva, A. Wong, and D. A. Clausi, *Hyperspectral image classification with limited labeled training samples using enhanced ensemble learning and conditional random fields*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7** (2014), no. 6, 2035–2043.

[87] J. Li, *Wavelet-based feature extraction for improved endmember abundance estimation in linear unmixing of hyperspectral signals*, Geoscience and Remote Sensing, IEEE Transactions on **42** (2004), no. 3, 644–649.

[88] J. Li, *Discriminative image segmentation: Applications to hyperspectral data*, Ph.D. thesis, Instituto Superior Técnico, 2011.

[89] 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 **48** (2010), no. 11, 4085–4098.

[90] ———, *Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 3, 809–823.

[91] N. Li, H. Huo, and T. Fang, *A novel texture-preceded segmentation algorithm for high-resolution imagery*, IEEE Transactions on Geoscience and Remote Sensing **48** (2010), no. 7, 2818–2828.

[92] S. Z. Li and S. Singh, *Markov random field modeling in image analysis*, vol. 26, Springer, 2009.

[93] W. Li, S. Prasad, and J. E. Fowler, *Hyperspectral image classification using gaussian mixture models and Markov random fields*, IEEE Geoscience and Remote Sensing Letters **11** (2014), no. 1, 153–157.

[94] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, *Locality-preserving dimensionality reduction and classification for hyperspectral image analysis*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 4, 1185–1198.

[95] W. Liao, A. Pizurica, P. Scheunders, W. Philips, and Y. Pi, *Semisupervised local discriminant analysis for feature extraction in hyperspectral images*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 1, 184–198.

[96] D. Lunga, S. Prasad, M. M. Crawford, and O. Ersoy, *Manifold-learning-based feature extraction for classification of hyperspectral data: A review of advances in manifold learning*, IEEE Signal Processing Magazine **31** (2014), no. 1, 55–66.

[97] P. Mather and M. Koch, *Computer processing of remotely-sensed images: an introduction*, John Wiley & Sons, 2011.

[98] P. Mather and B. Tso, *Classification methods for remotely sensed data*, CRC press, 2009.

[99] R. Meir and G. Rätsch, *An introduction to boosting and leveraging*, Advanced Lectures on Machine Learning, Springer, 2003, pp. 118–183.

[100] S. Melacci and M. Belkin, *Laplacian support vector machines trained in the primal*, The Journal of Machine Learning Research **12** (2011), 1149–1184.

[101] A. Merentitis, C. Debes, and R. Heremans, *Ensemble learning in hyperspectral image classification: Toward selecting a favorable bias-variance tradeoff*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7** (2014), no. 4, 1089–1102.

[102] M.-A. Moen, A. P. Doulgeris, S. N. Anfinsen, A. H. Renner, N. Hughes, S. Gerland, and T. Eltoft, *Comparison of feature based segmentation of full polarimetric SAR satellite sea ice images with manually drawn ice charts*, The Cryosphere **7** (2013), no. 6, 1693–1705.

[103] K. P. Murphy, Y. Weiss, and M. I. Jordan, *Loopy belief propagation for approximate inference: An empirical study*, Proceedings of Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.

[104] Q. D. Nam Hoai Ly and J. Fowler, *Laplacian eigenmaps-based polarimetric dimensionality reduction for SAR image classification*, IEEE Transactions on Geoscience and Remote Sensing **52** (2014), no. 7, 3872–3884.

[105] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, *Text classification from labeled and unlabeled documents using em*, Machine learning **39** (2000), no. 2-3, 103–134.

[106] R. Nock and F. Nielsen, *Statistical region merging*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **26** (2004), no. 11, 1452–1458.

[107] E. E. Osuna, R. Freund, and F. Girosi, *Support Vector Machines : Training and Applications*, Tech. Report 1602, Massachusetts Institute of Technology, 1999.

[108] N. Otsu, *A threshold selection method from gray-level histograms*, Automatica **11** (1975), no. 285-296, 23–27.

[109] M. Pal, *Random forest classifier for remote sensing classification*, International Journal of Remote Sensing **26** (2005), no. 1, 217–222.

[110] S. Parrilli, M. Poderico, C. V. Angelino, and L. Verdoliva, *A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 2, 606–616.

[111] K. C. Partington, J. D. Flach, D. Barber, D. Isleifson, P. J. Meadows, and P. Verlaan, *Dual-polarization C-band radar observations of sea ice in the Amundsen Gulf*, IEEE Transactions on Geoscience and Remote Sensing **48** (2010), no. 6, 2685–2691.

[112] B. Peng and D. Zhang, *Automatic image segmentation by dynamic region merging*, IEEE Transactions on Image Processing **20** (2011), no. 12, 3592–3605.

[113] M. Pesaresi and J. Benediktsson, *A new approach for the morphological segmentation of high-resolution satellite imagery*, IEEE Transactions on Geoscience and Remote Sensing **39** (2001), no. 2, 309–320.

[114] J. C. Platt, *Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods*, Advances in Large Margin Classifiers (1999).

[115] A. Plaza, J. A. Benediktsson, J. W. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri, et al., *Recent advances in techniques for hyperspectral image processing*, Remote Sensing of Environment **113** (2009), S110–S122.

[116] R. B. Potts, *Some generalized order-disorder transformations*, Mathematical Proceedings of the Cambridge Philosophical Society, vol. 48, Cambridge Univ Press, 1952, pp. 106–109.

[117] A. Qin and D. A. Clausi, *Multivariate image segmentation using semantic region growing with adaptive edge penalty*, IEEE Transactions on Image Processing **19** (2010), no. 8, 2157–2170.

[118] E. Rignot and R. Chellappa, *Segmentation of polarimetric synthetic aperture radar data.*, IEEE Transactions on Image Processing **1** (1992), no. 3, 281–300.

[119] J. J. Rodríguez, C. J. Alonso, and O. J. Prieto, *Bias and variance of rotation-based ensembles*, Computational Intelligence and Bioinspired Systems, Springer, 2005, pp. 779–786.

[120] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso, *Rotation forest: A new classifier ensemble method*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **28** (2006), no. 10, 1619–1630.

[121] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science **290** (2000), no. 5500, 2323–2326.

[122] Y. Saeys, T. Abeel, and Y. V. D. Peer, *Robust feature selection using ensemble feature selection techniques*, Machine Learning and Knowledge Discovery in Databases, Springer Berlin Heidelberg, 2008, pp. 313–325.

[123] Q. Sami ul Haq, L. Tao, F. Sun, and S. Yang, *A fast and robust sparse approach for hyperspectral data classification using a few labeled samples*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 6, 2287–2302.

[124] R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee, *Boosting the margin: A new explanation for the effectiveness of voting methods*, Annals of Statistics (1998), 1651–1686.

[125] R. E. Schapire and Y. Singer, *Improved boosting algorithms using confidence-rated predictions*, Machine Learning **37** (1999), no. 3, 297–336.

[126] K. Schindler, *An overview and comparison of smooth labeling methods for land-cover classification*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 11, 4534–4545.

[127] B. Schölkopf, A. Smola, and K.-R. Müller, *Kernel principal component analysis*, Proceedings of International Conference on Artificial Neural Networks, Springer, 1997, pp. 583–588.

[128] G. Schwarz, *Estimating the dimension of a model*, The annals of statistics **6** (1978), no. 2, 461–464.

[129] S. B. Serpico and G. Moser, *Weight parameter optimization by the Ho-Kashyap algorithm in MRF models for supervised image classification*, IEEE Transactions on Geoscience and Remote Sensing **44** (2006), no. 12, 3695–3705.

[130] B. M. Shahshahani and D. A. Landgrebe, *The effect of unlabeled samples in reducing the small sample size problem and mitigating the hughes phenomenon*, IEEE Transactions on Geoscience and Remote Sensing **32** (1994), no. 5, 1087–1095.

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

[132] J. Shotton, J. Winn, C. Rother, and A. Criminisi, *Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation*, ECCV, Springer, 2006, pp. 1–15.

[133] P. Siva and A. Wong, *URC: Unsupervised regional clustering of remote sensing imagery*, Proceedings of International Geoscience and Remote Sensing Symposium, 2014.

[134] P. C. Smits and S. G. Dellepiane, *Synthetic aperture radar image segmentation by a detail preserving Markov random field approach*, IEEE Transactions on Geoscience and Remote Sensing **35** (1997), no. 4, 844–857.

[135] L.-K. Soh and C. Tsatsoulis, *Texture analysis of SAR sea ice imagery*, IEEE Transactions on Geoscience and Remote Sensing **37** (1999), no. 2, 780–795.

[136] L.-K. Soh, C. Tsatsoulis, D. Gineris, and C. Bertoia, *ARKTOS: An intelligent system for SAR sea ice image classification*, IEEE Transactions on Geoscience and Remote Sensing **42** (2004), no. 1, 229–248.

[137] X. Su, C. He, Q. Feng, X. Deng, and H. Sun, *A supervised classification method based on conditional random fields with multiscale region connection calculus model for SAR image*, IEEE Geoscience Remote and Sensing Letters **8** (2011), no. 3, 497–501.

[138] M. Sugiyama, *Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis*, Journal of Machine Learning Research **8** (2007), 1027–1061.

[139] M. Sugiyama, T. Idé, S. Nakajima, and J. Sese, *Semi-supervised local fisher discriminant analysis for dimensionality reduction*, Machine learning **78** (2010), no. 1-2, 35–61.

[140] Y. Sun, A. Carlström, and J. Askne, *SAR image classification of ice in the Gulf of Bothnia*, International Journal of Remote Sensing **13** (1992), no. 13, 2489–2514.

[141] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, *A comparative study of energy minimization methods for markov random fields with smoothness-based priors*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **30** (2008), no. 6, 1068–1080.

[142] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. A. Benediktsson, *SVM-and MRF-based method for accurate classification of hyperspectral images*, IEEE Geoscience and Remote Sensing Letters **7** (2010), no. 4, 736–740.

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

[144] J. C. Tilton, Y. Tarabalka, P. M. Montesano, and E. Gofman, *Best merge region-growing segmentation with integrated nonadjacent region object aggregation*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 11, 4454–4467.

[145] M. E. Tipping.

[146] C. Tison, J.-M. Nicolas, F. Tupin, and H. Maître, *A new statistical model for Markovian classification of urban areas in high-resolution SAR images*, IEEE Transactions on Geoscience and Remote Sensing **42** (2004), no. 10, 2046–2057.

[147] D. Tuia, F. Pacifici, M. Kanevski, and W. J. Emery, *Classification of very high spatial resolution imagery using mathematical morphology and support vector machines*, IEEE Transactions on Geoscience and Remote Sensing **47** (2009), no. 11, 3866–3879.

[148] L. Van der Maaten, E. Postma, and H. Van Den Herik, *Dimensionality reduction: A comparative review*, Journal of Machine Learning Research **10** (2009), 1–41.

[149] L. Vincent and P. Soille, *Watersheds in digital spaces: an efficient algorithm based on immersion simulations*, IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991), no. 6, 583–598.

[150] J. Wang and C.-I. Chang, *Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis*, IEEE Transactions on Geoscience and Remote Sensing **44** (2006), no. 6, 1586–1600.

[151] J. Wang, A. Kalousis, and A. Woznica, *Parametric local metric learning for nearest neighbor classification*, Proceedings of Advances in Neural Information Processing Systems, 2012, pp. 1601–1609.

[152] L. Wang, S. Hao, Q. Wang, and Y. Wang, *Semi-supervised classification for hyperspectral imagery based on spatial-spectral label propagation*, ISPRS Journal of Photogrammetry and Remote Sensing **97** (2014), 123–137.

[153] B. Waske, S. van der Linden, J. A. Benediktsson, A. Rabe, and P. Hostert, *Sensitivity of support vector machines to random feature selection in classification of hyperspectral data*, IEEE Transactions on Geoscience and Remote Sensing **48** (2010), no. 7, 2880–2889.

[154] K. Q. Weinberger and L. K. Saul, *Distance metric learning for large margin nearest neighbor classification*, Journal of Machine Learning Research **10** (2009), 207–244.

[155] J. Weston and C. Watkins, *Support vector machines for multi-class pattern recognition*, Proceedings of the seventh European symposium on artificial neural networks (Bruges, Belgium), 1999.

[156] Wikipedia, *Radarsat-2 — wikipedia, the free encyclopedia*, 2015, [Online; accessed 13-May-2015].

[157] T.-F. Wu, C.-J. Lin, and R. C. Weng, *Probability estimates for multi-class classification by pairwise coupling*, Journal of Machine Learning Research **5** (2004), 975–1005.

[158] ———, *Probability estimates for multi-class classification by pairwise coupling*, The Journal of Machine Learning Research **5** (2004), 975–1005.

[159] Y. Wu, M. Li, P. Zhang, H. Zong, P. Xiao, and C. Liu, *Unsupervised multi-class segmentation of SAR images using triplet Markov fields models based on edge penalty*, Pattern Recognition Letters **32** (2011), no. 11, 1532–1540.

[160] J. Xia, J. Chanussot, P. Du, and X. He, *Spectral–spatial classification for hyperspectral data using rotation forests with local feature extraction and Markov random fields*, IEEE Transactions on Geoscience and Remote Sensing **53** (2015), no. 5, 2532–2546.

[161] J. Xia, S. Member, P. Du, S. Member, X. He, and J. Chanussot, *Hyperspectral remote sensing image classification based on rotation forest*, IEEE Geoscience and Remote Sensing Letters (2013), 1–5.

[162] L. Xu, J. Li, and A. Brenning, *A comparative study of different classification techniques for marine oil spill identification using RADARSAT-1 imagery*, Remote Sensing of Environment **141** (2014), 14–23.

[163] J.-M. Yang, B.-C. Kuo, P.-T. Yu, and C.-H. Chuang, *A dynamic subspace method for hyperspectral image classification*, IEEE Transactions on Geoscience and Remote Sensing **48** (2010), no. 7, 2840–2853.

[164] P. Yang, Y. Hwa Yang, B. B Zhou, and A. Y Zomaya, *A review of ensemble methods in bioinformatics*, Current Bioinformatics **5** (2010), no. 4, 296–308.

[165] J. P. Yicong Zhou and C. L. P. Chen, *Dimension reduction using spatial and spectral regularized local discriminant embedding for hyperspectral image classification*, IEEE Transactions on Geoscience and Remote Sensing **53** (2015), no. 2, 1082–1095.

[166] H. Yu, X. Zhang, S. Wang, and B. Hou, *Context-based hierarchical unequal merging for SAR image segmentation*, IEEE Transactions on Geoscience and Remote Sensing **51** (2013), no. 2, 995–1009.

[167] P. Yu, A. Qin, and D. A. Clausi, *Unsupervised polarimetric SAR image segmentation and classification using region growing with edge penalty*, IEEE Transactions on Geoscience and Remote Sensing **50** (2012), no. 4, 1302–1317.

[168] Q. Yu, *Automated SAR sea ice interpretation*, Ph.D. thesis, University of Waterloo, 2006.

[169] Q. Yu and D. A. Clausi, *SAR sea-ice image analysis based on iterative region growing using semantics.*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 12, 3919.

[170] _____, *IRGS: Image segmentation using edge penalties and region growing*, IEEE Transactions on Pattern Analysis and and Machine Intelligence **30** (2008), no. 12, 2126–2139.

[171] Y. Yu and S. T. Acton, *Speckle reducing anisotropic diffusion.*, IEEE Transactions on Image Processing **11** (2002), no. 11, 1260–1270.

[172] C.-X. Zhang and J.-S. Zhang, *RotBoost: A technique for combining rotation forest and AdaBoost*, Pattern Recognition Letters **29** (2008), no. 10, 1524–1536.

[173] G. Zhang and X. Jia, *Simplified conditional random fields with class boundary constraint for spectral-spatial based remote sensing image classification*, IEEE Geoscience and Remote Sensing Letters **9** (2012), no. 5, 856–860.

[174] P. Zhang, M. Li, Y. Wu, M. Liu, F. Wang, and L. Gan, *SAR image multiclass segmentation using a multiscale TMF Model in wavelet d omain*, IEEE Geoscience and Remote Sensing Letters **9** (2012), no. 6, 1099–1103.

[175] T. Zhang, J. Yang, D. Zhao, and X. Ge, *Linear local tangent space alignment and application to face recognition*, Neurocomputing **70** (2007), no. 7, 1547–1553.

[176] P. Zhong and R. Wang, *A multiple conditional random fields ensemble model for urban area detection in remote sensing optical images*, IEEE Transactions on Geoscience and Remote Sensing **45** (2007), no. 12, 3978–3988.

[177] _____, *Learning conditional random fields for classification of hyperspectral images*, IEEE Transactions on Image Processing **19** (2010), no. 7, 1890–1907.

[178] Y. Zhong, X. Lin, and L. Zhang, *A support vector conditional random fields classifier with a Mahalanobis distance boundary constraint for high spatial resolution remote sensing imagery*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7** (2014), no. 4, 1314–1330.

[179] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*, CRC Press, 2012.

[180] J. Zhu, H. Zou, S. Rosset, and T. Hastie, *Multi-class Adaboost*, Statistics and Its (2009).

[181] X. Zhu, *Semi-supervised learning literature survey*, Computer Science, University of Wisconsin-Madison **2** (2006), 3.