

JUNE 2024

Ph.D. in Electrical and Electronics Engineering

TUĞCAN DÜNDAR

REPUBLIC OF TÜRKİYE
GAZİANTEP UNIVERSITY
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

DEVELOPMENT OF MACHINE LEARNING ALGORITHMS
FOR ENHANCEMENT AND CLASSIFICATION OF
REMOTELY SENSED HYPERSPECTRAL IMAGES

Ph.D. THESIS
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

BY
TUĞCAN DÜNDAR

JUNE 2024

**DEVELOPMENT OF MACHINE LEARNING ALGORITHMS
FOR ENHANCEMENT AND CLASSIFICATION OF
REMOTELY SENSED HYPERSPECTRAL IMAGES**

Ph.D. Thesis

in

Electrical and Electronics Engineering

Gaziantep University

Supervisor

Assoc. Prof. Dr. Taner İNCE

by

Tuğcan DÜNDAR

June 2024



©2024[Gaziantep University]

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Tuğcan DÜNDAR

ABSTRACT

DEVELOPMENT OF MACHINE LEARNING ALGORITHMS FOR ENHANCEMENT AND CLASSIFICATION OF REMOTELY SENSED HYPERSPECTRAL IMAGES

DÜNDAR, Tuğcan

Ph.D. in Electrical and Electronics Engineering

Supervisor: Assoc. Prof. Dr. Taner İNCE

June 2024

91 pages

Hyperspectral imaging is a process of measuring the reflected radiance of a region over a wide-range of many contiguous spectrum. The distinct land cover materials absorb and reflect the light differently at a certain wavelength. Due to the plenty of spectral bands and the different reflection characteristics of the objects, each substance has its own spectral signature. Hyperspectral image classification is a research field that attempts to classify the objects using these signatures by the efficient methods. These methods are divided into two categories that is the spectral methods which use only the spectral information and the spatial-spectral methods which also use the local information in the hyperspectral image to enhance the classification results. Within the scope of this thesis, three new spatial-spectral hyperspectral image classification methods have been proposed. In the first method, weighted and label-consistent joint-sparse classifier has been proposed. In the second study, locality-constraint joint-sparse and weighted low-rank based method has been suggested. As for the last method, guided filter of random patches network and relaxed collaborative representation based approach has been proposed. The experiments have been performed on the real datasets. The results have confirmed that the proposed methods outperform the recent methods in terms of classification accuracy metrics.

Key Words: Hyperspectral image classification, Sparse representation, Low-rank, Relaxed collaborative representation, Random patches network.

ÖZET

UZAKTAN ALGILANAN HİPERSPEKTRAL GÖRÜNTÜLERİN İYİLEŞTİRİLMESİ VE SINIFLANDIRILMASI İÇİN MAKİNE ÖĞRENMESİ ALGORİTMALARININ GELİŞTİRİLMESİ

DÜNDAR, Tuğcan
Doktora Tezi, Elektrik-Elektronik Mühendisliği
Danışman: Doç. Dr. Taner İNCE
Haziran 2024
91 sayfa

Hiperspektral görüntüleme, geniş bir aralıktaki birçok bitişik spektrum üzerinden bir bölgenin yansıyan parlaklığını ölçme işlemidir. Farklı arazi örtüsü malzemeleri, ışığı belirli bir dalga boyunda farklı şekilde emer ve yansıtırlar. Spektral bantların çokluğu ve nesnelerin farklı yansıma özellikleri nedeniyle her maddenin kendine ait bir spektral imzası vardır. Hiperspektral görüntü sınıflandırma, bu imzaları kullanarak nesneleri verimli yöntemlerle sınıflandırmaya çalışan bir araştırma alanıdır. Bu yöntemler, sadece spektral bilgiyi kullanan spektral yöntemler ve sınıflandırma sonuçlarını geliştirmek için hiperspektral görüntüdeki yerel bilgileri de kullanan uzamsal-spektral yöntemler olmak üzere iki kategoriye ayrılır. Bu tez kapsamında üç yeni uzamsal-spektral hiperspektral görüntü sınıflandırma yöntemi önerilmiştir. İlk yöntemde ağırlıklı ve etiket-tutarlı ortak-seyrek sınıflandırıcı önerilmiştir. İkinci çalışmada ise yerellik-kısıtlı ortak-seyrek ve ağırlıklı düşük-rank tabanlı yöntem önerilmiştir. Son yöntemde ise rastgele yamalar ağının kılavuzlu filtresi ve rahatlatılmış işbirlikçi temsil tabanlı yaklaşım önerilmiştir. Deneyler gerçek veri setleri üzerinde gerçekleştirilmiştir. Sonuçlar, önerilen yöntemlerin sınıflandırma doğruluğu ölçütleri açısından son yöntemlerden daha iyi performans gösterdiğini doğrulamıştır.

Anahtar Kelimeler: Hiperspektral görüntü sınıflandırma, Seyrek temsil, Düşük-rank, Rahatlatılmış işbirlikçi temsil, Rastgele yamalar ağı.



"Dedicated to my family"

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Assoc. Prof. Dr. Taner İNCE for his guidance and support throughout the study. I am thankful for his encouragement and motivation.

I would like to express my love and gratitude to my family for their support, always best wishes. My heartfelt appreciation also goes to my nephews Aysima DEMİRCİ and Ela DEMİRCİ, who make me happy even in the most difficult times.

My sincere gratitude also goes to my esteemed Ph.D. thesis monitoring committee members, Asst. Prof. Dr. Hasari KARCI and Asst. Prof. Dr. Seydi KAÇMAZ. I am profoundly grateful for their expert guidance, constructive feedback, and dedicated involvement in shaping this thesis. Their wisdom and insights have been pivotal in refining my work and broadening my scholarly horizons.

To Dr. Önder POLAT, I extend my gratitude for his enduring friendship and encouragement, which have provided me with a constant source of support.

TABLE OF CONTENTS

	Page
ABSTRACT	i
ÖZET	ii
DEDICATION	iii
ACKNOWLEDGEMENTS	iv
TABLE OF CONTENTS	v
LIST OF TABLES	vii
LIST OF FIGURES	viii
LIST OF ABBREVIATIONS	xi
CHAPTER 1 INTRODUCTION	1
1.1 Hyperspectral Imaging	1
1.2 Hyperspectral Image Classification	2
1.3 Literature Survey	3
1.4 Contributions of the Thesis	10
1.4.1 The Proposed WLC-JSC Method	10
1.4.2 The Proposed LCJS-WLR Method	10
1.4.3 The Proposed GRR Method	11
1.5 Outline of the Thesis	12
CHAPTER 2 DATASETS AND EVALUATION METRICS	13
2.1 Indian Pines Dataset	13
2.2 Pavia University Dataset	14
2.3 WHU-Hi-HongHu Dataset	14
2.4 Evaluation Metrics	17
CHAPTER 3 WEIGHTED AND LABEL-CONSISTENT JOINT-SPARSE	
 CLASSIFIER FOR HYPERSPECTRAL IMAGE	
 CLASSIFICATION	18
3.1 Introduction	18
3.2 Problem Formulation	18

3.3 Experimental Results	24
3.4 Summary	29
CHAPTER 4 LOCALITY CONSTRAINT JOINT-SPARSE AND WEIGHTED LOW-RANK BASED HYPERSPECTRAL IMAGE CLASSIFICATION	30
4.1 Introduction	30
4.2 Problem Formulation	30
4.3 Experimental Results	36
4.4 Summary	40
CHAPTER 5 GUIDED FILTER OF RANDOM PATCHES NETWORK AND RELAXED COLLABORATIVE REPRESENTATION BASED HYPERSPECTRAL IMAGE CLASSIFICATION . .	41
5.1 Introduction	41
5.2 Problem Formulation	42
5.2.1 RPNet Feature Extraction	42
5.2.2 Guided Filter	43
5.2.3 Pseudo-Sample Generation	45
5.2.4 Linear Discriminant Analysis	46
5.2.5 Relaxed Collaborative Representation	47
5.3 Experimental Results	52
5.3.1 Indian Pines 2 Dataset	52
5.3.2 Pavia University Dataset	60
5.3.3 WHU-Hi-HongHu Dataset	67
5.4 Summary	75
CHAPTER 6 CONCLUSION AND FUTURE WORK	76
6.1 Conclusion	76
6.2 Future Work	78
REFERENCES	79
CURRICULUM VITAE	90

LIST OF TABLES

	Page
Table 2.1 Sixteen ground-truth classes of the IP1 and IP2 dataset	13
Table 2.2 Nine ground-truth classes of the PU dataset	15
Table 2.3 Twenty-two ground-truth classes of the WHHH dataset	16
Table 3.1 IP1 dataset: Classification accuracy results (%) of SVM, SRC, JSRC, JSRNN, NLW-JSRC, CCJSR and proposed WLC-JSC	27
Table 4.1 IP1 dataset: Classification results of the methods LCLRR, JSRC, RSJSRC, LRSRC-SCCE and proposed LCJS-WLR	38
Table 5.1 IP2 dataset: Classification accuracy results under 5% labeled sam- ples obtained by RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, JCRC-MTL, SSRPNet, GR-SVM and proposed GRR	57
Table 5.2 PU dataset: Classification accuracy results under 1% labeled sam- ples obtained by RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, JCRC-MTL, SSRPNet, GR-SVM and proposed GRR	65
Table 5.3 WHHH dataset: Classification accuracy results under 0.5% labeled samples obtained by RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, JCRC-MTL, SSRPNet, GR-SVM and proposed GRR	72

LIST OF FIGURES

	Page
Figure 1.1 A visual illustration of hyperspectral imaging	1
Figure 1.2 HSI datacube	2
Figure 2.1 IP1 dataset: (a) False-color composite image and (b) Ground-truth .	14
Figure 2.2 IP2 dataset: (a) False-color composite image and (b) Ground-truth .	14
Figure 2.3 PU dataset: (a) False-color composite image and (b) Ground-truth .	15
Figure 2.4 WHHH dataset: (a) False-color composite image and (b) Ground-truth	16
Figure 3.1 Flowchart of the proposed WLC-JSC method	23
Figure 3.2 IP1 dataset: OA and AA results of the proposed WLC-JSC with the chosen parameter intervals for I , w_{low} and w_{high}	25
Figure 3.3 IP1 dataset: (a) Ground-truth and classification maps obtained by (b) SVM, (c) SRC, (d) JSRC, (e) JSRNN, (f) NLW-JSRC, (g) CCJSR and (h) proposed WLC-JSC	28
Figure 4.1 IP1 dataset: Effect of different γ and τ parameters on OA result of the proposed LCJS-WLR	37
Figure 4.2 IP1 dataset: (a) Ground-truth; classification maps and OA results of the methods (b) LCLRR, (c) JSRC, (d) RSJSRC, (e) LRSRC-SCCE and (f) proposed LCJS-WLR.	39
Figure 5.1 Flowchart of the proposed GRR	51
Figure 5.2 IP2 dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w	53
Figure 5.3 IP2 dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA	54
Figure 5.4 IP2 dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth	54

Figure 5.5	IP2 dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ	55
Figure 5.6	IP2 dataset: (a) False-color image, (b) Ground-truth; classification maps obtained by (c) RpNet, (d) RpNet-RF, (e) RPCC, (f) GRPC, (g) MS-RpNet, (h) JCRC-MTL, (i) SSRpNet, (j) GR-SVM and (k) GRR	58
Figure 5.7	IP2 dataset: OA results of the methods RpNet, RpNet-RF, RPCC, GRPC, MS-RpNet, JCRC-MTL, SSRpNet, GR-SVM and proposed GRR under varying training percentages	59
Figure 5.8	PU dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w	60
Figure 5.9	PU dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA	61
Figure 5.10	PU dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth	62
Figure 5.11	PU dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ	63
Figure 5.12	PU dataset: (a) False-color image, (b) Ground-truth; classification maps with OAs obtained by (c) RpNet, (d) RpNet-RF, (e) RPCC, (f) GRPC, (g) MS-RpNet, (h) JCRC-MTL, (i) SSRpNet, (j) GR-SVM and (k) GRR	66
Figure 5.13	PU dataset: OA results of the methods RpNet, RpNet-RF, RPCC, GRPC, MS-RpNet, JCRC-MTL, SSRpNet, GR-SVM and proposed GRR under varying training samples per class	67
Figure 5.14	WHHH dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w	68
Figure 5.15	WHHH dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA	69
Figure 5.16	WHHH dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth	69

Figure 5.17 WHHH dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ	70
Figure 5.18 WHHH dataset: (a) False-color image, (b) Ground-truth; classification maps with OAs obtained by (c) RpNet, (d) RpNet-RF, (e) RPCC, (f) GRPC, (g) MS-RpNet, (h) JCRC-MTL, (i) SSRpNet, (j) GR-SVM and (k) GRR.	73
Figure 5.19 WHHH dataset: OA results of the methods RpNet, RpNet-RF, RPCC, GRPC, MS-RpNet, JCRC-MTL, SSRpNet, GR-SVM and proposed GRR under varying training samples per class	74



LIST OF ABBREVIATIONS

AA	Average Accuracy
ADMM	Alternating Direction Method of Multipliers
CCJSR	Correlation Coefficient JSR
CRC	Collaborative Representation Classifier
DMP	Differential Morphological Profiles
DWT	Discrete Wavelet Transform
EMAP	Extended Multi-Attribute Profile
ERS	Entropy Rate Superpixels
GF	Guided Filter
GRPC	Gabor Filter with Random Patch Convolution
GRR	Guided Filter of Random Patches Network and Relaxed Collaborative Representation
HSI	Hyperspectral Image
HSIC	Hyperspectral Image Classification
IP	Indian Pines
JSR	Joint Sparse Representation
JSRC	Joint Sparse Representation Classifier
JSRNN	Joint Sparse Representation Nearest Neighbor
LBP	Local Binary Pattern
LCJS-WLR	Locality Constraint Joint-Sparse and Weighted Low-Rank
LCMR	Local Covariance Matrix Representation
LDA	Linear Discriminant Analysis
LRR	Low-Rank Representation
MNF	Maximum Noise Fraction
MSRPNet	Multiscale Superpixelwise Random Patches Network
NLW-JSRC	Non-Local Weighted JSRC

NRS	Nearest-Regularized Subspace
OA	Overall Accuracy
PCA	Principal Component Analysis
PU	Pavia University
RCR	Relaxed Collaborative Representation
ReLU	Rectified Linear Units
RPCC	Random Patches Convolution and Local Covariance
RPNet	Random Patches Network
RPNET-RF	Random Patches Network with the Recursive Filtering
S3-PCA	Spectral-Spatial and Superpixelwise Principal Component Analysis
SGF	Spectral Gradient Feature
SOMP	Simultaneous Orthogonal Matching Pursuit
SRC	Sparse Representation Classifier
SSA	Singular Spectrum Analysis
SSRPNet	Spectral–Spatial Random Patches Network
SVM	Support Vector Machines
SVF	Spectral Value Feature
WHHH	WHU-Hi HongHu
WJCRC	Weighted Joint Collaborative Representation Classifier
WLC-JSC	Weighted and Label-Consistent Joint-Sparse Classifier

CHAPTER 1

INTRODUCTION

1.1 Hyperspectral Imaging

A hyperspectral imaging sensor, which is mounted on an aircraft or spacecraft, is capable of collecting radiance data of a region on the Earth surface [1]. The radiance information is usually measured from the visible through the near-infrared regions (0.4-2.5 μm) of the electromagnetic spectrum [2]. The hyperspectral imaging sensor has the ability to collect hundreds of spectral band information within this wide, continuous and contiguous wavelength range with high spatial-spectral resolution [3]. Since the land-cover materials reflect the emitted light differently at a specific wavelength, the vast number of collected spectral bands enables the formation of distinct spectral signatures of the different objects [4]. Such spectral signatures are used to classify or detect the objects in the fields of military application, mineral exploration and cultivation [5]. A visual representation of hyperspectral imaging is given in Figure 1.1.

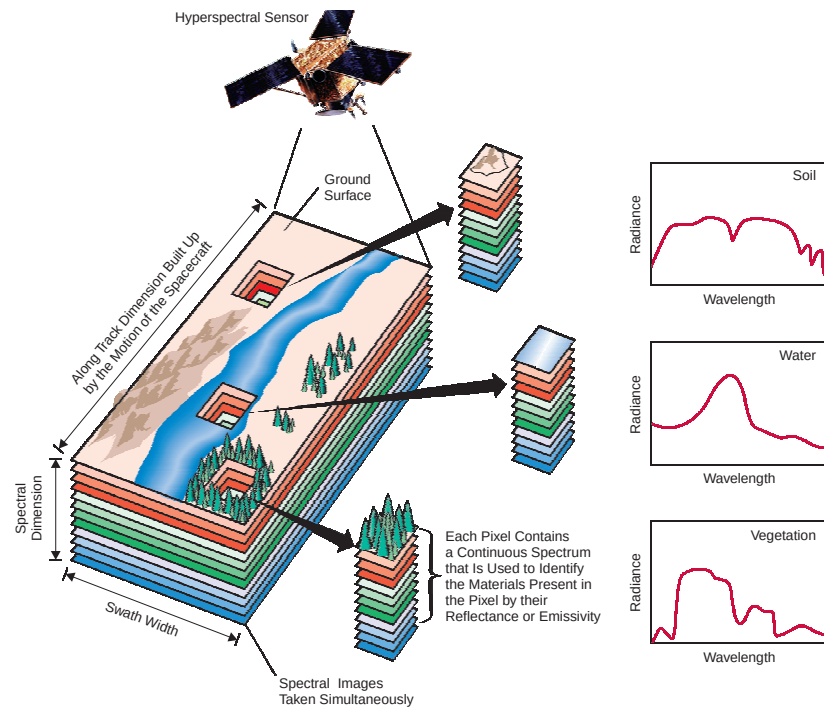


Figure 1.1 A visual illustration of hyperspectral imaging [6].

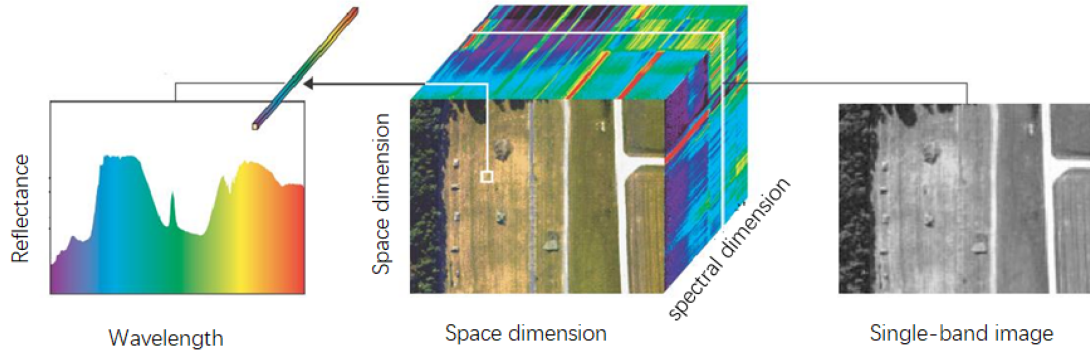


Figure 1.2 HSI datacube [7].

A hyperspectral image (HSI) can be thought as a datacube whose front face and depth indicates the spatial and spectral perspectives, respectively. In other words, it is a three-dimensional (3D) data consisting of concatenated two-dimensional (2D) images of a region measured at different wavelengths. An example of a HSI is given in the Figure 1.2.

1.2 Hyperspectral Image Classification

Hyperspectral image classification (HSIC) is an active research topic among the remote sensing community [8]. Thanks to the increasing spatial and spectral resolution of the sensors with the development of remote sensing systems, it is possible to classify and detect objects without contact. The purpose of the HSIC is to assign a unique class label to the pixels in the image. Each pixel is a vector with a length equal to the number of spectral channels that the sensor measures.

The HSIC methods are basically divided into two parts such as supervised and unsupervised [9]. If the training samples are provided by the experts, such methods are referred as supervised classification. Otherwise, the methods where there are no train samples perform unsupervised classification. Similarly, the classifiers are categorized into two parts such as spectral and spatial-spectral classifiers [10]. While the spectral classifiers use only the spectral information, the spatial-spectral classifiers utilize the local geometrical structures within the HSI, that is, spatial structure, in addition to the spectral information. When the HSIC researches are analyzed, the supervised spatial-spectral HSIC methods have been studied broadly by the researchers [11–14]. This is due to the fact that the train samples are mostly provided by the experts and the use of

spatial information within the image increases the classification accuracy results [15].

Despite the advantages of hyperspectral imaging such as the high spatial and spectral resolution for the HSIC, it also has some disadvantages. These are the high dimensionality of the HSI, the small number of available training samples, the imbalance of the number of samples between classes, the high variance of the samples in the same class and the similarity of the spectral signatures of samples from different classes in the borders [16], [17]. Therefore, researchers have contributed to the literature by proposing different methods for years to overcome these difficulties and increase the classification accuracy results. The following section provides a detailed information about these studies.

1.3 Literature Survey

Early proposed HSIC algorithms utilizes only the spectral information such as support vector machines (SVM) [18] which solves a series of optimization problems to find the optimal hyperplanes that separates the samples from different classes as much as possible. It has attracted great attention among the researchers, therefore, many methods have been proposed to increase the capability of the SVM. The proposed first spectral SVM-based HSIC method [19] ignores the curse of high dimensionality under limited number of train samples. Since such problems decrease the performance of the SVM, many SVM-driven methods have been developed to overcome these drawbacks. These methods can be categorized as composite kernel-based SVMs [20–23] to incorporate the spatial structure of the HSI into SVM, semi-supervised SVMs [24–27] to utilize the unlabeled samples in addition to the few labeled samples for the training stage, and, active SVMs [28–31] to generate and increase the training dictionary iteratively. Although all the proposed SVM-based methods provide satisfactory results, there are some limitations [32]. For example, the selection of optimal kernel and its parameters for the composite kernel-based SVMs is a challenging task. The lack of available training samples limits the performance of both the semi-supervised and active SVMs. In addition to these disadvantages, SVM-driven methods are not capable of handling outliers that can be existed in the training dictionary.

In order to handle with the drawbacks of the SVM-based methods, representation-

based HSIC algorithms such as sparse representation classifier (SRC) [33–35] was proposed. The method SRC relies on the observation that a test sample can be linearly represented by only a set of train samples of the same class in a given training dictionary. It means that the coefficient vector which is used to predict the class label of the given test sample is sparse. The SRC is a pixel-wise classifier meaning that only the spectral structure is used. The authors of [33] also proposed joint SRC (JSRC) which incorporates the neighborhood information of the given test pixel using a fixed-size square patch. They proved that the use of spatial information greatly affects the accuracy results. In recent years, different JSRC-based HSIC methods have been proposed [36–40] to employ both the spectral and spatial structures efficiently. In [36], multiple fixed-size square patches with different size are used to extract the spatial information better because the optimal size of the patch for the given test pixel may vary in accordance with the spatial geometry. In [37], the shape-adaptive patches are constructed for each test pixel, and then, the pixels are classified with the support of its neighbor samples. Although the methods in [36, 37] provides higher accuracy results than the JSRC, the computational costs of these methods are higher. This is due to the fact that only a given test sample is classified at a time, hence, classifying the entire HSI takes quite a long time. To deal with it, superpixel-based HSIC methods [38–40] have been proposed. A superpixel [41, 42] is a homogeneous region containing pixels with similar spectral signatures, and does not have a fixed-size. In [38], the HSI is first segmented into many superpixels and then each superpixel is classified by the joint-sparse model. Instead of exploiting a single-scale superpixel map, [39] segments the HSI into multiple-scale superpixel maps to extract and utilize the spatial information better than the method proposed in [38]. In order to increase the performance of [39], guided filter (GF) [43] is applied to the multiple classification maps provided by the multiple-scale superpixel maps [40]. Apart from these methods, some researches [44–47] have focused on the impact of the neighbor pixels to the classification performance. The proposed method in [44] devotes distinct weights to neighbors of a test pixel with a nonlocal weighting process. A weighting function is exploited to evaluate the similarity between the test pixel and its neighbors in [45], then, appropriate weights are given to each neighbor pixel using the calculated similarities. In [46], the impact of neighbors of the test pixel with dissimilar spectral characteristics are eliminated us-

ing weighting mechanisms. In [47], different spectral matching algorithms are used to distinguish the similar and dissimilar neighbors of the test sample. All the studies mentioned so far make use of original feature space of the HSI which is insufficient to interpret the broad information underlying the HSI. There are also researches [48–51] on the feature extraction from the HSI, and then, classification with the sparse models. Song *et al.* [48] extracts a series of spatial features using the attribute profiles (AP) and applies the SRC to classify the features. In [49], a shapelet method is utilized to extract the spatial-spectral features in a fixed-size patch, then, the features are classified by the SRC. Jia *et al.* [50] applies Gabor filters with different parameters to obtain underlying feature space in the HSI. Then, considering the impact of multi-feature sets to the classification performance, multitask JSRC is employed to assign predicted labels to the pixels. Instead of using one feature extraction algorithm, Fang *et al.* [51] implements multiple feature extraction methods to obtain different spatial-spectral features where each reflects a distinct property embedded in the HSI. Then, the shape-adaptive regions containing neighbor pixels are extracted and classified by the JSRC.

As mentioned above, many methods have been proposed that calculate spatial similarities in the HSI and exploit these similarities to improve the performance of the sparse models. To further utilize the high spatial similarity and to capture global data information, the sparse models are combined with the low-rank representation (LRR) based methods [52–59]. In [52], a low-rank term is incorporated into the optimization problem of the JSRC to reveal the neighbor similarities and discriminate the samples better. Pan *et al.* [53] obtain the representation matrix as a result of the global low-rank and sparse models with a constraint that measures the similarities between the test and train samples. To obtain the local and global structure of the HSI better, [54] imposes structure and locality regularizations into the low-rank and sparse terms. Ding *et al.* [55] focus on the intraclass representation rather than the interclass representation using locally weighted sparse and low-rank models. In [56], the HSI is first segmented into many superpixels, then, the local low-rank is applied on each superpixel to diminish spectral diversity of the pixels. In this way, the local structure is preserved better. Xue *et al.* [57] treat the HSI as a tensor to utilize the local and global information better using the low-rank and sparse representations. In addition, a dictionary-learning term is included into the optimization problem to increase the representation capacity of the

original dictionary. In [58], the representation matrix of each superpixel is obtained to have discriminative features without noise, and the local affinity matrix using an optimization problem containing local low-rank. Nie *et al.* [59] keep the local and global information of the HSI using structure-prior low-rank and sparsity expressions. The structure constraint enhances the intraclass similarity and interclass variance. Instead of using a pre-given train dictionary, a dictionary-learning stage is employed to obtain discriminative and confident train samples.

Beside the SRC and its variants, collaborative representation classifier (CRC) [60] is noticed as significant as sparsity. The SRC supposes a test pixel as a linear combination of the dictionary, whereas, the CRC deems a test pixel as a collaboration of the dictionary. There are many CRC-based HSIC researches [61–65] in the literature. Li *et al.* [61] calculate the resemblance between the test pixel and train samples utilizing a distance-weighted Tikhonov regularization, and then integrate it to a nearest-regularized subspace (NRS) classifier based on CRC. A spatial version of the NRS is proposed in [62] which calculates the mean of neighbors of the test pixel. It exploits the evaluated average spectrum in the residual calculation where the train samples of the classes are also averaged. Since the neighbor pixels do not contribute equally, weighted joint CRC (WJCRC) is presented in [63] which calculates and gives proper weights to the neighbors. The self-superior points of the CRC, NRS and WJCRC methods are combined and enhanced in [64] to further increase the classification accuracy results. To emphasize the effect of the distinct classes, Shen *et al.* [65] allocates the train samples according to their labels. Then, class-based group weights are evaluated and incorporated into the optimization problem. Similar to the extracted features in the SRC-based methods, there are also many studies focused on the CRC-based classification of the extracted feature sets. For example, discrete wavelet transform (DWT) [66] is implemented on the HSI to obtain the informative features classified by the CRC. In [67], 3D Gabor filters [68] are used to extract the features classified by the CRC. Instead of extracting single feature space, Yang *et al.* [69] exploit many feature extraction algorithm such as spectral value feature (SVF), extended multi-attribute profile (EMAP) [70] and Gabor features to better characterize the HSI. Then, the test pixel is assigned a unique label using the SRC and CRC methods in a unified framework. Jia *et al.* [71] extract the EMAP features which is also filtered by the 3D Gabor

wavelets additionally. Then, the CRC-based pixel-wise classification is performed on the obtained rich spectral features. To enhance the results, the classification map is smoothed by the constructed multiscale superpixels. Liu *et al.* [72] utilize different feature extraction methods such as Gabor features, local binary pattern (LBP) [73], differential morphological profiles (DMP) [74], and the SVF. The coefficient vectors of the train samples are separately calculated for each acquired feature space using the CRC. Then, the residual vector is computed for all features, and the label of the test pixel is determined as the class whose residual is minimum. The methods described so far use different feature extraction algorithms to project the raw HSI features onto a new informative feature space. Then, the CRC-based problems are solved to obtain an optimum coefficient vector of the train samples. Although the accuracy results are satisfactory, the extracted features cannot be used efficiently because it is assumed that the effects of the features in the classification process are equal.

In order to handle with the discussed issue above, relaxed collaborative representation (RCR) [75] which is an extended version of the CRC, has been proposed. The optimization problem of the RCR has the ability to provide optimum coefficient vectors in accordance with the extracted features. Unlike CRC-based methods, the RCR also incorporates the similarity, diversity and impact of the feature sets into the optimization problem. Thanks to this important aspect of the RCR, it has been utilized in the HSI anomaly detection [76], [77] and classification [78–80] studies. The RCR-based multitask learning approach is proposed in [78] on the extracted features such as SVF, spectral gradient feature (SGF) [81], Gabor, and DMP. It is observed that [78] RCR-based multitask classification method provides better accuracy results compared to the CRC-based multitask classification, because the RCR takes into account the resemblance, discrimination and weight of the features. The spatial information is also considered in [78] utilizing a fixed-size patch around the test sample to enhance the accuracy results. In [79], an extended version of the RCR with spectral channel weighting computed on the superpixels is proposed. Since the individual bands of the HSI corresponds to an image, the scene of the each spectral band is partitioned into the same superpixel arrangements. In addition, different features such as EMAP, LBP, and Gabor are extracted and used in the RCR problem. Beside the feature weighting, the spectral bands of the HSI are also weighted. Unlike existing many HSIC methods

using randomly chosen train samples, [79] constructs the dictionary with the selected pixels from every superpixel. In order to use more informative dictionary instead of the atoms obtained from the HSI, Gao *et al.* [80] presented superpixel-guided RCR method with the dictionary learning stage. It first constructs an illuminating dictionary, then assigns class label to each superpixel using the RCR. In lieu of the residual computation, [80] evaluates novel indicators called self and average contributions to classify the superpixels. While the self contribution is exploited to reveal the distinct properties of the classes, the average contribution is used to emphasize the importance of each class. The described RCR-based methods [78–80] provide convincing accuracy results with some drawbacks. For instance, each method operates different feature extraction techniques that have many parameters required to be adjusted for more representative features. After the optimal parameters are set, the comprehensive mathematical formulations of the methods required to be calculated which raises the running time of the methods [78–80] significantly. Beside these points, [78] creates a patch of fixed-size centered at the test pixel to include the neighbors which raises the running time and also induces misclassification around the borders. In [79], the train samples are chosen from the superpixels, which is not preferred in the most of existing HSIC methods. Such random selection may decrease the performance of the method in [79]. The band weighting and dictionary learning stages in [80] may unfavorably affect the running time if such a high-dimensional HSI is need to be classified. Therefore, the existing RCR-based HSIC methods required to be enhanced in terms of feature extraction and computation time.

Recently, a simple yet effective deep learning based feature extraction method named random patches network (RPNet) [82] in which there is no training process, has been introduced. The convolution patches are randomly chosen from the HSI, then, the whole HSI is convolved by the patches. This procedure is repeated until the specified number of layers is reached. In this way, the shallow and deep features found in the HSI are obtained. It means that the output of each layer provides a feature set. As the number of layers increases, the deep features are extracted. After each feature set is obtained, the complete feature matrix is formed by stacking these feature sets, and, classified by the SVM. Due to the its fast, simple and efficacy structure of the RPNet, several RPNet-based approaches have been presented [83–87]. Sun *et al.* [83] propose

random patches convolution and local covariance (RPCC) which first reduces the spectral bands of the HSI with the maximum noise fraction (MNF) [88] algorithm. Then, the dimension-reduced HSI is processed by the RpNet and local covariance matrix representation (LCMR) [89] to extract spatial and spectral features, respectively. The acquired feature sets are combined, and labeled by the SVM. In [84], spectral-spatial RpNet (SSRPNet) which reduces the dimension of the HSI, then obtains the LBP features. The raw HSI and LBP features are concatenated, and then processed by the principal component analysis (PCA) [90] to reduce the dimension of the combined features. After that, the RpNet is utilized to extract the deep features of the dimension-reduced data. In order to classify the spatial-spectral features, a graph-based learning algorithm is implemented. In the Gabor filter with random patch convolution (GRPC) [85], a series of operations such as PCA, Gabor filtering and linear discriminant analysis (LDA) [91] is utilized, and then the output image is fed into the RpNet to extract the deep details. The resultant features of the Gabor and RpNet processes are then classified by the SVM. Uchaev *et al.* [86] introduced RpNet with the recursive filtering (RPNET-RF) which reduces the dimension and filters the features of the RpNet by the PCA and RF, respectively. The use of RF enables to preserve edge details of the HSI. The raw HSI and filtered features are stacked together, and then the combined features are classified by the SVM. In [87], multiscale superpixel-wise RpNet (MSRPNet) which utilizes 2D singular spectrum analysis (SSA) [92] for denoising possible noise in the HSI, and exploits spectral-spatial superpixelwise PCA (S3-PCA) [93] for obtaining local and global features, is presented. After obtaining these features, the RpNet is employed to extract the deep features of the output of the S3-PCA. Then, the extracted deep features are united with the raw HSI features and classified by the SVM. It has been observed that the original RpNet and its enhanced methods provide encouraging outcomes with regard to classification accuracy and time consumption. However, there are still some points required to be handle. For example, the concatenated feature matrices have quite high-dimension prior to be classified, which negatively affect the accuracy results [94]. Additionally, the resemblance, dissimilarity and weight of the extracted features are ignored due to the SVM-based classification.

The following studies have been carried out within the scope of thesis in order to provide solutions to the problems and shortcomings described so far.

1.4 Contributions of the Thesis

In the context of this thesis, three new HSIC methods called

- Weighted and label-consistent joint-sparse classifier (WLC-JSC) [95]
- Locality constraint joint-sparse and weighted low-rank (LCJS-WLR) [96]
- GF of RpNet and RCR (GRR) [97]

have been proposed. Each contribution is shortly explicated in the ensuing subsections prior to the exhaustive explanations, theoretical backgrounds and experiments in the thesis's consequent chapters.

1.4.1 The Proposed WLC-JSC Method

In order to emphasize the unequal contribution of the neighbors around the test pixel, the WLC-JSC applies a weighting rule using the Gaussian weight function. According to the evaluated Gaussian weights and the preset parameters of the thresholds, the contribution of the neighbors are considered in three different cases. It is assumed that some neighbors contribute fully or nothing, and the rest contribute a certain rate. This strategy enables elimination or selection of the irrelevant or relevant neighbors, respectively. Then, in order to find the coefficient matrix, the simultaneous orthogonal matching pursuit (SOMP) [33] method is utilized. Apart from the neighbor weighting, the label-consistencies of the classes in accordance with the test samples are also evaluated by the Gaussian weight function. These weights are utilized in the residual calculation to enforce the test samples to be assigned with the label of the most similar class. A detailed description and formulation of the WLC-JSC is given in Chapter 3.

1.4.2 The Proposed LCJS-WLR Method

To have coefficient matrices with more characteristic and discriminating, the LCJS-WLR implements an optimization problem including weighted low-rank and locality constraint joint-sparse terms for the homogeneous areas. These regions are obtained by the entropy rate superpixels (ERS) [98] which segments the HSI into many non-overlapping homogeneous subregions. Since the pixels in a superpixel exhibit similarity in terms of the spectral feature, the coefficient matrix of the samples should fulfill the low-rank [99, 100] and joint-sparse [33, 44] properties. In order to assign more weights to the train samples having similar spectral characteristics to the superpixel

of interest, the locality constraint is inserted to the joint-sparse term. Also, weighted LRR is exploited [101] to enhance the efficiency of the LRR. Beside these weightings inserted into the optimization problem, the sparse noise elimination term is also incorporated to the optimization due to the possible noise caused by the nature of the HSI. The proposed LCJS-WLR is clarified in detail in Chapter 4.

1.4.3 The Proposed GRR Method

The GRR combines the advantages of the RPNet, which allows to obtain the deep features, and the RCR, which enables the use of the extracted features in an efficient way. Since the spatial information positively affects the classification accuracy results, the GRR also considers the spatial structure using the GF, which is an edge-preserving filter, and superpixels, which constructs homogeneous regions. As discussed before, the current RCR-based algorithms operates some traditional feature extraction methods which take quite a long time to calculate due to the complicated theoretical backgrounds. It is also difficult to the choice of the proper feature extraction methods. Furthermore, the SVM is mostly used for the classification of the features extracted by the RPNet-based algorithms. Since the traditional SVM does not consider the spatial structure and correlation of the features, the RPNet-based methods ignore these key points. To handle with these shortcomings, the GRR is proposed which unifies the outstanding attributes of the RPNet and RCR such as efficient feature extraction and effective usage of the features, respectively. The GRR also improves the classification performance by the use of further processes such as GF for the protection of the borders, LDA for increasing the discrimination capacity of the classifier, and superpixel-guided labeling for the fast and efficient classification. The working scheme of the GRR can be ordered as follows: (1) The dimension of the HSI is reduced by the PCA, and the RPNet is utilized to obtain the rich features. (2) The GF is respectively applied on the original HSI and RPNet features to protect the edge details, which enhances the results [40, 43, 102, 103]. Upon completion of this step, different feature sets are acquired such as RPNet feature, GF of the RPNet feature and GF of the HSI feature. (3) The raw HSI feature is concatenated to these obtained features to maintain the original characteristics of the dataset. (4) The obtained high-dimensional feature sets are separately processed by the LDA to reduce the dimension, to increase the discrimination capability of the classes and to have representative samples. (5) Superpixel-based

pseudo samples are produced using the existent train samples to enhance the efficacy of the LDA because it is a supervised technique and its capability depends on the train samples. (6) The obtained low-dimensional feature sets are pre-classified by the SVM to assign appropriate RCR-weight to each feature set in proportion to the accuracy results. (7) As the last step, the superpixel-guided RCR is implemented to classify the features. The superpixel-based RCR allows the proposed GRR to use the extracted features efficiently, to utilize the spatial structure during classification, and to classify the samples faster in comparison with the pixel-based classification. The extensive explanations, mathematical formulations and experiments of the proposed GRR is presented in Chapter 5. The key points of the proposed GRR can be listed as follows:

- Fast and straightforward HSI feature extraction
- Preserving the edge details of the extracted feature sets
- Protecting the original characteristics of the HSI in each feature set
- Ability to work under few train samples through the pseudo sample generated
- Efficient usage of the features, the spatial structure and the computation time

1.5 Outline of the Thesis

The remainder of this thesis is arranged as follows; the datasets and evaluation metrics used to evaluate the efficacy of the proposed methods are explained in Chapter 2. The detailed backgrounds and formulations of the proposed WLC-JSC, LCJS-WLR and GRR are presented in Chapter 3, Chapter 4 and Chapter 5, respectively. Finally, all inferences obtained from the thesis are discussed and concluded in Chapter 6.

CHAPTER 2

DATASETS AND EVALUATION METRICS

2.1 Indian Pines Dataset

The IP dataset [104] contains 145×145 pixels and 220 bands in the spatial and spectral dimension, respectively. The imaging sensor provides 20 m/pixel spatial resolution, and measures the $0.4\text{--}2.5\text{ }\mu\text{m}$ portion of the spectrum in 10 nm intervals. Some of the spectral bands [104–108, 150–163, 220] are taken out from the dataset because of the water absorption bands which provide no information about the land-cover materials. Hence, size of the dataset obtained for use in experiments becomes $145 \times 145 \times 200$. The IP dataset is collected over a region enclosed by 16 different classes mostly comprised of agricultural products. According to the available sample numbers, two different versions of the IP dataset, i.e., IP1 and IP2, are used in this thesis. Table 2.1 presents the number of sample and the name of each class for both the IP1 and IP2 datasets. Figure 2.1 and Figure 2.2 show the ground-truth and false-color images for the IP1 and IP2 datasets, respectively.

Table 2.1 Sixteen ground-truth classes of the IP1 and IP2 dataset

Class	Name	Samples—IP1	Samples—IP2
1	Alfalfa	54	46
2	Corn-notill	1434	1428
3	Corn-mintill	834	830
4	Corn	234	237
5	Grass-pasture	497	483
6	Grass-trees	747	730
7	Grass-pasture-mowed	26	28
8	Hay-windrowed	489	478
9	Oats	20	20
10	Soybean-notill	968	972
11	Soybean-mintill	2468	2455
12	Soybean-clean	614	593
13	Wheat	212	205
14	Woods	1294	1265
15	Buildings-Grass-Trees-Drives	380	386
16	Stone-Steel-Towers	95	93
Total		10366	10249

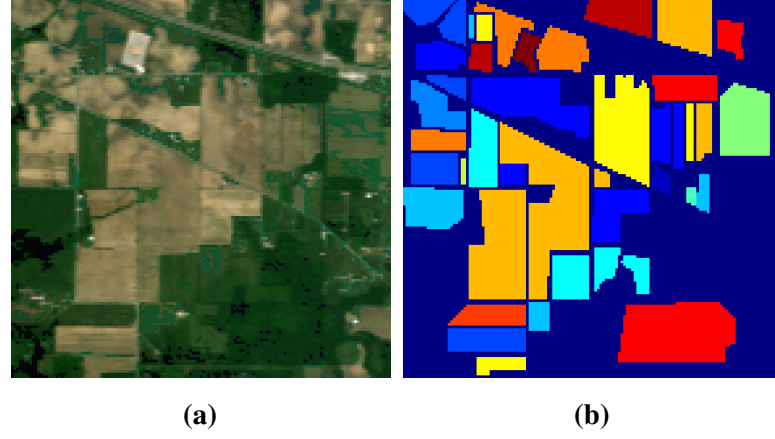


Figure 2.1 IP1 dataset: (a) False-color composite image and (b) Ground-truth.

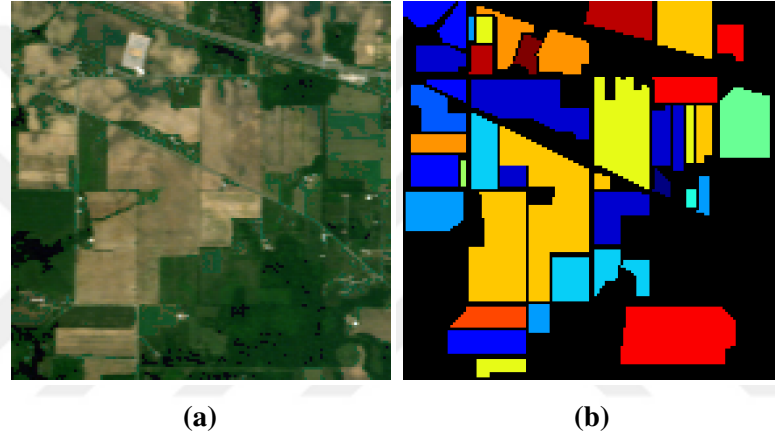


Figure 2.2 IP2 dataset: (a) False-color composite image and (b) Ground-truth.

2.2 Pavia University Dataset

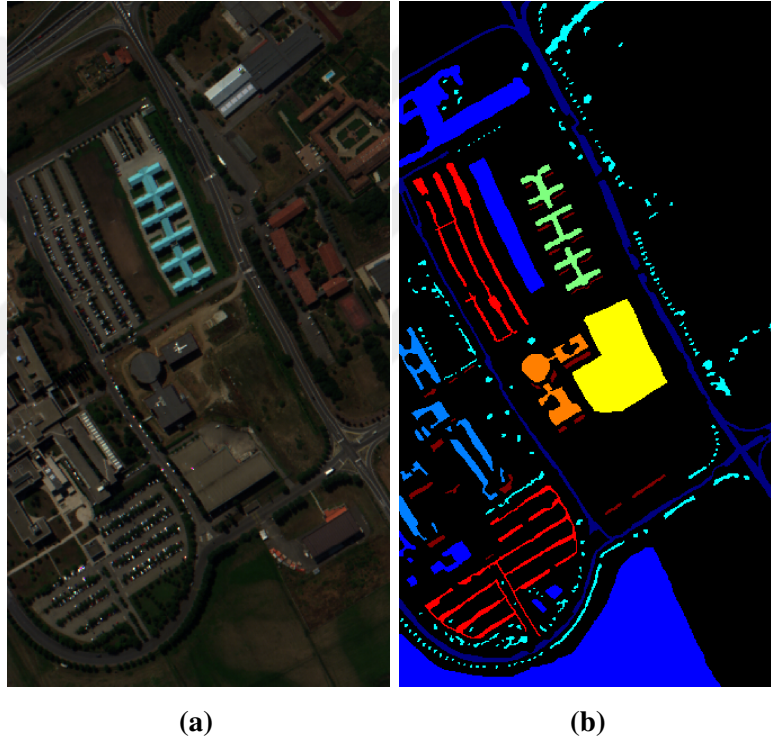
The PU dataset [105] has 610×340 pixels and 103 bands in the spatial and spectral dimension, respectively. The imaging sensor ensures 1.3 m/pixel spatial resolution, and scans the $0.43\text{--}0.86\ \mu\text{m}$ part of the electromagnetic spectrum in 4 nm steps. The PU dataset is gathered over a university area comprised of 9 distinct classes. Table 2.2 indicates the number of sample and the name of each class. Figure 2.3 presents the ground-truth and false-color images.

2.3 WHU-Hi-HongHu Dataset

The large-scale WHHH dataset [106] covers 940×475 pixels and 270 spectral bands in the spatial and spectral dimension, respectively. The imaging sensor has 4.3 cm/pixel

Table 2.2 Nine ground-truth classes of the PU dataset

Class	Name	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
Total		42776

**Figure 2.3** PU dataset: (a) False-color composite image and (b) Ground-truth.

quite high spatial resolution, and spans the $0.4\text{--}1\ \mu\text{m}$ section of the spectrum in approximate 2 nm spacings. The WHHH dataset is obtained over an agricultural field surrounded by 22 different land-cover classes. Table 2.3 introduces the number of sample and the name of each class. Figure 2.4 represents the ground-truth and false-color images.

Table 2.3 Twenty-two ground-truth classes of the WHHH dataset

Class	Name	Samples
1	Red roof	14041
2	Road	3512
3	Bare soil	21821
4	Cotton	163285
5	Cotton firewood	6218
6	Rape	44557
7	Chinese cabbage	24103
8	Pakchoi	4054
9	Cabbage	10819
10	Tuber mustard	12394
11	Brassica parachinensis	11015
12	Brassica chinensis	8954
13	Small Brassica chinensis	22507
14	Lactuca sativa	7356
15	Celtuce	1002
16	Film covered lettuce	7262
17	Romaine lettuce	3010
18	Carrot	3217
19	White radish	8712
20	Garlic sprout	3486
21	Broad bean	1328
22	Tree	4040
Total		386693

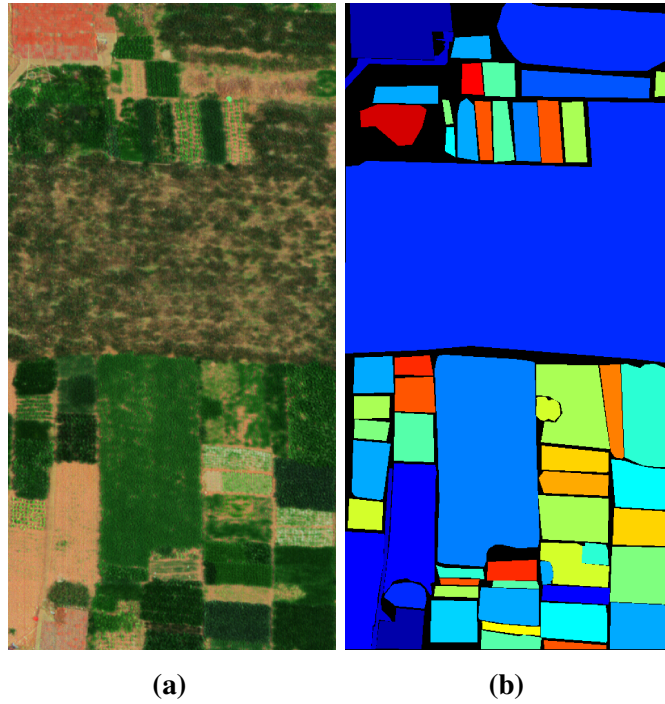


Figure 2.4 WHHH dataset: (a) False-color composite image and (b) Ground-truth.

2.4 Evaluation Metrics

In order to evaluate the classification performance of proposed methods, the metrics [33] such as overall accuracy (OA), average accuracy (AA), kappa statistics (κ) and class-based accuracy (CA) results are utilized. Suppose that the following represents a confusion matrix of two-classes classification problem. Then, the metrics are calculated as follows

		Actual		Total
		Class 1	Class 2	
Predicted	Class 1	a	b	a+b
	Class 2	c	d	c+d
Total		a+c	b+d	N

$$OA = \frac{\text{Number of samples classified accurately}}{\text{Total number of samples}} = \frac{a+d}{N} \quad (2.1)$$

$$\begin{aligned} CA &= \frac{\text{Number of samples classified accurately in the class of interest}}{\text{Total number of samples in the class of interest}} \\ &= \frac{a}{a+c} \quad \text{indicating class accuracy of Class 1 (CA1)} \\ &= \frac{d}{b+d} \quad \text{indicating class accuracy of Class 2 (CA2)} \end{aligned} \quad (2.2)$$

$$AA = \frac{\text{Sum of class accuracy results}}{\text{Total number of classes}} = \frac{CA1 + CA2}{2} \quad (2.3)$$

$$\begin{aligned} \kappa &= \frac{\text{Overall accuracy} - \text{Expected accuracy}}{1 - \text{Expected accuracy}} \\ &= \frac{\left[\frac{a+d}{N} \right] - \left[\frac{(a+c)(a+b)}{N} + \frac{(b+d)(c+d)}{N} \right]}{1 - \left[\frac{(a+c)(a+b)}{N} + \frac{(b+d)(c+d)}{N} \right]} \end{aligned} \quad (2.4)$$

CHAPTER 3

WEIGHTED AND LABEL-CONSISTENT JOINT-SPARSE CLASSIFIER FOR HYPERSPECTRAL IMAGE CLASSIFICATION

3.1 Introduction

The effective use of the spatial information is an important task to obtain high classification accuracy results in HSIC. The proposed WLC-JSC provides a straightforward yet yielding technique to handle with this issue. The neighbors in a patch constructed around the given test pixel are assigned weights based on similarity to the test pixel using the Gaussian weight function. While some of the neighbors are completely discarded, the rests are given distinct weights. After this step, the formulated optimization problem is solved by the SOMP algorithm to obtain the sparse coefficient matrix used for the residual calculation. In addition to the spatial structure, the evaluation of the residual vector is also significant because the label of a sample is determined in accordance with the residuals. Therefore, the label-consistency of the train samples which measures the similarity between the dictionary and test sample, is incorporated into the residual calculation. The following sections provide comprehensive formulations and experiments to clarify the WLC-JSC and to verify the effectiveness of the WLC-JSC, respectively. The experiments are carried out on the IP1 dataset, and recent studies are utilized to compare the results with the proposed WLC-JSC.

3.2 Problem Formulation

A 3D HSI $\mathcal{H} \in \mathbb{R}^{r \times c \times b}$ can be symbolized as a 2D form as $\mathbf{H} \in \mathbb{R}^{b \times m}$ where b and $m (= r \times c)$ indicates the size of spectral channels and the number of pixels in a single band, respectively. Prior to the classification, each column of the $\mathbf{H}$ is normalized into the interval of $[0,1]$ to obtain columns with unit norm. Suppose that the $\mathbf{H}$ is randomly partitioned as train and test samples according to the user-defined percentage. Let $\mathbf{D} \in \mathbb{R}^{b \times n}$ be the dictionary with P classes, where $n = n_1 + \dots + n_P$ denotes the count of available train samples and each $n_p (p = 1, 2, \dots, P)$ represents the count of train

samples in the p th class. A test pixel can be represented as $\mathbf{y}_t \in \mathbb{R}^{b \times 1}$ and the samples in a patch of size $K = k \times k$ can be defined as $\mathbf{Y} = [\mathbf{y}_1 \ \mathbf{y}_2 \ \cdots \ \mathbf{y}_t \ \cdots \ \mathbf{y}_K] \in \mathbb{R}^{b \times K}$.

The method JSRC [33] hypothesizes that the signal $\mathbf{Y}$ containing the given test sample and its neighbors can be sparsely and linearly represented by few atoms in the dictionary $\mathbf{D}$. So, the following optimization problem of the JSRC is written as

$$\hat{\mathbf{X}} = \arg \min \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F \quad \text{subject to} \quad \|\mathbf{X}\|_0 \leq L \quad (3.1)$$

where $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1 \ \hat{\mathbf{x}}_2 \ \cdots \ \hat{\mathbf{x}}_t \ \cdots \ \hat{\mathbf{x}}_K] \in \mathbb{R}^{n \times K}$ denotes the sparse coefficient matrix, $\|\cdot\|_F$ represents the Frobenius norm (i.e., calculates the estimation errors), $\|\cdot\|_0$ specifies the count of non-zero rows of $\mathbf{X}$ and L indicates the sparsity level of the reconstructed signal. In order to solve the equation (3.1), the SOMP [33] algorithm is implemented. The output of this solution gives the joint-sparse coefficient matrix $\hat{\mathbf{X}}$ which has only L non-zero rows. This can be interpreted as the use of only few train samples is sufficient to represent the samples in $\mathbf{Y}$. A pseudocode to calculate $\hat{\mathbf{X}}$ using the SOMP is presented in the Algorithm 1.

Algorithm 1: SOMP

Input:

Pixel set $\mathbf{Y} \in \mathbb{R}^{b \times K}$, Train dictionary $\mathbf{D} \in \mathbb{R}^{b \times n} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \cdots \ \mathbf{d}_n]$

Sparsity level L

Initialize:

Residual $\mathbf{R}_0 = \mathbf{Y}$, Index set $\Phi_0 = \emptyset$, Iteration counter $l = 1$

for $l \leq L$

1. Determine which atom's index λ_l best matches with the residual

$$\lambda_l = \arg \max_{i=1,2,\dots,n} |\langle \mathbf{R}_{l-1}^\top, \mathbf{d}_i \rangle|$$

2. Update the index set $\Phi_l = \Phi_{l-1} \cup \{\lambda_l\}$

3. Calculate the estimated signal using least squares

$$\begin{aligned} \hat{\mathbf{X}}_{\Phi_l} &= \arg \min_{\mathbf{X}_{\Phi_l}} \|\mathbf{Y} - \mathbf{D}_{\Phi_l} \mathbf{X}_{\Phi_l}\|_F \\ &= (\mathbf{D}_{\Phi_l}^\top \mathbf{D}_{\Phi_l})^{-1} \mathbf{D}_{\Phi_l}^\top \mathbf{Y} \end{aligned}$$

4. Compute the new residual $\mathbf{R}_L = \mathbf{Y} - \mathbf{D}_{\Phi_l} \hat{\mathbf{X}}_{\Phi_l}$

5. $l \leftarrow l + 1$

end

Output:

Joint-sparse coefficient matrix $\hat{\mathbf{X}}$ which has nonzero rows indexed at Φ and the value of $(\mathbf{D}_{\Phi}^\top \mathbf{D}_{\Phi})^{-1} \mathbf{D}_{\Phi}^\top \mathbf{Y}$

The next step is to evaluate the residual vector $\mathbf{e} = [e_1 \ e_2 \ \cdots \ e_P]$ for each class. Suppose that each $e_p = 1, 2, \dots, P$ represents a class-based error, and calculated as

$$e_p = \|\mathbf{Y} - \mathbf{D}_p \hat{\mathbf{X}}_p\|_F \quad \text{where } p = 1, 2, \dots, P \quad (3.2)$$

where $\mathbf{D}_p$ indicates the train samples in the p th class and $\hat{\mathbf{X}}_p$ points out the coefficients of the class p . Finally, the given test pixel $\mathbf{y}_t$ is assigned to a class whose residual is the least. The following equation is computed to obtain the label of $\mathbf{y}_t$ as

$$\text{class}(\mathbf{y}_t) = \arg \min(\mathbf{e}) \quad (3.3)$$

As can be seen that, the JSRC method does not consider the impact of each neighbors and assumes the equal contribution of the neighbors. Suppose that a given test sample is existed around the class borders. In such a situation, the patch around the test sample may contain neighbors of the different classes. This issue may negatively affect the classification accuracy results since the spectral signatures of the distinct objects (i.e., classes) are different. In order to handle with this problem, the proposed WLC-JSC considers the impact of the neighbors in the patch using a weight assignment strategy. The neighbors of the test sample $\mathbf{y}_t$ are given suitable weights based on the similarities to the $\mathbf{y}_t$. To estimate the weights, a Gaussian weight function is utilized. According to the pre-refined thresholds and the estimated weights, the irrelevant neighbors are entirely discarded, and the rest are assigned appropriate weights. The following Gaussian weight function is utilized to obtain the weights between $\mathbf{y}_t$ and its spatial neighbors in $\mathbf{Y} = [\mathbf{y}_1 \ \mathbf{y}_2 \ \cdots \ \mathbf{y}_t \ \cdots \ \mathbf{y}_K]$ as

$$w_{i=1,2,\dots,K} = e^{-\left(\frac{\|\mathbf{y}_t - \mathbf{y}_i\|_2}{2\sigma^2}\right)} \quad (3.4)$$

where σ controls the degree of weighting and $\|\cdot\|_2$ indicates the l_2 norm of vector difference. After the weights are calculated, two parameters such as w_{low} and w_{high} are used to readjust the weights as follows

$$\tilde{w}_{i=1,2,\dots,K} = \left\{ \begin{array}{ll} 1, & \text{if } w_i \geq w_{high} \\ w_i, & \text{if } w_{low} < w_i < w_{high} \\ 0, & \text{if } w_i \leq w_{low} \end{array} \right\} \quad (3.5)$$

According to the equation (3.5), the neighbors having weights lower than w_{low} are removed, the neighbors having weights higher than w_{high} are assigned full weights,

and the neighbors having weights amongst w_{low} and w_{high} are considered as less importance. To incorporate these readjusted weights into the neighborhood matrix $\mathbf{Y}$, a diagonal matrix $\mathbf{W} \in \mathbb{R}^{K \times K}$ whose entries are formed by the rearranged weights $\tilde{w}_{i=1,2,\dots,K}$, is created as

$$\mathbf{W} = \begin{bmatrix} \tilde{w}_1 & 0 & \cdots & 0 \\ 0 & \tilde{w}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \tilde{w}_K \end{bmatrix}_{K \times K} \quad (3.6)$$

Once $\mathbf{W}$ is provided, the weights are multiplied by the spatial neighbors in $\mathbf{Y}$, and a weighted matrix of the spatial neighbors $\mathbf{Y}_w$ is obtained as

$$\mathbf{Y}_w = \mathbf{Y}\mathbf{W} \quad (3.7)$$

In $\mathbf{Y}_w$, some of the columns (i.e., neighbors) are entirely zero due to the zero weights calculated in the equation (3.5). These vectors should be removed from $\mathbf{Y}_w$ to discard unrelated neighbors of $\mathbf{y}_t$. After discarding the neighbors with no effect, the arranged weighted matrix of the spatial neighbors $\tilde{\mathbf{Y}}_w = [\tilde{\mathbf{y}}_1 \ \tilde{\mathbf{y}}_2 \ \cdots \ \mathbf{y}_t \ \cdots \ \tilde{\mathbf{y}}_T]$ with size of $b \times T$ ($T < K$) is constructed. Due to the matrix multiplication process in the equation (3.7), the vectors in $\tilde{\mathbf{Y}}_w$ do not have unit norm anymore. Then, it is essential to normalize the columns of $\tilde{\mathbf{Y}}_w$ as follows

$$\tilde{\mathbf{Y}}_w(:, j) = \frac{\tilde{\mathbf{Y}}_w(:, j)}{\|\tilde{\mathbf{Y}}_w(:, j)\|_2} \quad \text{where } j = 1, 2, \dots, T \quad (3.8)$$

As in the JSRC, the latter stage is the estimation of the joint-sparse coefficient matrix $\hat{\mathbf{X}}_w$ affiliated with the $\mathbf{Y}_w$ as

$$\hat{\mathbf{X}}_w = \arg \min \|\tilde{\mathbf{Y}}_w - \mathbf{D}\mathbf{X}_w\|_F \quad \text{subject to} \quad \|\mathbf{X}_w\|_0 \leq I \quad (3.9)$$

where I denotes the level of sparsity of the proposed WLC-JSC. After obtaining the joint-sparse representation matrix $\hat{\mathbf{X}}_w$, the residual vector $\mathbf{e}$ should be calculated. Different from the JSRC, the proposed WLC-JSC also associates the label-consistencies between the train and test samples in each class into the residual calculation. For this, the mean of each class in the dictionary $\mathbf{D}$ is calculated, and then the similarities between the averaged classes and samples in $\tilde{\mathbf{Y}}_w$ are evaluated.

In order to obtain the mean matrix $\mathbf{D}_{mean} \in \mathbb{R}^{b \times P}$ containing class-based averaged samples in the dictionary, the following equation is computed as

$$\mathbf{D}_{mean}(:, p) = \frac{1}{n_p} \sum_{s=1}^{n_p} \mathbf{D}_p^s \quad (3.10)$$

where $\mathbf{D}_p^s$ denotes the s th train sample in the p th class. Similar to the equation (3.4), the Gaussian weight function is utilized to construct the similarity matrix $\mathbf{W}_{sim} \in \mathbb{R}^{P \times T}$ between $\mathbf{D}_{mean}$ and $\tilde{\mathbf{Y}}_w$ as follows

$$\mathbf{W}_{sim}(p, j) = e^{-\left(\frac{\|\tilde{\mathbf{Y}}_j - \mathbf{D}_{mean}(:, p)\|_2}{2\sigma^2}\right)} \quad \text{where} \quad \begin{matrix} p = 1, 2, \dots, P \\ j = 1, 2, \dots, T \end{matrix} \quad (3.11)$$

Each column of $\mathbf{W}_{sim}$ represents the similarities between a sample in $\tilde{\mathbf{Y}}_w$ and the mean of train samples in each class. Each row of $\mathbf{W}_{sim}$ denotes the similarities between all samples in $\tilde{\mathbf{Y}}_w$ and the mean of train samples in a certain class. Hence, if the rows of $\hat{\mathbf{X}}_w$ relating to one class and one row of $\mathbf{W}_{sim}$ relating to the same class are multiplied by element-wise, the label consistency is incorporated into the residual computation. In order to multiply element-wise, dimension of the two matrices have to be same. So, while computing the residual vector, the coincident row of the $\mathbf{W}_{sim}$ is enlarged and replicated as the number of rows in $\hat{\mathbf{X}}_w$ in the corresponding class. Let $\mathbf{w}_{sim}^p \in \mathbb{R}^{1 \times T}$ denotes a row vector of $\mathbf{W}_{sim}$ in the p th class and $\hat{\mathbf{X}}_w^p \in \mathbb{R}^{n_p \times T}$ represents the joint-sparse coefficient matrix of the p th class, then

$$\mathbf{W}_{sim}^p = repmat(\mathbf{w}_{sim}^p, \hat{\mathbf{X}}_w^p) \quad (3.12)$$

where the operator $repmat(\cdot, \cdot)$ copies the row elements of $\mathbf{w}_{sim}^p$ as the count of rows n_p of $\hat{\mathbf{X}}_w^p$. The output of this operator generates $\mathbf{W}_{sim}^p \in \mathbb{R}^{n_p \times T}$ for the p th class. Using these calculated $\mathbf{W}_{sim}^p$ for each class, the residual vector $\mathbf{e} = [e_1 \ e_2 \ \dots \ e_P]$ is obtained as follows

$$e_p = \|\tilde{\mathbf{Y}}_w - \mathbf{D}_p(\hat{\mathbf{X}}_w^p \odot \mathbf{W}_{sim}^p)\|_F \quad \text{where} \quad p = 1, 2, \dots, P \quad (3.13)$$

where the mathematical operation $\odot$ stands for the element-wise multiplication of the matrices. At last, label of the test pixel $\mathbf{y}_t$ is assigned as the class with the least residual. A flowchart of the proposed WLC-JSC method is presented in Figure 3.1.

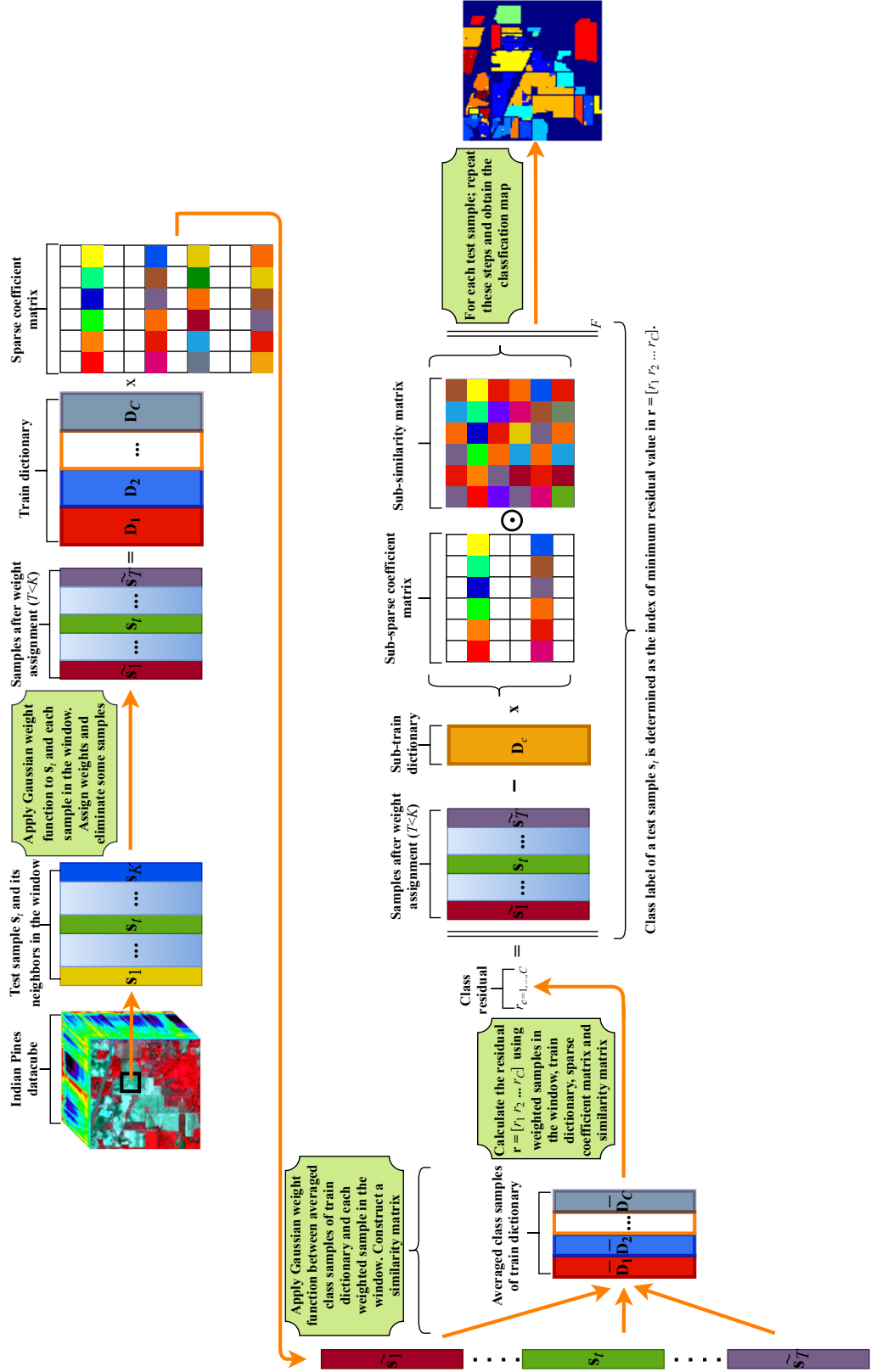
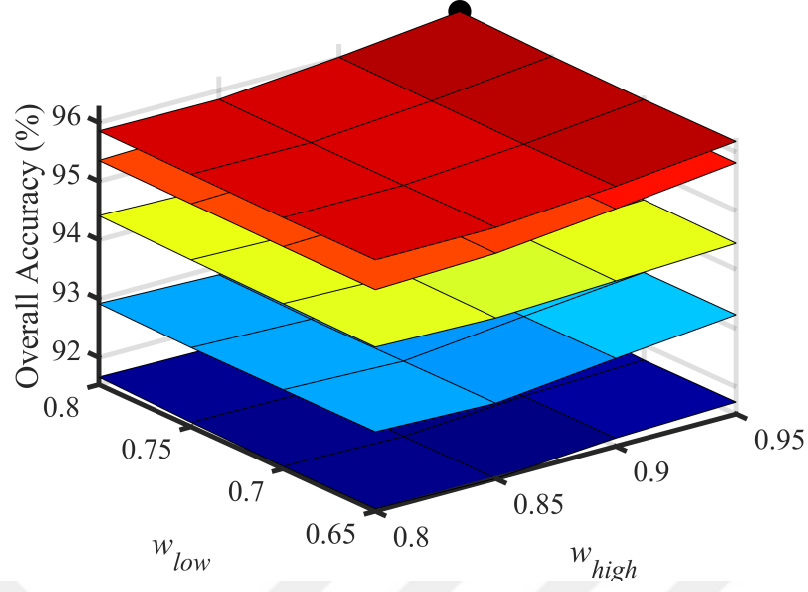


Figure 3.1 Flowchart of the proposed WLC-JSC method.

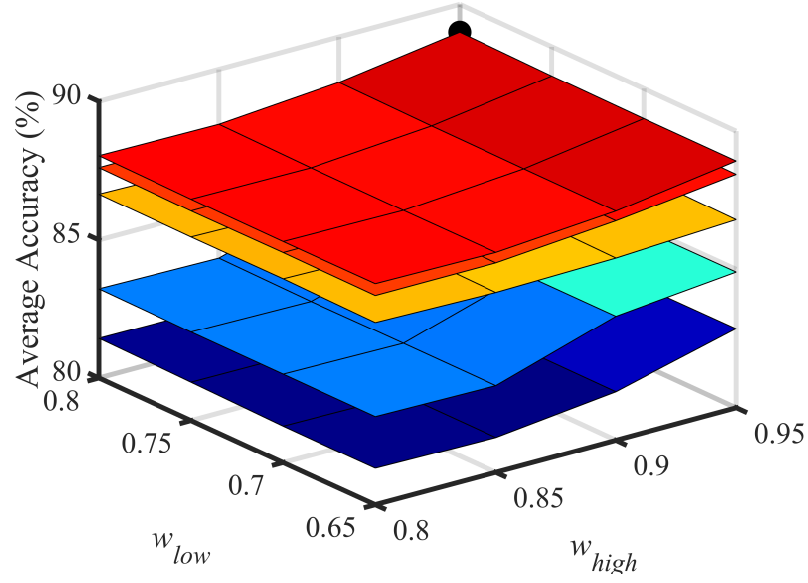
3.3 Experimental Results

To verify the efficacy of the WLC-JSC in terms of classification accuracies compared to the recent methods, the IP1 dataset is employed. The HSI is partitioned as train and test samples by choosing 10% and 90% samples per class, respectively. Since the samples are randomly separated, the experiments including the parameter evaluation and performance comparison, is reiterated 10 times and then the results are averaged. The recent and popular methods such as SVM [107], SRC [33], JSRC [33], joint-sparse representation nearest neighbor (JSRNN) [108], non-local weighted JSRC (NLW-JSRC) [44] and correlation coefficient JSR (CCJSR) [109] are used to compare the results with the WLC-JSC. Among them, the SVM and SRC use only the spectral information, and the rest utilize both the spectral and spatial information during the classification stage. The classification capability of the methods are evaluated with respect to the overall accuracy (OA), average accuracy (AA), kappa statistics (κ) and class-based accuracy results. The optimal parameter settings of the methods, which are specified by the authors of the original papers [33, 108, 44, 109], are utilized throughout the experiments. Since the classes of the HSI are not linearly separable, the SVM is operated with the Gaussian radial basis function kernel whose parameters are adjusted by five-fold cross validation. The single parameter of the SRC called sparsity level is set to 5. Similar to the SRC, the sparsity level and patch size of the JSRC is adjusted as 30 and 9×9 , respectively. For JSRNN, higher bound of the sparsity level, the residual fault tolerance and the patch size are tuned as 100, 0.1 and 5×5 , respectively. The local and nonlocal patch sizes, and the sparsity level of the NLW-JSRC are set as 9×9 , 7×7 and 30, respectively. The patch size, level of sparsity, regularization parameter and count of the closest train samples of the CCJSR are utilized as 6×6 , 2, 0.6 and 4, respectively.

The WLC-JSC has several parameters to be tuned called patch size K , sparsity level I , Gaussian weighting parameter σ and the thresholds w_{low} and w_{high} . To adjust the optimal values of I , w_{low} and w_{high} ; K and σ are fixed as 5×5 and 0.5, respectively. Then, the intervals of parameters for $I = \{2, 4, 6, 8, 10\}$, $w_{low} = \{0.65, 0.70, 0.75, 0.80\}$ and $w_{high} = \{0.80, 0.85, 0.90, 0.95\}$ are chosen and cross-validated by changing



(a)



(b)

Figure 3.2 IP1 dataset: OA and AA results of the proposed WLC-JSC with the chosen parameter intervals for I , w_{low} and w_{high} . (For the sub-figures, the layers from top to bottom point out the results for $I = \{2, 4, 6, 8, 10\}$, respectively. The black markers indicate the highest accuracies).

the search parameter and keeping the remaining as constant. The OA and AA metrics are evaluated and tracked with these changing parameters to determine the optimal parameters. According to the Figure 3.2, the best accuracy results are obtained by the parameters $I = 2$, $w_{low} = 0.80$ and $w_{high} = 0.95$.

After the experiments of parameter tuning are fulfilled, the classification performance of the proposed WLC-JSC is compared to the methods mentioned above. The OA, AA, κ , class-based accuracies and the number of partitioned train and test samples are presented in Table 3.1. It is clearly seen that the SVM and SRC using only the spectral information provide lower accuracy results than the others utilizing also the spatial information. From these results, it can be inferred that the usage of spatial information enhances the accuracy results. Compared to the results of the SRC, the JSRC provides better results because it incorporates the spatial structure of the HSI using a patch around the test sample. The OA and AA results of the JSRC are approximately 21% and 12% higher than the results of the SRC. The use of nonlocal weighting scheme of the neighbors in the NLW-JSRC affects the AA results rather than the OA compared to the JSRC. As can be seen, the OAs of the both JSRC and NLW-JSRC are quite similar while the NLW-JSRC increases the AA results of the JSRC by 4% nearly. The direct usage of the neighbors without any weighting, and nearest-neighbor-based residual calculation in the JSRNN increase the results of the JSRC in terms of OA and AA by approximately 1% and 7%, respectively. The correlation-coefficient-based residual calculation in the CCJSR considers the similarity between train and test samples. It is observed that the improved residual computation increases the OA and AA results of the JSRC by almost 1.5% and 9%. The analyzed results show that weighting of the neighbors and advanced residual evaluation enhance the accuracy results. The proposed WLC-JSC attains the highest OA, AA and κ results in comparison with the state-of-the-arts algorithms. It is seen that the WLC-JSC advances OA of the JSRC, JSRNN, NLW-JSRC, CCJSR by nearly 2%, 1%, 2%, 0.5% and AA by approximately 10%, 2.5%, 5.5%, 1%, respectively. Hence, the quantitative results indicate that the weighting and incorporating label-consistency strategies of the proposed WLC-JSC contributed to the literature.

In order to see the classification results visually, the classification maps of the methods are given in Figure 3.3. It is seen that the maps of SVM and SRC exhibit non-smooth characteristics due to the lack of spatial information. Since the other methods including the proposed WLC-JSRC utilize both the spatial-spectral information in the HSI, their classification maps clear off the non-smooth characteristics. The visual results also demonstrate that the proposed WLC-JSC exhibits the best classification performance.

Table 3.1 IP1 dataset: Classification accuracy results (%) of SVM, SRC, JSRC, JSRNN, NLW-JSRC, CCJSR and proposed WLC-JSC
 *The highest and second highest results are bolded and underlined, respectively.

Class	Train	Test	SVM	SRC	JSRC	JSRNN	NLW-JSRC	CCJSR	WLC-JSC
1	6	48	79.17	56.25	85.42	89.58	89.58	<u>87.50</u>	89.58
2	144	1290	86.36	65.89	93.02	96.82	95.66	<u>96.98</u>	97.67
3	84	750	76.40	58.27	91.47	88.00	87.87	90.80	<u>91.20</u>
4	24	210	79.05	45.71	94.29	81.90	85.71	96.19	93.81
5	50	447	94.63	90.16	89.71	95.75	<u>96.64</u>	96.87	96.20
6	75	672	96.13	91.67	98.96	98.81	100	97.92	98.36
7	3	23	73.91	65.22	13.04	<u>82.61</u>	47.80	86.96	<u>82.61</u>
8	49	440	98.86	97.95	100	100	100	100	100
9	2	18	66.67	44.44	0	50.00	27.78	38.89	<u>61.11</u>
10	97	871	76.00	71.76	88.52	94.72	88.98	94.49	95.64
11	247	2221	79.02	71.90	97.75	94.64	<u>96.62</u>	95.14	95.59
12	62	552	84.78	54.89	86.41	95.11	91.12	96.20	95.65
13	22	190	98.95	97.37	98.95	100	100	<u>99.47</u>	<u>99.47</u>
14	130	1164	93.04	92.18	99.40	99.83	99.05	100	100
15	38	342	57.02	43.57	96.78	92.98	93.27	92.11	<u>95.91</u>
16	10	85	92.94	92.94	95.29	87.06	97.65	98.82	94.12
OA	—	—	84.24	73.94	94.49	95.24	<u>94.87</u>	<u>96.03</u>	96.42
AA	—	—	83.31	71.26	83.06	90.49	87.36	<u>91.77</u>	92.93
κ	—	—	82.08	70.31	93.71	94.58	94.14	<u>95.48</u>	95.93

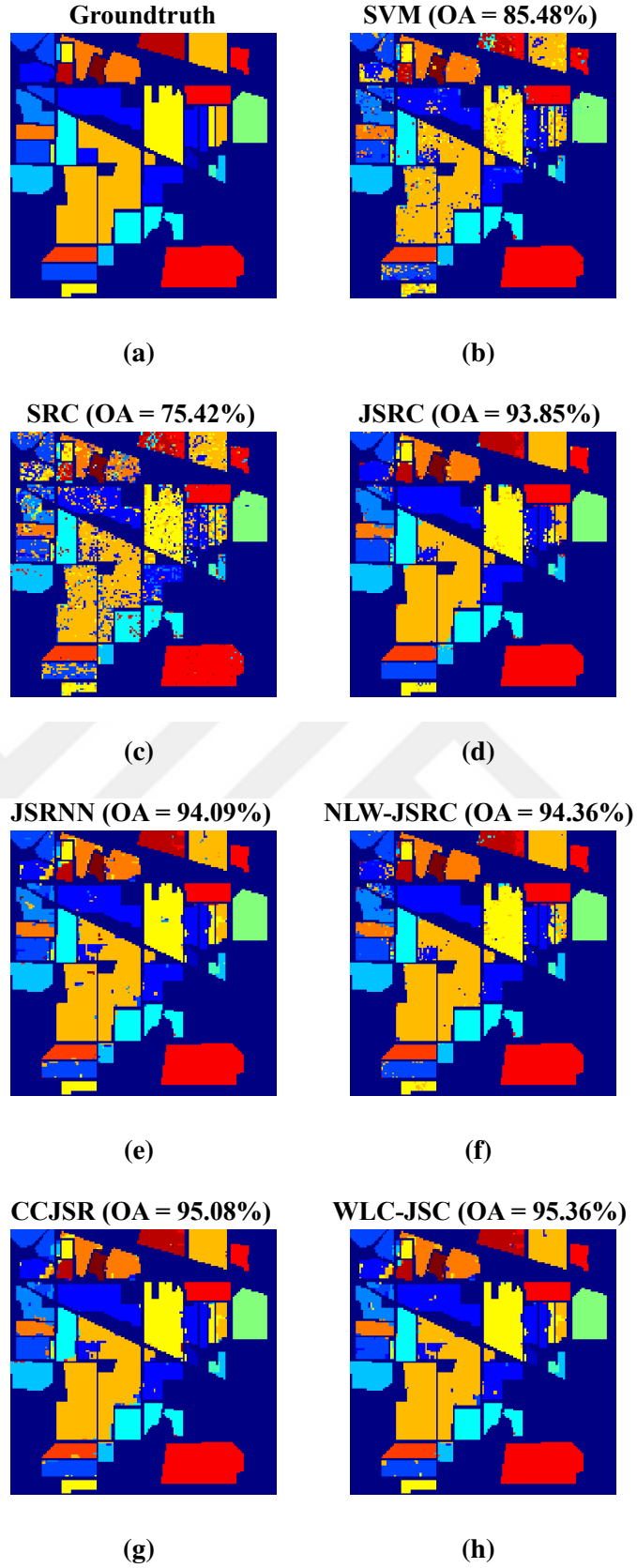


Figure 3.3 IP1 dataset: (a) Ground-truth and classification maps obtained by (b) SVM, (c) SRC, (d) JSRC, (e) JSRNN, (f) NLW-JSRC, (g) CCJSR and (h) proposed WLC-JSC.

3.4 Summary

This chapter has been provided comprehensive descriptions, formulations and experimental studies related to the proposed WLC-JSC method. Both quantitative and qualitative experimental results demonstrated that the WLC-JSC provides satisfactory classification results and outperforms the recent state-of-the-art methods in the literature. It has been proven that the Gaussian weight function-based neighborhood weighting eliminates the irrelevant neighbors while keeping and suppressing the suitable neighbors. In addition to the neighbor weighting, the proposed WLC-JSC also incorporates the label-consistency into the residual computation by evaluating class-based similarities between the dictionary, and selected neighbor samples in the patch. In this way, the chance of assigning the test pixel to the convenient class is improved.

CHAPTER 4

LOCALITY CONSTRAINT JOINT-SPARSE AND WEIGHTED LOW-RANK BASED HYPERSPECTRAL IMAGE CLASSIFICATION

4.1 Introduction

In this chapter, the proposed spectral-spatial HSIC method called LCJS-WLR is introduced in detail. The method solves locality-constraint joint-sparse and weighted low-rank based optimization problem for the local regions obtained by superpixel segmentation. Since the pixels in a local region have similar spectral characteristics, their representation matrix should satisfy the joint-sparse and low-rank circumstances. The raw joint-sparse term in the optimization problem does not consider the impact of each train sample, however, it is essential to incorporate the similarities between the train and test samples. Therefore, the locality-constraint term, which measures the similarities of the train and test samples, is involved to the optimization problem by inserting an additional term to the joint-sparse function. Besides, the original low-rank term is replaced by the weighted low-rank expression to increase the low-rank characteristic of the HSI. Due to the possible Gaussian and sparse noises caused by the imaging sensor, the noise elimination terms are also integrated into the optimization problem. The extensive experiments are conducted on the IP1 dataset to verify the effectiveness of the proposed LCJS-WLR classifier.

4.2 Problem Formulation

A 3D HSI is symbolized as $\mathcal{H} \in \mathbb{R}^{r \times c \times b}$, and can be transformed to the 2D form as $\mathbf{H} \in \mathbb{R}^{b \times rc}$ where b is the number of spectral bands and the multiplication rc represents the count of pixels at each spectral bands. The proposed LCJS-WLR starts with the ERS-based superpixel segmentation to segment the HSI into many non-overlapping sub-regions. To utilize the ERS, a base image and an appropriate number of superpixels are required to segment the HSI. The first-three components of the PCA output is utilized as the base image which has to be gray-level or colorful image. The selection of correct

number of superpixel is quite important because it directly affects the classification accuracy results. Since the aim of the proposed LCJS-WLR is to solve the optimization problem in local regions, the areas should be formed as homogeneous as possible. In order to ensure this, a simple yet efficient technique in [40] is utilized to determine the number of superpixels (*SupNum*) as follows

$$SupNum = \frac{rc}{T \times T} \quad (4.1)$$

where equation (4.1) guarantees that each superpixel has almost $T \times T$ pixels. Let $\mathbf{Y} \in \mathbb{R}^{b \times K}$ be the samples inside a superpixel including K samples with similar spectral characteristics. Due to the possible sparse and Gaussian noises in the HSI, the samples $\mathbf{Y}$ can be estimated as

$$\mathbf{Y} = \mathbf{D}\mathbf{Z} + \mathbf{N} + \mathbf{E} \quad (4.2)$$

where $\mathbf{D} \in \mathbb{R}^{b \times n}$ indicates the train dictionary and n denotes the number of available train samples. Since the dictionary includes train samples from different classes, it can be arranged as $\mathbf{D} = [\mathbf{D}_1 \ \mathbf{D}_2 \ \cdots \ \mathbf{D}_P]$ where P represents the number of classes. The sub-dictionaries containing train samples of distinct classes can be written as $\mathbf{D}_{p=1, \dots, P} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \cdots \ \mathbf{d}_{n_p}]$ and $n = \sum_{p=1}^P n_p$. The expression $\mathbf{Z} \in \mathbb{R}^{n \times K}$ denotes the joint-sparse coefficient matrix, $\mathbf{N} \in \mathbb{R}^{b \times K}$ and $\mathbf{E} \in \mathbb{R}^{b \times K}$ represent the potential Gaussian and sparse noises, respectively. Since the pixels inside a superpixel have similar spectral signatures, then, their coefficient matrix should meet the joint-sparse and low-rank properties. Then, the optimization problem can be designed as

$$\min_{\mathbf{Z}, \mathbf{E}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\mathbf{Z} - \mathbf{E}\|_F^2 + \gamma \|\mathbf{Z}\|_{2,1} + \tau \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1} \quad (4.3)$$

where $\|\mathbf{Z}\|_{2,1} = \sum_{u=1}^n \|\mathbf{z}^u\|_2$ means the $l_{2,1}$ norm of $\mathbf{Z}$ where $\mathbf{z}^u$ indicates the u th row vector of $\mathbf{Z}$ and $\|\mathbf{E}\|_{2,1} = \sum_{j=1}^K \|\mathbf{e}^j\|_2$ means the $l_{2,1}$ norm of $\mathbf{E}$ where $\mathbf{e}^j$ points out the j th column vector of $\mathbf{E}$. The nuclear norm of $\mathbf{Z}$ is symbolized as $\|\mathbf{Z}\|_* = \sum_k \sigma_k(\mathbf{Z})$ where $\sigma_k(\mathbf{Z})$ stands for the k th singular value of $\mathbf{Z}$. While $\|\mathbf{Z}\|_{2,1}$ norm encourages the row-sparse structure of $\mathbf{Z}$, $\|\mathbf{E}\|_{2,1}$ norm finds the clean or corrupted samples in $\mathbf{Y}$.

Current researches [110–113] have proven that the locality adaptor, which is integrated to the joint-sparse matrix, improves the performance of the classifier by incorporating the similarities between the train and test samples. The purpose of this additional term is to give suitable weights to the train samples. While the coefficients of train samples,

assigned large weights, are reset to zero as a result of optimization, the remaining train samples, assigned small weights, are presented in the joint-sparse coefficient matrix. In this wise, the impact of irrelevant train samples are completely eliminated by the optimizer. This increases the potential of train samples, belonging to the same class as the test samples, found in the coefficient matrix. Therefore, the proposed LCJS-WLR imposes the locality information into the equation (4.3). On the other hand, the recent studies [114–117] have shown that the usage of weighted low-rank in place of low-rank increases the results. After these arrangements, the proposed LCJS-WLR is re-formulated as the following

$$\min_{\mathbf{Z}, \mathbf{E}} \frac{1}{2} \|\mathbf{Y} - \mathbf{D}\mathbf{Z} - \mathbf{E}\|_F^2 + \gamma \|\mathbf{C} \odot \mathbf{Z}\|_{2,1} + \tau \|\mathbf{Z}\|_{\omega,*} + \lambda \|\mathbf{E}\|_{2,1} \quad (4.4)$$

where $\mathbf{C} \in \mathbb{R}^{n \times K}$ denotes the locality adaptor and the operator $\odot$ indicates the element-wise multiplication. The choice of weighting function is of great importance because it should reveal the similarity between dictionary and superpixel thoroughly. Let $[\mathbf{y}_1, \dots, \mathbf{y}_K]$ be the pixels inside a superpixel, and $[\mathbf{d}_1, \dots, \mathbf{d}_n]$ be the train samples. Suppose that the coordinates of the train samples and test samples within a superpixel are normalized to the range of $[0,1]$, and denoted as $[\mathbf{y}_{1_l}, \dots, \mathbf{y}_{K_l}]$ and $[\mathbf{d}_{1_l}, \dots, \mathbf{d}_{n_l}]$, respectively. Then, the locality adaptor $\mathbf{C}$, which utilizes the both spectral and coordinate information of the samples, can be calculated as

$$\mathbf{C} = \sqrt{(\mathbf{y}_a - \mathbf{d}_i)^2 + (\mathbf{y}_{a_l} - \mathbf{d}_{i_l})^2} \quad (4.5)$$

where $a = 1, \dots, K$ and $i = 1, \dots, n$

The weighted nuclear norm of $\mathbf{Z}$ in equation (4.4) can be explicitly formulated as

$$\|\mathbf{Z}\|_{\omega,*} = \sum_{k=1}^{\text{rank}(\mathbf{Z})} \omega_k \sigma_k(\mathbf{Z}) \quad (4.6)$$

where ω_k represents the nonnegative weight coefficient of the k th singular value of $\mathbf{Z}$. Suppose that the current estimation of $\mathbf{Z}$ is written as $\mathbf{Z}^{(t)}$ at the t th iteration. Then, the coefficient ω_k at the iteration t is computed as [114]

$$\omega_k^{(t)} = \frac{1}{\sigma_k(\mathbf{Z}^{(t)}) + \epsilon} \quad (4.7)$$

where the constant $\epsilon = 10^{-16}$ is used to prevent possible singularities. In order to solve the proposed optimization problem in equation (4.4), alternating direction method of

multipliers (ADMM) [118] algorithm, which updates the variables alternately, is implemented. According to the ADMM, the equation (4.4) can be solved by defining three new variables such as $\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3$. Then, the equation (4.4) can be re-written as

$$\begin{aligned} \min_{\mathbf{Z}, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{E}} & \left\{ \frac{1}{2} \|\mathbf{Y} - \mathbf{V}_1 - \mathbf{E}\|_F^2 + \gamma \|\mathbf{C} \odot \mathbf{V}_2\|_{2,1} \right. \\ & \left. + \tau \|\mathbf{V}_3\|_{\omega,*} + \lambda \|\mathbf{E}\|_{2,1} \right\} \\ \text{s.t. } & \mathbf{V}_1 = \mathbf{DZ}, \mathbf{V}_2 = \mathbf{Z}, \mathbf{V}_3 = \mathbf{Z} \end{aligned} \quad (4.8)$$

which can be stated in a closed form as follow

$$\min_{\mathbf{Z}, \mathbf{V}, \mathbf{E}} f(\mathbf{V}, \mathbf{E}) \quad \text{s.t.} \quad \mathbf{JZ} + \mathbf{FV} = \mathbf{0} \quad (4.9)$$

where

$$\begin{aligned} f(\mathbf{V}, \mathbf{E}) &= \frac{1}{2} \|\mathbf{Y} - \mathbf{V}_1 - \mathbf{E}\|_F^2 + \gamma \|\mathbf{C} \odot \mathbf{V}_2\|_{2,1} + \tau \|\mathbf{V}_3\|_{\omega,*} + \lambda \|\mathbf{E}\|_{2,1} \\ \mathbf{V} &= \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix}, \mathbf{J} = \begin{bmatrix} \mathbf{D} \\ \mathbf{I} \\ \mathbf{I} \end{bmatrix}, \mathbf{F} = \begin{bmatrix} -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{I} \end{bmatrix} \end{aligned} \quad (4.10)$$

where $\mathbf{I}$ represents the identity matrix. The augmented Lagrangian function of the problem in equation (4.9) is written as

$$\mathcal{L}(\mathbf{V}, \mathbf{Z}, \mathbf{E}, \mathbf{P}) = f(\mathbf{V}, \mathbf{E}) + \frac{\mu}{2} \|\mathbf{JZ} + \mathbf{FV} - \mathbf{P}\|_F^2 \quad (4.11)$$

where $\mathbf{P} = [\mathbf{P}_1^T \quad \mathbf{P}_2^T \quad \mathbf{P}_3^T]^T$, and $\mathbf{P}_{h=1,2,3}/\mu$ are the Lagrangian multipliers where $\mu > 0$ is the penalty variable. The Lagrangian function in the equation (4.11) can be expanded as

$$\begin{aligned} \mathcal{L}(\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{Z}, \mathbf{E}, \mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3) &= \frac{1}{2} \|\mathbf{Y} - \mathbf{V}_1 - \mathbf{E}\|_F^2 + \gamma \|\mathbf{C} \odot \mathbf{V}_2\|_{2,1} + \tau \|\mathbf{V}_3\|_{\omega,*} \\ &+ \lambda \|\mathbf{E}\|_{2,1} + \frac{\mu}{2} \left(\|\mathbf{DZ} - \mathbf{V}_1 - \mathbf{P}_1\|_F^2 + \|\mathbf{Z} - \mathbf{V}_2 - \mathbf{P}_2\|_F^2 \right. \\ &\quad \left. + \|\mathbf{Z} - \mathbf{V}_3 - \mathbf{P}_3\|_F^2 \right) \end{aligned} \quad (4.12)$$

After formulating the equation (4.12), each variable should be computed alternately. The solution of $\mathbf{Z}$ is provided as

$$\mathbf{Z}^{(t+1)} \leftarrow (\mathbf{D}^T \mathbf{D} + 2\mathbf{I})^{-1} \left[\mathbf{D}^T (\mathbf{V}_1^{(t)} + \mathbf{P}_1^{(t)}) + \mathbf{V}_2^{(t)} + \mathbf{P}_2^{(t)} + \mathbf{V}_3^{(t)} + \mathbf{P}_3^{(t)} \right] \quad (4.13)$$

The reduced optimization problem of $\mathbf{E}$ after taking the derivative of the equation (4.12) with respect to $\mathbf{E}$, is written as

$$\mathbf{E}^{(t+1)} \leftarrow \arg \min_{\mathbf{E}} \frac{1}{2} \|\mathbf{Y} - \mathbf{V}_1^{(t)} - \mathbf{E}\|_F^2 + \lambda \|\mathbf{E}\|_{2,1} \quad (4.14)$$

which is solved by performing the following function to each column of $\mathbf{E}$, i.e., $\mathbf{E}_j$, as

$$\mathbf{E}_j^{(t+1)} \leftarrow \text{vect_soft_col}(\vartheta_j, \lambda) \quad (4.15)$$

where $\vartheta = \mathbf{Y} - \mathbf{V}_1^{(t)}$ and $\text{vect_soft_col}(\mathbf{t}, \xi)$ is a well-known function [119] employed to get column-wise sparsity. The solution of equation (4.15) is given as $\mathbf{t} \mapsto (\max\{\|\mathbf{t}\|_2 - \xi, 0\} / \max\{\|\mathbf{t}\|_2 - \xi, 0\} + \xi) \mathbf{t}$. For the variable $\mathbf{V}_1$, the reduced optimization problem yields

$$\mathbf{V}_1^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_1} \frac{1}{2} \|\mathbf{Y} - \mathbf{V}_1 - \mathbf{E}^{(t+1)}\|_F^2 + \frac{\mu}{2} \|\mathbf{DZ}^{(t+1)} - \mathbf{V}_1 - \mathbf{P}_1^{(t)}\|_F^2 \quad (4.16)$$

which is solved by the following

$$\mathbf{V}_1^{(t+1)} \leftarrow \frac{1}{1 + \mu} (\mathbf{Y} - \mathbf{E}^{(t+1)} + \mu (\mathbf{DZ}^{(t+1)} - \mathbf{P}_1^{(t)})) \quad (4.17)$$

The reduced optimization problem for $\mathbf{V}_2$ is given as

$$\mathbf{V}_2^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_2} \gamma \|\mathbf{C} \odot \mathbf{V}_2\|_{2,1} + \frac{\mu}{2} \|\mathbf{Z}^{(t+1)} - \mathbf{V}_2 - \mathbf{P}_2^{(t)}\|_F^2 \quad (4.18)$$

which is solved similar to the equations (4.14) and (4.15), respectively. The equation (4.18) can be computed for each row of $\mathbf{V}_2$, i.e., $\mathbf{V}_{2,u}$, as

$$\mathbf{V}_{2,u}^{(t+1)} \leftarrow \text{vect_soft_row_loc}(\psi_u, \gamma/\mu, \mathbf{C}) \quad (4.19)$$

where $\psi = \mathbf{Z}^{(t+1)} - \mathbf{P}_2^{(t)}$ and $\text{vect_soft_row_loc}(\mathbf{t}, \xi, \mathbf{s})$ ensures locality-preserved row-wise sparsity. The solution of this function is computed by $\mathbf{t} \mapsto (\max\{\|\mathbf{t}\|_2 - \xi * \|\mathbf{s}\|_2, 0\} / \max\{\|\mathbf{t}\|_2 - \xi * \|\mathbf{s}\|_2, 0\} + \xi * \|\mathbf{s}\|_2) \mathbf{t}$. Similar to the $\mathbf{V}_1$ and $\mathbf{V}_2$, the reduced optimization problem of $\mathbf{V}_3$ is given as

$$\mathbf{V}_3^{(t+1)} \leftarrow \arg \min_{\mathbf{V}_3} \tau \|\mathbf{V}_3\|_{\omega,*} + \frac{\mu}{2} \|\mathbf{Z}^{(t+1)} - \mathbf{V}_3 - \mathbf{P}_3^{(t)}\|_F^2 \quad (4.20)$$

which has a closed form solution [101] if and only if the weights ω_i are non-ascending order and non-negative. Then, this closed form solution can be calculated by

$$\mathbf{V}_3^{(t+1)} \leftarrow \text{SVT}_{\frac{\tau}{\mu} \omega}(\mathbf{Z}^{(t+1)} - \mathbf{P}_3^{(t)}) \quad (4.21)$$

Algorithm 2: ADMM for solving LCJS-WLR

Input: $\mathbf{Y}, \mathbf{D}, \mathbf{C}, \gamma, \tau, \lambda$

Initialize: $\mu = 10^{-1}, \text{maxiter} = 200, t = 0$

$$\mathbf{R} = (\mathbf{D}^T \mathbf{D} + 2\mathbf{I})^{-1}, \mathbf{Z}^{(0)} = \mathbf{R} \mathbf{D}^T \mathbf{Y}$$

$$\mathbf{V}_1^{(0)} = \mathbf{D} \mathbf{Z}^{(0)}, \mathbf{V}_2^{(0)} = \mathbf{V}_3^{(0)} = \mathbf{Z}^{(0)}$$

$$\mathbf{P}_1^{(0)} = \mathbf{P}_2^{(0)} = \mathbf{P}_3^{(0)} = \mathbf{0}$$

repeat

$$1. \mathbf{Z}^{(t+1)} \leftarrow \mathbf{R}[\mathbf{D}^T(\mathbf{V}_1^{(t)} + \mathbf{P}_1^{(t)}) + \mathbf{V}_2^{(t)} + \mathbf{P}_2^{(t)} + \mathbf{V}_3^{(t)} + \mathbf{P}_3^{(t)}]$$

$$2. \mathbf{E}^{(t+1)} \leftarrow \text{vect_soft_col}(\mathbf{Y} - \mathbf{V}_1^{(t)}, \lambda)$$

$$3. \mathbf{V}_1^{(t+1)} \leftarrow 1/(1 + \mu)(\mathbf{Y} - \mathbf{E}^{(t+1)} + \mu(\mathbf{D} \mathbf{Z}^{(t+1)} - \mathbf{P}_1^{(t)}))$$

$$4. \mathbf{V}_2^{(t+1)} \leftarrow \text{vect_soft_row_loc}(\mathbf{Z}^{(t+1)} - \mathbf{P}_2^{(t)}, \gamma/\mu, \mathbf{C})$$

$$5. \mathbf{V}_3^{(t+1)} \leftarrow \text{SVT}_{\frac{\tau}{\mu}\omega}(\mathbf{Z}^{(t+1)} - \mathbf{P}_3^{(t)})$$

$$6. \mathbf{P}_1^{(t+1)} \leftarrow \mathbf{P}_1^{(t)} - \mathbf{D} \mathbf{Z}^{(t+1)} + \mathbf{V}_1^{(t+1)}$$

$$7. \mathbf{P}_h^{(t+1)} \leftarrow \mathbf{P}_h^{(t)} - \mathbf{Z}^{(t+1)} + \mathbf{V}_h^{(t+1)}, h = 2, 3.$$

$$8. t \leftarrow t + 1$$

until *maxiter* is reached

Output: optimal solutions $\hat{\mathbf{Z}}$ and $\hat{\mathbf{E}}$

where SVT denotes the singular value thresholding function, and formulated as

$$\text{SVT}_{\zeta}(H) = U S_{\zeta}(\Sigma) V^T \quad (4.22)$$

where $U \Sigma V^T$ means the singular value decomposition of H and the well-known operator $S_{\varphi}(\nu) = \text{sgn}(\nu) \max(\|\nu\| - \varphi, 0)$ called as soft-thresholding operator. Eventually, the scaled Lagrangian multipliers are iterated as

$$\begin{aligned} \mathbf{P}_1^{(t+1)} &\leftarrow \mathbf{P}_1^{(t)} - \mathbf{D} \mathbf{Z}^{(t+1)} + \mathbf{V}_1^{(t+1)} \\ \mathbf{P}_h^{(t+1)} &\leftarrow \mathbf{P}_h^{(t)} - \mathbf{Z}^{(t+1)} + \mathbf{V}_h^{(t+1)}, h = 2, 3. \end{aligned} \quad (4.23)$$

After the iteration counter is reached to the pre-defined maximum iteration count, the optimizer provides the optimum solutions $\hat{\mathbf{Z}}$ and $\hat{\mathbf{E}}$, respectively. Then, the residual error vector is evaluated for each class p as follow

$$e_p(\mathbf{Y}) = \|\mathbf{Y} - \mathbf{D}_p \hat{\mathbf{Z}}_p - \hat{\mathbf{E}}\|_F \quad (4.24)$$

where $\mathbf{D}_p$ and $\hat{\mathbf{Z}}_p$ represent the sub-dictionary and the reconstructed joint-sparse coefficient matrix of the p th class, respectively. Finally, the label of superpixel is determined as the class with the least residual

$$\text{class}(\mathbf{Y}) = \min \left(e^p(\mathbf{Y}) \right) \quad \text{where } p = 1, \dots, P \quad (4.25)$$

A pseudocode of the proposed LCJS-WLR is presented in Algorithm 2.

4.3 Experimental Results

In order to confirm the effectiveness of the proposed LCJS-WLR, the IP1 dataset is used throughout the experiments. Each class of the dataset is divided by 10% and 90% as the train and test samples, respectively. However, the number of train samples, which is below 10 in a class, is completed to 10 due to the instability of the methods used to comparison. All of the experiments are reiterated 10 times and then the obtained results averaged. Different joint-sparse and low-rank based HSIC methods are implemented to compare the results of the LCJS-WLR, such as locality constraint low-rank representation (LCLRR) [120], JSRC [33], robust superpixel-level joint-sparse representation classifier (RSJSRC) [121] and low-rank and sparse representation classifier with a spectral consistency constraint and spatial extension (LRSR-SCCE) [53]. The optimal parameter settings are applied as in the original papers tuned by the authors of them. The regularization parameters of the LCLRR [120] are given as $\lambda = 0.35$ and $\gamma = 2.85$, respectively. The sparsity level and patch size of the JSRC are tuned as 30 and 7 [33, 121], respectively. The optimal parameters of the RSJSRC [121] are provided as number of superpixels $p = 700$, sparsity $K_0 = 50$ and sparse noise regularization $\lambda = 0.003$. The existing two regularizations of the LRSR-SCCE [53] are adjusted as $\lambda = 10$ and $\beta = 0.1$, then the patch size T for the later use of the spatial information is set as 11. As for the proposed LCJS-WLR, it has four parameters to be adjusted such as T , γ , τ and λ , respectively. To determine the optimal values of γ and τ , the remaining two parameters are fixed as $T = 5$ and $\lambda = 1e-2$, respectively. Then, different parameter intervals are chosen as $\gamma = \{1e-4, 5e-4, 1e-3, 5e-3, 1e-2, 5e-2, 1e-1\}$ and $\tau = \{1e-4, 5e-4, 1e-3, 5e-3, 1e-2\}$, respectively. Figure 4.1 demonstrates the OA results of the LCJS-WLR with respect to the two parameters γ and τ . It is seen that $\gamma = 1e-2$ and $\tau = 1e-4$ provide the highest OA value.

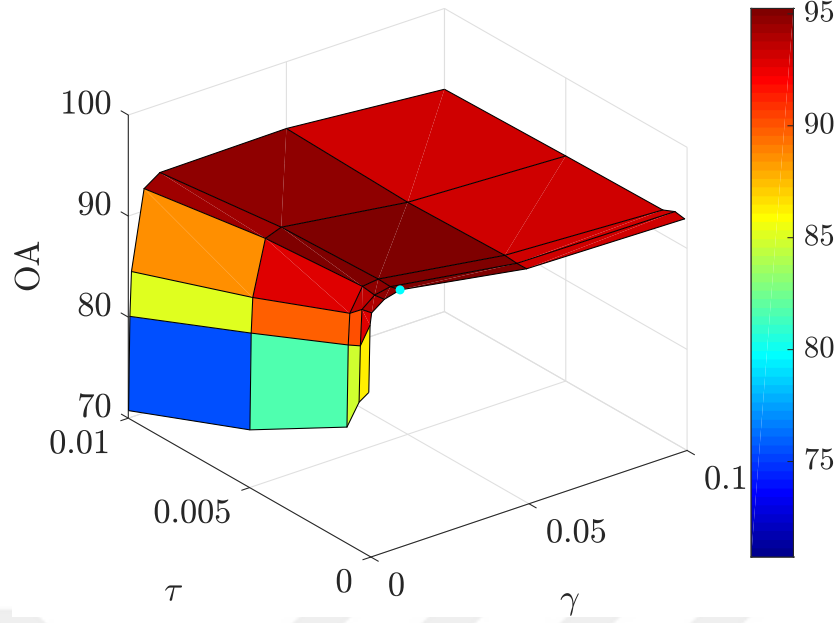


Figure 4.1 IP1 dataset: Effect of different γ and τ parameters on OA result of the proposed LCJS-WLR (The highest result is indicates with blue dot).

After parameter tuning of the methods, a fair comparison is applied to observe the efficacy of the proposed LCJS-WLR with respect to the similar recent methods. The obtained results such as OA, AA, κ , class-specific accuracies and computation times, are given in Table 4.1. Among them, the LCLRR obtains the worst results due to the following reasons; it solves the low-rank globally which increase the computation time, and it only uses the spectral information which decrease the accuracy results. The JSRC outperforms the others in terms of computation time because it does not solve any low-rank-based optimization and also does not use any additional regularization. Also, the accuracy results of the JSRC are higher than the results of the LCLRR. The improved version of the JSRC, called RSJSRC, few surpasses the results of the JSRC due to the use of superpixel segmentation, and small sparse noise elimination term into the optimization. The OA and AA of the RSJSRC are approximately 1% higher than the results of the JSRC. The method LRSR-SCCE which uses locality-constraint global low-rank and sparse terms, sparse noise removal term in the optimization and also additional spatial extension obtains better accuracy results than the RSJSRC out of running time.

It is clearly observed that the OA and AA of the proposed LCJS-WLR exceed the OAs and AAs of the LCLRR, JSRC, RSJSRC and LRSR-SCCE by approximately 16%,

Table 4.1 IP1 dataset: Classification results of the methods LCLRR, JSRC, RSJSRC, LRSRC-SCCE and proposed LCJS-WLR
 *The best results are bolded and the second best results are underlined.

Class	Train	Test	LCLRR	JSRC	RSJSRC	LRSRC-SCCE	LCJS-WLR
1	10	44	90.45±3.37	93.64±2.49	89.09±6.31	95.00 ±2.49	87.73±8.29
2	144	1290	70.34±1.74	93.12±1.28	90.81±2.16	92.90±2.34	93.22 ±0.70
3	84	750	40.21±4.22	85.81±3.62	89.09 ±3.97	86.19±2.87	86.64±2.95
4	24	210	48.00±7.63	90.67±3.34	89.71±7.37	87.62±2.69	94.57 ±2.83
5	50	447	90.74±3.68	90.78±2.70	94.27 ±2.09	93.38±1.03	94.27 ±1.35
6	75	672	93.81±1.33	98.90±0.69	97.14±2.19	97.86±0.52	99.35 ±0.08
7	10	16	72.50±12.96	93.75±8.84	93.75±6.25	96.25±3.42	100 ±0.00
8	49	440	99.82±0.10	99.64±0.12	99.77±0.00	99.91 ±0.20	99.77±0.00
9	10	10	20.00±44.72	64.00±18.17	82.00±21.68	100 ±0.00	100 ±0.00
10	97	871	73.98±4.35	84.23±2.28	93.62 ±2.56	91.94±3.32	93.36±3.69
11	247	2221	88.51±1.16	96.69±0.79	95.64±1.42	96.66±0.47	97.65 ±0.57
12	62	552	58.19±4.94	85.04±3.80	86.99±5.46	93.19 ±1.15	90.18±3.90
13	22	190	99.05±0.94	99.37±0.69	99.47±0.00	99.37±0.24	99.58 ±0.24
14	130	1164	98.09±1.61	99.57±0.46	99.19±1.28	99.67 ±0.33	98.93±1.49
15	38	342	65.26±4.32	93.22±6.00	86.73±8.46	79.24±1.97	94.15 ±3.60
16	10	85	95.53±3.37	95.06±3.94	93.65±7.83	97.88 ±0.98	95.29±7.89
OA	–	–	79.58±0.50	93.56±0.11	93.97±1.08	94.32±0.37	95.31 ±0.22
AA	–	–	75.28±2.00	91.47±1.25	92.56±1.06	94.19±0.41	95.29 ±0.63
κ	–	–	76.57±0.57	92.63±0.12	93.12±1.23	93.52±0.42	94.65 ±0.25
Time (sec)	–	–	614.61	50.76	91.36	296.18	301.11

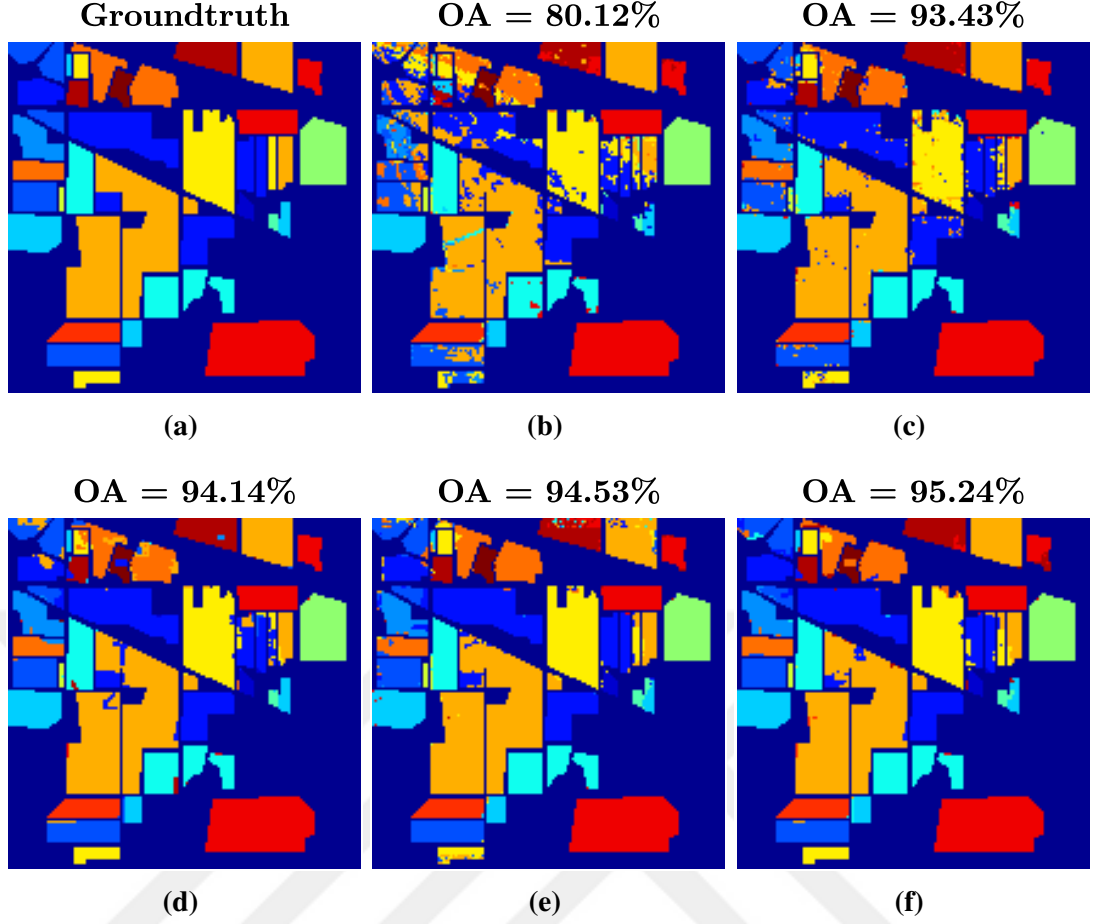


Figure 4.2 IP1 dataset: (a) Ground-truth; classification maps and OA results of the methods (b) LCLRR, (c) JSRC, (d) RSJSRC, (e) LRSRC-SCCE and (f) proposed LCJS-WLR.

2%, 1.5% and 1%, and then by almost 20%, 4%, 3% and 1%, respectively. As for the class-based accuracies, the LCJS-WLR usually provides the best or second best results. Compared to the LRSR-SCCE, the LCJS-WLR solves the data locally instead of globally. The spatial structure is incorporated during the optimization stage, however, LRSR-SCCE involves the spatial information after the optimization by averaging the spectra of the neighbors around each test sample. It can be inferred from the obtained results that; (1) the use of spatial information increases the all classification results, (2) low-rank-based methods increase the computation time due to the singular value decomposition, (3) the usage of locality-constraint positively affects the results, (4) incorporation of the spatial information in the optimization stage directly improves the performance of the classifier, and (5) the use of sparse noise removal regularization can few increase the accuracy results.

Figure 4.2 shows the classification maps of the methods to analyze the visual outputs of the classifiers. The effect of not using spatial information can be observed as the map of LCLRR. However, the other classification maps seem more smoothed owing to the use of spatial structure. It is seen that the proposed LCJS-WLR provides the classification map closest to the ground-truth map. It also verifies that the use of locality-constraint joint-sparse, weighted low-rank and superpixels together in the optimization problem improve the performance compared to the similar studies in the literature.

4.4 Summary

In this chapter, the proposed spatial-spectral HSIC method called LCJS-WLR is explained and analyzed in detail. The local regions obtained by superpixels are classified using an optimization problem consisting locality-constraint joint-sparse, weighted low-rank and noise removal terms. The extensive experimental results verify that the proposed LCJS-WLR classifier outperforms the existing similar classifiers due to the efficient usage of the spatial information, locality information of joint-sparse structure, weighted low-rank expression and possible small sparse noise removal term.

CHAPTER 5

GUIDED FILTER OF RANDOM PATCHES NETWORK AND RELAXED COLLABORATIVE REPRESENTATION BASED HYPERSPECTRAL IMAGE CLASSIFICATION

5.1 Introduction

In this chapter, the proposed GRR classifier is explained and analyzed in detail. To obtain a HSIC method providing high performance, the use of only the original features is not enough sometimes. Even though there are informative features, their correct classification is also important. The proposed GRR seeks a solution to extract more characteristics features, and classify these features efficiently. To do this, a compact HSIC method using the GF of RpNet and superpixel-guided RCR-based classification is proposed. Both the shallow and deep features presented in the HSI are extracted by the RpNet without any network training. In order to protect the border details of the classes, the raw HSI and the extracted RpNet features are filtered by the GF. The obtained three-different feature sets such as the extracted RpNet features, GF of the RpNet features and GF of the raw HSI features, are combined with the raw HSI features to maintain the original textures, respectively. After these concatenations, each feature set has quite high-dimensional space, which is then handled by the LDA to maximize the interclass variances and to reduce the dimension. Since the capacity of LDA is completely related to the available train samples, rarely found in HSIC task, superpixel-guided pseudo-sample production is implemented. After LDA processing, the dimension-reduced feature sets are classified by the SVM, and then the obtained OA results are converted to the distinct weights to be used by the RCR. Hence, the superpixel-based RCR is implemented to classify these feature sets efficiently due to the incorporation of the similarity and diversity of feature sets into the RCR classifier. The validity of the proposed GRR is tested on the IP2, PU and WHHH datasets, respectively. The comprehensive experimental results are represented that the GRR outperforms the recent RpNet and RCR-based HSIC methods.

5.2 Problem Formulation

The stages of GRR can be summarized as: (1) extraction of the shallow and deep features by the RPNet, (2) filtering the raw HSI and RPNet features by the GF, (3) concatenation of the obtained different feature sets with the original HSI, (4) generation of the pseudo-samples for the LDA implementation, (5) processing the high-dimensional feature spaces by the, (6) pre-classification of the feature sets by the SVM to obtain RCR weights and (7) superpixel-guided RCR-based classification. The following subsections describe and formulate the steps of GRR in detail.

5.2.1 RPNet Feature Extraction

While the traditional HSIC methods using deep networks [122–124] needs quite a lot train samples for training, RPNet does not need any train samples to extract the deep features. It operates a series of processes to obtain the embedded features in the HSI, such as reducing the dimension, selection of the convolution kernels, convolution of the HSI with these kernels and activation of the features, respectively. The flowchart of RPNet can be summarized as follows. First, the PCA is utilized to obtain the first-few representative principal components. Second, a whitening function [125] is applied on these principal components to have spectral bands with similar variance and reduced correlation. Thirdly, different fixed-size convolution kernels are formed in the whitened data by randomly selected t pixels. Then, t different feature maps are obtained as a result of convolving the whitened data by the constructed kernels. In the last step, these obtained feature maps are given to the activation layer named rectified linear units (ReLU). The stages described so far are repeated along the pre-determined parameter of the layer number.

Let $\mathcal{H} \in \mathbb{R}^{r \times c \times b}$ be the 3D HSI where r and c denotes the spatial dimension and b represents the number of spectral channels. Suppose that $\mathcal{H}_P \in \mathbb{R}^{r \times c \times h}$ indicates the first h principal components of $\mathcal{H}$ calculated by PCA and, the output of whitening function on $\mathcal{H}_P$ is given as $\mathcal{H}_W \in \mathbb{R}^{r \times c \times h}$. The convolution process on $\mathcal{H}_W$, which is performed by the convolution kernels $\mathcal{C}_{1,2,\dots,t} \in \mathbb{R}^{w \times w \times h}$ with a fixed size of $w \times w$, is given as

$$\mathbf{N}_j = \sum_{i=1}^h \mathcal{H}_W^i * \mathcal{C}_j^i \quad ; \quad j = 1, 2, \dots, t \quad (5.1)$$

where $*$ indicates the 2D convolution operator, $\mathbf{N}_j \in \mathbb{R}^{r \times c}$ denotes the j th feature matrix, $\mathcal{H}_W^i \in \mathbb{R}^{r \times c}$ is the i th channel of $\mathcal{H}_W$ and $\mathcal{C}_j^i$ represents the j th patch of channel i . Then, the obtained feature matrices $\mathbf{N}_j$ are stacked into the $\mathcal{N} \in \mathbb{R}^{r \times c \times t}$ for each layer. Suppose that ReLU activation function is applied on $\mathcal{N}$, and then the outgoing features $\mathcal{U}^1 \in \mathbb{R}^{r \times c \times t}$ of the first layer are obtained as the following

$$\mathcal{U}^1 = \max(0, \mathcal{N} - \mathcal{M}) \quad (5.2)$$

where $\mathcal{M} \in \mathbb{R}^{r \times c \times t}$ denotes a 3D matrix computed by the average of $\mathcal{N}$ and then replicated t times in the third dimension as

$$\mathcal{M}(:, :, j) = \frac{1}{t} \sum_{j=1}^t \mathcal{N}(:, :, j), \quad ; \quad j = 1, 2, \dots, t \quad (5.3)$$

The initial layer of RpNet produces the feature $\mathcal{U}^1$ containing the shallow features. In order to obtain the deeper features, RpNet passes the resulting features of the prior layer as entry of the subsequent layer. Suppose that the feature $\mathcal{U}^{(\ell-1)}$ represents the output of layer $(\ell - 1)$, and it is given as the input of layer ℓ . Hereafter, all stages are repeated same as the previous layer except the input until the pre-determined layer number l . For example, if input and output of the 1st layer are $\mathcal{H}$ and $\mathcal{U}^1$, respectively, then the input and output of the 2nd layer would be $\mathcal{U}^1$ and $\mathcal{U}^2$, which will be the input of the 3rd layer. In this way, both the low and high level feature representations $\mathcal{U}^{1,2,\dots,l}$ are obtained by the layers. As the number of layers increases, the features become deeper because each subsequent layer utilizes the output of the previous one. After extracting such features, they are concatenated into the $\mathcal{U} = \{\mathcal{U}^1, \mathcal{U}^2, \dots, \mathcal{U}^l\} \in \mathbb{R}^{r \times c \times tl}$ to obtain a feature set of the RpNet.

5.2.2 Guided Filter

The GF is used to keep and extract the border information of the classes. The working mechanism of GF is to project the textural information of a given guidance image to the filtered image. The choice of guidance image can be gray or color image. Since a color image can hold more structural information rather than a gray image, the first-three principal components of PCA is generally utilized as the base image in HSIC studies [43, 40]. In such a situation where color image is used as the guidance image, it is necessary to implement GF to each spectral channel of the color image separately [102], which means that more information is transferred to the input image.

As mentioned above, a local linear connection among the output and guidance image [102] exists in a patch ω_z which is centred at the pixel z . Such a linear projection utilizing color image $\mathcal{G} \in \mathbb{R}^{r \times c \times 3}$ in the patch ω_z can be written as follows

$$q_u = \alpha_z^\top \mathbf{g}_u + \beta_z \quad ; \quad \forall u \in \omega_z \quad (5.4)$$

where the scalar $q_u \in \mathbb{R}^{1 \times 1}$ denotes the filter output at the pixel u in ω_z , $\alpha_z \in \mathbb{R}^{3 \times 1}$ and $\beta_z \in \mathbb{R}^{1 \times 1}$ represent the filter coefficients, $\mathbf{g}_u \in \mathbb{R}^{3 \times 1}$ is the color vector of $\mathcal{G}$ at pixel u , and the operator $(\cdot)^\top$ indicates transpose of the vector. For the purpose of acquisition of the coefficients α_z and β_z , the following energy function is created as the following

$$E(\alpha_z, \beta_z) = \sum_{u \in \omega_z} \left((\alpha_z^\top \mathbf{g}_u + \beta_z - p_u)^2 + \epsilon \alpha_z^2 \right) \quad (5.5)$$

where the parameter ϵ regulates the degree of smoothing, and $p_u \in \mathbb{R}^{1 \times 1}$ denotes the pixel intensity of the input image $\mathbf{P} \in \mathbb{R}^{r \times c}$ presented at the pixel u in the patch ω_z . After the optimum values α_z and β_z are determined, the filter output q_u is simply calculated [102] by equation (5.4). If the stages of GF are evaluated for all pixels similar to the pixel z , the filtered image $\tilde{\mathbf{P}} \in \mathbb{R}^{r \times c}$ in accordance with the input image $\mathbf{P}$ and guidance image $\mathcal{G}$, is attained.

As mentioned before, the proposed GRR separately implements the GF process on the original HSI and extracted RpNet features. Let $\mathcal{G}^{\mathcal{H}} \in \mathbb{R}^{r \times c \times 3}$ and $\mathcal{G}^{\mathcal{U}} \in \mathbb{R}^{r \times c \times 3}$ be the first-three principal components of PCA computed on the $\mathcal{H}$ and $\mathcal{U}$, respectively. Suppose that these components $\mathcal{G}^{\mathcal{H}}$ and $\mathcal{G}^{\mathcal{U}}$ are used as the guidance images of original HSI and RpNet features, respectively. Since the equations (5.4) and (5.5) are suitable for the calculation over a single-band image, each spectral band of the $\mathcal{H}$ and $\mathcal{U}$ should be separately processed by the GF using the guidance images $\mathcal{G}^{\mathcal{H}}$ and $\mathcal{G}^{\mathcal{U}}$, respectively. Then, the filtered results such as $\tilde{\mathcal{H}} \in \mathbb{R}^{r \times c \times b}$ and $\tilde{\mathcal{U}} \in \mathbb{R}^{r \times c \times tl}$ of the $\mathcal{H}$ and $\mathcal{U}$ are provided.

Followed by the GF implementation, different feature sets of the GRR are arranged as

$$\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}} = \{\mathcal{H}, \tilde{\mathcal{H}}\} \in \mathbb{R}^{r \times c \times 2b} \quad (5.6a)$$

$$\mathcal{F}_{\mathcal{H}\mathcal{U}} = \{\mathcal{H}, \mathcal{U}\} \in \mathbb{R}^{r \times c \times (b+tl)} \quad (5.6b)$$

$$\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}} = \{\mathcal{H}, \tilde{\mathcal{U}}\} \in \mathbb{R}^{r \times c \times (b+tl)} \quad (5.6c)$$

where the feature sets $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathcal{F}_{\mathcal{H}\mathcal{U}}$ and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$ are acquired by concatenation in the third dimension of $\mathcal{H}$ with the $\tilde{\mathcal{H}}$, $\mathcal{U}$ and $\tilde{\mathcal{U}}$, respectively. The incorporation of raw HSI $\mathcal{H}$ into the feature sets $\tilde{\mathcal{H}}$, $\mathcal{U}$ and $\tilde{\mathcal{U}}$ allows to utilize the original structure of HSI for the further operations.

5.2.3 Pseudo-Sample Generation

Due to the high-dimensionality of the feature spaces $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathcal{F}_{\mathcal{H}\mathcal{U}}$ and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$, the LDA will be implemented for the purpose of reducing the high-dimensionality. In addition, it also improves the inter-class discrimination and reduces the computational load. However, the capability of LDA entirely depends on the existing train samples because it is a supervised algorithm. Since the HSIC methods endured by the available only few samples [1], it is required to produce pseudo-samples to increase the efficacy of the LDA. In order to overcome this issue, superpixel-guided pseudo-samples are produced simply but effectively. Firstly, the HSI $\mathcal{H}$ is segmented by the ERS algorithm to obtain homogeneous areas. The first principal component of PCA is utilized as the reference image for the ERS-based segmentation. To find the appropriate number of superpixels K , the same technique used in [40, 96] is utilized where $K = rc/T^2$, and T adjusts the approximate number of samples in a superpixel and rc denotes the spatial size of the HSI.

Suppose that the dictionary is formed by n train samples, i.e., $\mathbf{d}_{1,\dots,n} \in \mathbb{R}^{b \times 1}$, as $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_n] \in \mathbb{R}^{b \times n}$ including P distinct classes, and each class contains n_p samples inside, then $n = \sum_{p=1}^P n_p$. Let $S = (s_1, s_2, \dots, s_K)$ be the group of segmented regions, and $L(\cdot)$ be an operator specifying the class of a superpixel denoted as $s_{k=1,2,\dots,K}$. Then, the following operation is performed for each superpixel $s_{k=1,2,\dots,K}$ as

$$L(s_k) = \begin{cases} \text{mode}(L\{\mathbf{d}_{s_k}\}), & \text{if any } \mathbf{d}_{1,\dots,n} \in s_k \\ \emptyset, & \text{otherwise} \end{cases} \quad (5.7)$$

where $L(s_k)$ indicates to the label of s_k , and $\{\mathbf{d}_{s_k}\}$ represents train samples in s_k . The empty set $\emptyset$ indicates that the region s_k encloses no train samples, therefore, no label is dedicated. The function $\text{mode}(L\{\mathbf{d}_{s_k}\})$ considers the train labels found in s_k , then, applies the mathematical *mode* operator which detects the most common label presented in s_k , if exists. To sum up, equation (5.7) is used to enforce the following

two conditions: (Case 1) If a segmented region comprises of any train samples, then the most common label is given to that region. (Case 2) If a segmented region includes no train samples, then no label is assigned to that superpixel.

Due to the proposed pseudo-sample production, the amount of available train samples is improved from n to η , i.e., $\eta \gg n$. Hence, the symbol and size of current dictionary $\mathbf{D} \in \mathbb{R}^{b \times n}$ is updated as $\hat{\mathbf{D}} \in \mathbb{R}^{b \times \eta}$ for the use of LDA process.

5.2.4 Linear Discriminant Analysis

As mentioned above, the LDA is a supervised method which reduces the dimension of a given data, and also increases the inter-class variance at the same time. According to [126], the performance of LDA is seriously affected by the lack of train samples. Moreover, it cannot be applied if the number of train samples is lower than the size of the data, which causes singularity. Due to the created pseudo-samples and expanded dictionary, these drawbacks of the LDA is highly eliminated.

Suppose that the expanded dictionary including $\eta = \sum_{p=1}^P \eta_p$ samples is given by $\hat{\mathbf{D}} = [\hat{\mathbf{d}}_1, \hat{\mathbf{d}}_2, \dots, \hat{\mathbf{d}}_\eta] \in \mathbb{R}^{b \times \eta}$. If each class p contains η_p samples, then the average of each class and all samples denoted as $\mathbf{m}_p \in \mathbb{R}^{b \times 1}$ and $\mathbf{m} \in \mathbb{R}^{b \times 1}$, respectively, can be computed as follows

$$\mathbf{m}_p = \frac{1}{\eta_p} \sum_{\iota=1}^{\eta_p} \hat{\mathbf{d}}_{p_\iota} \quad (5.8a)$$

$$\mathbf{m} = \frac{1}{P} \sum_{p=1}^P \mathbf{m}_p \quad (5.8b)$$

where $\hat{\mathbf{d}}_{p_\iota}$ indicates the ι th sample in class p . Then, the intra-class and inter-class scatter matrices $\mathbf{T}_W \in \mathbb{R}^{b \times b}$ and $\mathbf{T}_B \in \mathbb{R}^{b \times b}$, respectively, are computed by

$$\mathbf{T}_W = \sum_{p=1}^P \sum_{\iota=1}^{\eta_p} (\hat{\mathbf{d}}_{p_\iota} - \mathbf{m}_p)(\hat{\mathbf{d}}_{p_\iota} - \mathbf{m}_p)^\top \quad (5.9a)$$

$$\mathbf{T}_B = \sum_{p=1}^P \eta_p (\mathbf{m}_p - \mathbf{m})(\mathbf{m}_p - \mathbf{m})^\top \quad (5.9b)$$

For the purpose of improving inter-class and reducing intra-class variances, the following equation has to be maximized with respect to the projection matrix $\mathbf{V} \in \mathbb{R}^{b \times b}$

$$\max_{\mathbf{V}} \left\{ \frac{\mathbf{V}^\top \mathbf{T}_B \mathbf{V}}{\mathbf{V}^\top \mathbf{T}_W \mathbf{V}} \right\} \quad (5.10)$$

where $\mathbf{V}$ is computed by solving a generalized eigenvalue problem, given as

$$\mathbf{T}_B \mathbf{V} = \mathbf{\Phi} \mathbf{T}_W \mathbf{V} \quad (5.11)$$

where the colons of $\mathbf{V}$ contain the eigenvectors corresponding to the eigenvalues which are the diagonal entries of $\mathbf{\Phi}$. After calculating $\mathbf{V}$ and $\mathbf{\Phi}$, the eigenvalues are sorted in descending sequence together with their associated eigenvectors. After this arrangement, the first $P - 1$ colons of sorted $\mathbf{V}$ are chosen since the matrix $\mathbf{T}_B$ has a rank of $P - 1$, and also the first $P - 1$ eigenvalues of sorted $\mathbf{\Phi}$ are nonzero. Consequently, the first $P - 1$ colons of sorted $\mathbf{V}$, i.e., $\mathbf{V}_d \in \mathbb{R}^{b \times (P-1)}$, is utilized as the projection matrix for reducing the dimension.

Assume that the concatenated feature sets existed in sub-equations (5.6a–5.6c), given as $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{r \times c \times 2b}$, $\mathcal{F}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{r \times c \times (b+tl)}$ and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{r \times c \times (b+tl)}$, are transformed to the 2D features as $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{2b \times m}$, $\mathbf{F}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(b+tl) \times m}$ and $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(b+tl) \times m}$, respectively. The expanded dictionaries associated with these 2D feature sets can be represented as $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{2b \times \eta}$, $\hat{\mathbf{D}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(b+tl) \times \eta}$ and $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(b+tl) \times \eta}$, i.e., ($\eta < m$), respectively. In order to provide the projection matrices $\mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{H}}}} \in \mathbb{R}^{2b \times (P-1)}$, $\mathbf{V}_{d_{\mathcal{H}\mathcal{U}}} \in \mathbb{R}^{(b+tl) \times (P-1)}$ and $\mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{U}}}} \in \mathbb{R}^{(b+tl) \times (P-1)}$ corresponding to these enlarged dictionaries $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\hat{\mathbf{D}}_{\mathcal{H}\mathcal{U}}$ and $\hat{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}}$, all the equations between (5.8a) and (5.11) are calculated in order. Afterwards, the transformations are computed using these projection matrices and 2D feature sets as the following

$$\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}} = \mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{H}}}}^\top \mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}} \quad (5.12a)$$

$$\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}} = \mathbf{V}_{d_{\mathcal{H}\mathcal{U}}}^\top \mathbf{F}_{\mathcal{H}\mathcal{U}} \quad (5.12b)$$

$$\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}} = \mathbf{V}_{d_{\mathcal{H}\tilde{\mathcal{U}}}}^\top \mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}} \quad (5.12c)$$

where $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{(P-1) \times m}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(P-1) \times m}$ and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(P-1) \times m}$ represent the dimension-reduced feature sets associated with $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\mathbf{F}_{\mathcal{H}\mathcal{U}}$ and $\mathbf{F}_{\mathcal{H}\tilde{\mathcal{U}}}$, respectively. Since the next step is the classification of samples, each column of these feature sets in the sub-equations (5.12a–5.12c) are normalized in the range of [0,1].

5.2.5 Relaxed Collaborative Representation

The RCR is based on the idea that in order to stress the diversity of feature sets, the collaborative representation vectors/matrices of the features must demonstrate both

similarities and differences. The RCR [75] introduced for the first time, is utilized for pixel-based pattern classification, therefore, it requires quite a long time for HSIC due to the presence of many samples. In order to alleviate this high computational load, superpixel-guided RCR is utilized, where the same segmentation map obtained in the pseudo-sample generation process is used. The use of superpixels enables RCR-based classifier to reduce the running time and to benefit from the spatial structure at the same time. Assume that v denotes an extracted feature set in a total of ϑ distinct feature sets where $v = 1, 2, \dots, \vartheta$. In the proposed GRR, each $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}$ and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$ represents a feature set, and their associated dictionaries are provided as $\check{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{H}}} \in \mathbb{R}^{(P-1) \times n}$, $\check{\mathbf{D}}_{\mathcal{H}\mathcal{U}} \in \mathbb{R}^{(P-1) \times n}$ and $\check{\mathbf{D}}_{\mathcal{H}\tilde{\mathcal{U}}} \in \mathbb{R}^{(P-1) \times n}$, respectively.

Suppose that the v th feature set in the k th superpixel contains the samples $\mathbf{Y}_k^v$ to be classified, the dictionary $\mathbf{A}^v$ and its associated collaborative coefficient matrix $\mathbf{X}_k^v$. So, the minimization problem of RCR is formulated as the following

$$\begin{aligned} \arg \min_{\{\mathbf{X}_k^v\}, \{\psi^v\}} \sum_{v=1}^{\vartheta} \left(\|\mathbf{Y}_k^v - \mathbf{A}^v \mathbf{X}_k^v\|_F^2 + \lambda \|\mathbf{X}_k^v\|_F^2 \right. \\ \left. + \tau \psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2 \right) \quad s.t. \text{ prior } \{\psi^v\}_{v=1,2,\dots,\vartheta} \end{aligned} \quad (5.13)$$

where λ and τ are the regularization constants. The weight of feature set v and the average of entire coefficient matrices $\{\mathbf{X}_k^v\}_{v=1,2,\dots,\vartheta}$ of feature set v in the k th superpixel, are represented as ψ^v and $\bar{\mathbf{X}}_k^v = \sum_{v=1}^{\vartheta} \mathbf{X}_k^v / \vartheta$, respectively. The constraint $\psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2$ guarantees that the representation coefficients $\{\mathbf{X}_k^v\}$ are similar since the equation (5.13) forces each coefficient matrix $\mathbf{X}_k^v$ to be minimized as close to averaged coefficients $\bar{\mathbf{X}}_k^v$. Besides, the impact of each feature set is incorporated into the optimization problem in equation (5.13) by assigning unique weight ψ^v to the each feature. It is stated in [75] that, the choice of *prior* $\{\psi^v\}$ in equation (5.13) rely on three different situations like strong, moderate and weak priorities. The RCR classifier directly solves equation (5.13) for the moderate and weak priorities by optimizing both $\{\mathbf{X}_k^v\}$ and $\{\psi^v\}$ which requires further computations. In the case where strong prior is used, the optimization problem in (5.13) just requires minimization of $\{\mathbf{X}_k^v\}$ because the strong prior stands for providing the weight of each feature set before classification. As described, the use of strong prior, i.e., previously known weights, reduces the computation time. In order to give proper weights to the feature sets, these are pre-classified with SVM followed by the superpixel-based RCR classifier. Suppose

that the feature sets $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{L}}}$, $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}$ and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$ are classified by the SVM, and each feature set v attains an OA results indicated as OA_v . Then, the obtained OAs are utilized to calculate the feature weights ψ^v as the following

$$\{\psi^v\}_{v=1,2,\dots,\vartheta} = \frac{\text{OA}_v}{\sum_{v=1}^{\vartheta} \text{OA}_v} \quad (5.14)$$

The above equation states that the weight ψ^v of a feature set v is determined as the ratio of its OA value OA_v divided by summation of OAs of all feature sets $\sum_{v=1}^{\vartheta} \text{OA}_v$. Additionally, equation (5.14) provides that total of the weights is equal to one. After this SVM-based weight assignment, the objective function in equation (5.13) can be updated as minimizing just $\{\mathbf{X}_k^v\}$ due to the previously known weights. Hence, equation (5.13) can be rearranged as

$$\arg \min_{\{\mathbf{X}_k^v\}} \sum_{v=1}^{\vartheta} \left(\|\mathbf{Y}_k^v - \mathbf{A}^v \mathbf{X}_k^v\|_F^2 + \lambda \|\mathbf{X}_k^v\|_F^2 + \tau \psi^v \|\mathbf{X}_k^v - \bar{\mathbf{X}}_k^v\|_F^2 \right) \quad (5.15)$$

which possesses a closed-form solution [75, 78] for $v = 1, 2, \dots, \vartheta$ given by

$$\mathbf{X}_k^v = \mathbf{X}_k^{0,v} + \tau \frac{\psi^v}{\sum_{\gamma=1}^{\vartheta} \psi^\gamma} \mathbf{B}_k^v \mathbf{Q}_k \sum_{\gamma=1}^{\vartheta} \psi^\gamma \mathbf{X}_k^{0,\gamma} \quad (5.16)$$

$$\bar{\mathbf{X}}_k^v = \sum_{v=1}^{\vartheta} \psi^v \mathbf{X}_k^v / \sum_{v=1}^{\vartheta} \psi^v \quad (5.17)$$

where the terms are given as $\mathbf{B}_k^v = ((\mathbf{A}^v)^\top \mathbf{A}^v + \mathbf{I}(\lambda + \tau \psi^v))^{-1}$, $\mathbf{X}_k^{0,v} = \mathbf{B}_k^v (\mathbf{A}^v)^\top \mathbf{Y}_k^v$, $\mathbf{Q}_k = (\mathbf{I} - \sum_{\gamma=1}^{\vartheta} \bar{\psi}^\gamma \mathbf{B}_k^\gamma)^{-1}$, $\bar{\psi}^\gamma = \tau (\psi^\gamma)^2 / \sum_{v=1}^{\vartheta} \psi^v$ and $\mathbf{I} \in \mathbb{R}^{n \times n}$ denotes an identity matrix. After obtaining the optimum solutions of $\{\mathbf{X}_k^v\}$, the residual error value e of the p th class is computed as

$$e_p = \sum_{v=1}^{\vartheta} \psi^v \|\mathbf{Y}_k^v - \mathbf{A}_p^v \mathbf{X}_{k_p}^v\|_F^2 \quad (5.18)$$

where $\mathbf{A}_p^v$ denotes the dictionary of the p th class of v th feature set. Similarly, the term $\mathbf{X}_{k_p}^v$ indicates the coefficient matrix of the p th class of v th feature presented in the k th superpixel. Finally, the class label of $\mathbf{Y}_k^v$ representing the samples of v th feature set inside k th superpixel, is identified as follows

$$L(\mathbf{Y}_k^v) = \arg \min_{p=1,2,\dots,P} e_p \quad (5.19)$$

where $L(\mathbf{Y}_k^v)$ denotes the assigned label of $\mathbf{Y}_k^v$. After evaluating the equations between (5.13) and (5.19) for all segmented regions, the classification map of proposed GRR is acquired. A comprehensive flowchart of the GRR supported by visuals is given in Figure 5.1. In addition, a detailed pseudocode of the GRR is provided in Algorithm 3.

Algorithm 3: Pseudocode of the proposed GRR

Input: Hyperspectral image $\mathcal{H}$; RpNet parameters t, h, w and l

Supapixel segmentation parameter T ; RCR parameters λ and τ

S1: Extract the RpNet features

for each layer ℓ up to l

- 1) Apply PCA to $\mathcal{H}$ and obtain $\mathcal{H}_P$ with first h PCs.
- 2) Whiten the data $\mathcal{H}_P$ and obtain $\mathcal{H}_W$.
- 3) Extract t convolution kernels $\mathcal{C}_{1,2,\dots,t}$ from the $\mathcal{H}_W$.
- 4) Convolve the $\mathcal{H}_W$ with each of $\mathcal{C}_{1,2,\dots,t}$ by (5.1) and then concatenate the features into the $\mathcal{N}$.
- 5) Obtain the feature $\mathcal{U}^\ell$ of the corresponding layer after applying the ReLU activation function to $\mathcal{N}$ by (5.2) and (5.3).
- 6) If $\ell < l$, repeat the process by changing $\mathcal{H}$ with $\mathcal{U}^\ell$.

end

S2: Stack the features into $\mathcal{U} = \{\mathcal{U}^1, \mathcal{U}^2, \dots, \mathcal{U}^l\}$.

S3: Apply GF on $\mathcal{U}$ and $\mathcal{H}$ by (5.4) and (5.5), then obtain $\tilde{\mathcal{U}}$ and $\tilde{\mathcal{H}}$.

S4: Stack the extracted features $\mathcal{U}, \tilde{\mathcal{U}}$ and $\tilde{\mathcal{H}}$ with $\mathcal{H}$ separately and construct the feature sets $\mathcal{F}_{\mathcal{H}\mathcal{U}}, \mathcal{F}_{\mathcal{H}\tilde{\mathcal{U}}}$ and $\mathcal{F}_{\mathcal{H}\tilde{\mathcal{H}}}$.

S5: Segment the $\mathcal{H}$ to many superpixels with ERS and generate pseudo-samples by (5.7).

S6: Apply the LDA on the pseudo-samples from (5.8a) to (5.11) and evaluate the dimension reduced feature sets $\check{\mathbf{F}}_{\mathcal{H}\mathcal{U}}, \check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{U}}}$ and $\check{\mathbf{F}}_{\mathcal{H}\tilde{\mathcal{H}}}$ from (5.12a) to (5.12c).

S7: Preclassify the features by SVM to determine convenient feature weights to each future.

S8: Supapixel-guided RCR classification

for each superpixel region k up to K

- 1) Solve the RCR from (5.13) to (5.17) and find $\{\mathbf{X}_k^v\}$.
- 2) Evaluate the residuals by (5.18).
- 3) Assign a label to superpixel by (5.19).

end

Output: Predicted class labels of the $\mathcal{H}$.

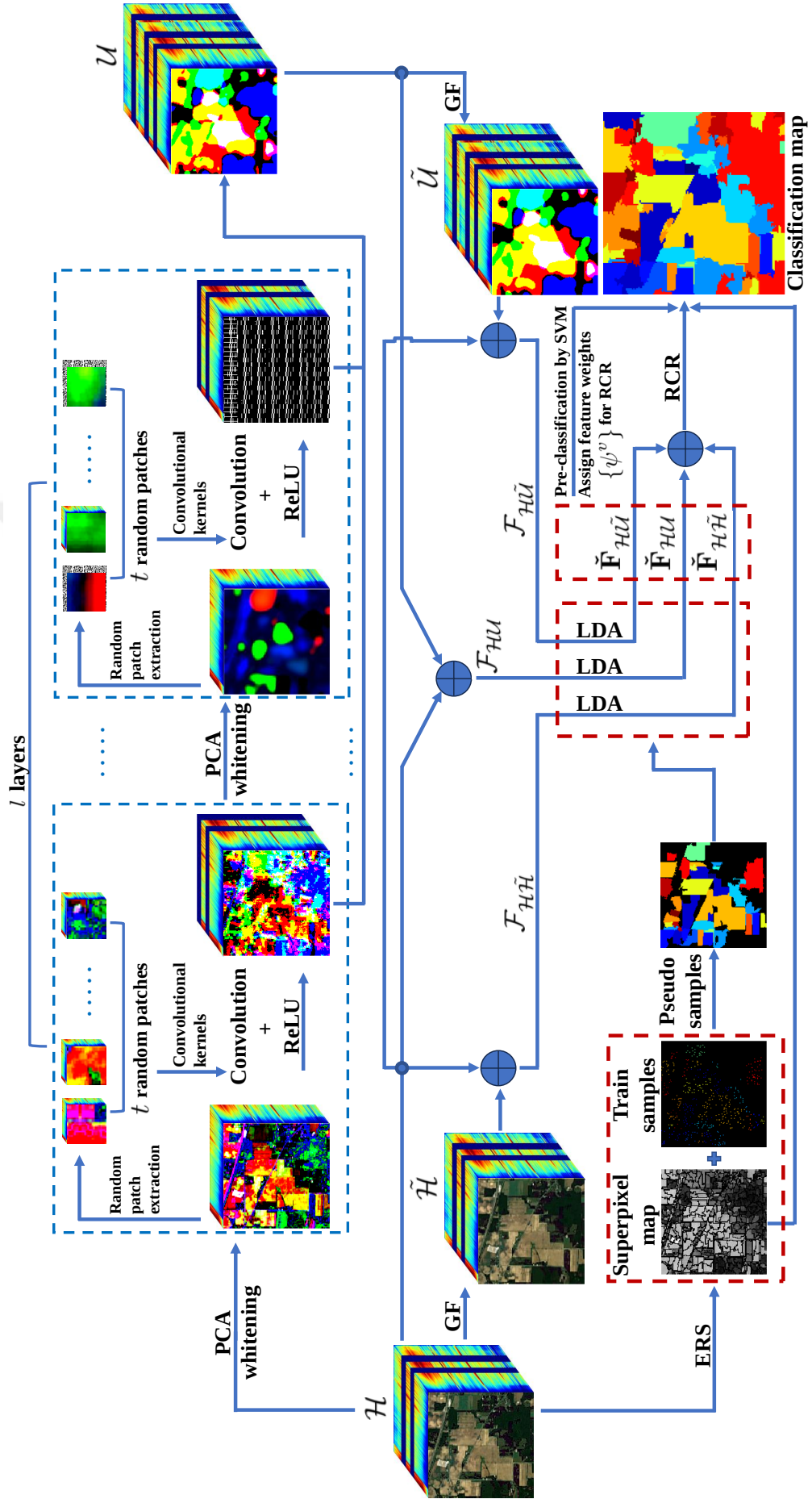


Figure 5.1 Flowchart of the proposed GRR.

5.3 Experimental Results

The adjustment of the parameters and comprehensive comparison results are provided in this section. The experiments are tested on the IP2, PU and WHHH datasets, respectively. There are several parameters of the GRR such as number of random patches t , number of principal components h , patch size w , layer depth l , superpixel patch size T , RCR regularizers λ and τ . While the optimal value of a parameter within a range of values is determined, other variables are kept fixed to observe the effect of the parameter of interest. In order to demonstrate the efficacy of GRR with its determined optimum parameters, a few recently proposed algorithms called JCRC-MTL [78], RPNet [82], RPCC [83], SSRPNet [84], GRPC [85], RPNet-RF [86] and MS-RPNet [87] are utilized, respectively. The optimal parameters of the methods used for comparison are tuned to obtain the highest accuracy results. Additionally, the RCR-based classifier of the proposed GRR is replaced by the SVM-based GRR (GR-SVM) to indicate the impact of RCR-based classification on the extracted feature sets. This GR-SVM is also utilized in comparisons. The following subsections present the experimental results in detail for the IP2, PU and WHHH datasets, respectively. For both comparisons and parameter adjustments, all experiments are performed ten times and then the average of the trials is calculated.

5.3.1 Indian Pines 2 Dataset

Throughout the experiments, the random distribution of train and test samples are determined in each class as 5% and 95%, respectively for the IP2 dataset. The OA and AA results of GRR with respect to the parameter intervals of t , h , l and w , are given in Figure 5.2. It is observed from Figure 5.2(a) that the number of convolution kernels $t = 60$ provides the highest OA and AA. In accordance with the Figure 5.2(b), between $h = 3$ and $h = 7$, the impact of principal components on the OA and AA results is increasing considerably, however, the results appear stable after $h = 7$ and decrease after $h = 13$. Hence, the optimal parameter of h is chosen as 13. According to the Figure 5.2(c), the impact of layer number l is remarkable between $l = 3$ and $l = 5$. After this point, l has less impact on the OA and AA results up to $l = 9$ which is the value

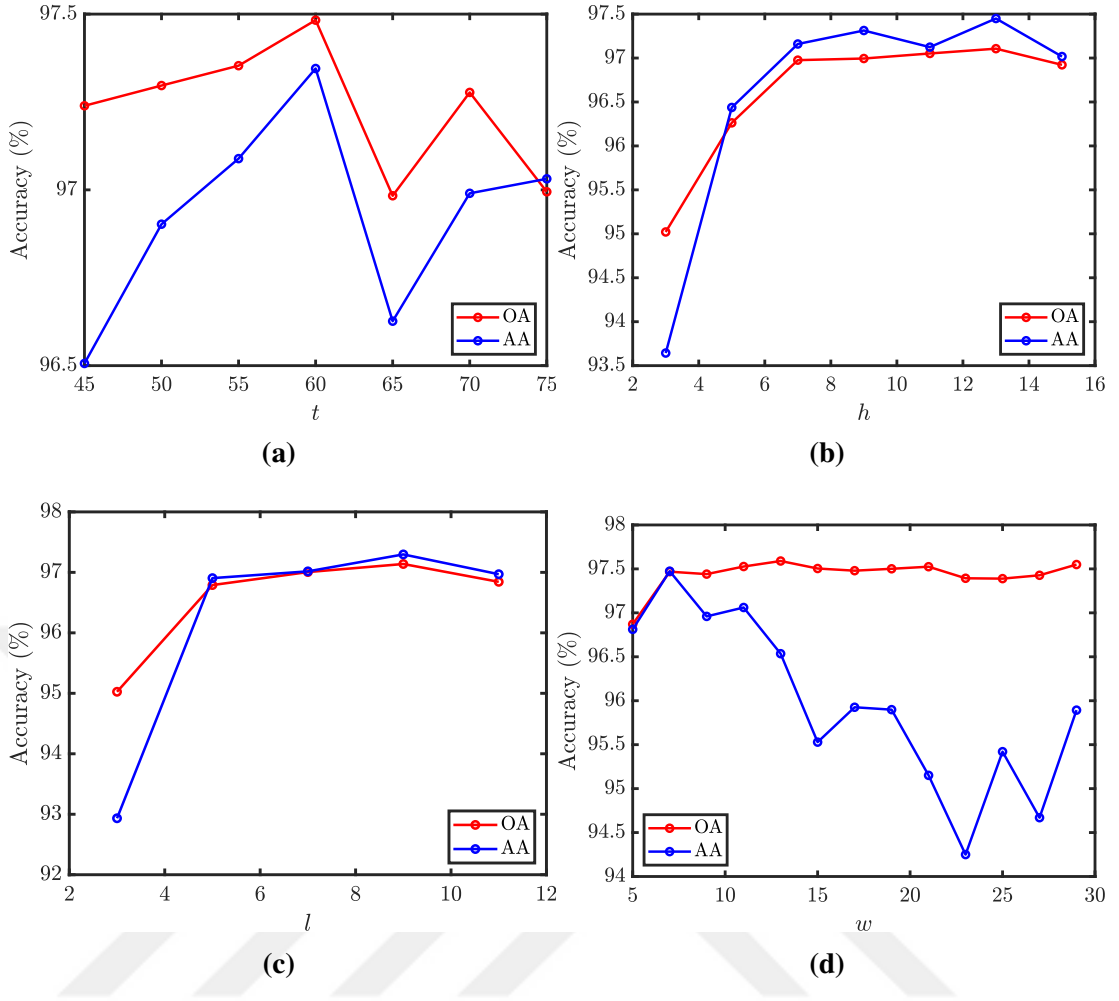


Figure 5.2 IP2 dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w .

providing the highest accuracies. It is seen from the Figure 5.2(d) that the window size of kernels does not have great influence on the OA results, however, it affects the AA results a lot. Since the highest AA is obtained at $w = 7$, this parameter is chosen as the optimal. To sum up, the optimum parameters of GRR are tuned as $t = 60$, $h = 13$, $l = 9$ and $w = 7$, respectively.

The GRR uses segmented HSI for both pseudo-sample production and RCR-based classification of the extracted feature sets. Figure 5.3 demonstrates the effect of selected parameter intervals of T on the OA and AA. It is seen that the highest results of OA and AA are obtained at different parameters. For instance, while OA hits the highest at $T = 8$, AA attains the top at $T = 4$. When the effect of $T = 4$ and $T = 8$ parameters on OA and AA are compared, the OA of $T = 4$ is approximately 1%

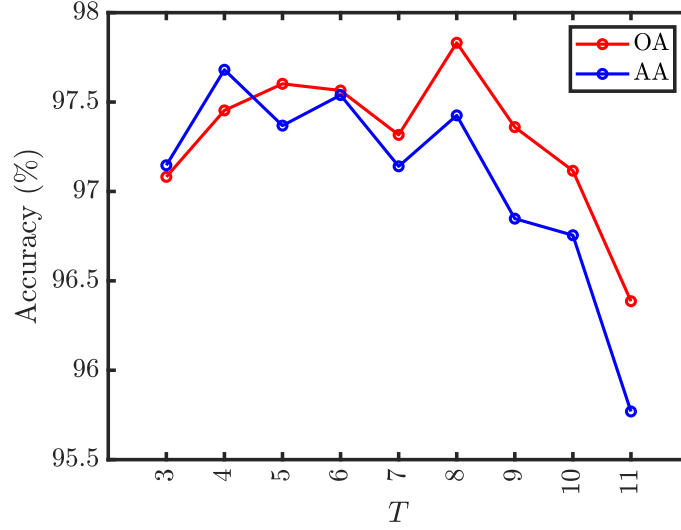


Figure 5.3 IP2 dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

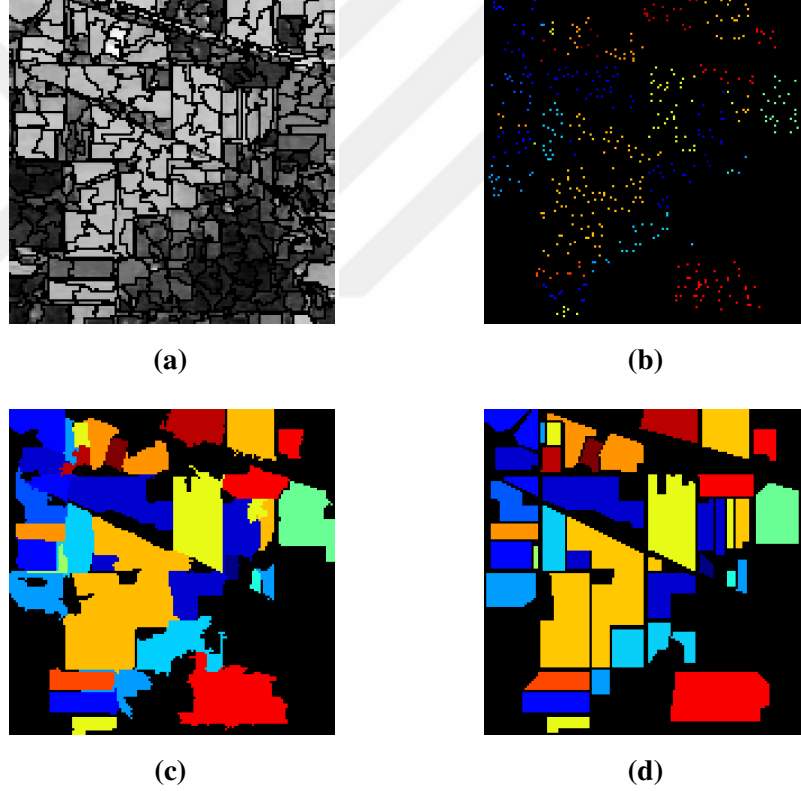
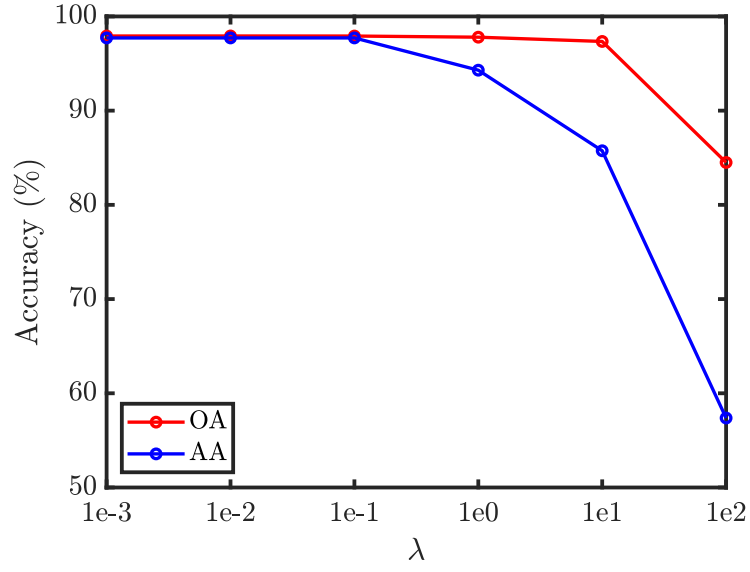
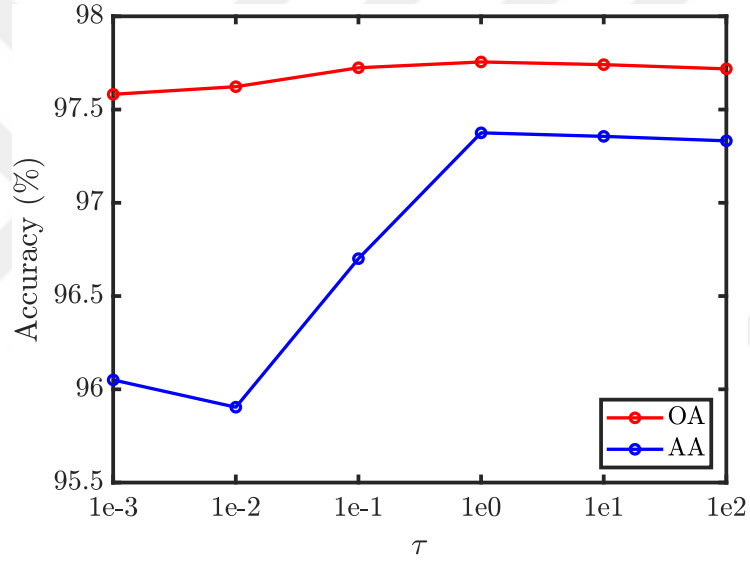


Figure 5.4 IP2 dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth.

less than at $T = 8$. Similarly, the AA of $T = 4$ is approximately 0.5% more than at $T = 8$. On account of this observation, the optimal value is determined as $T = 8$ because it has more impact on the results compared to $T = 4$. After this adjustment,



(a)



(b)

Figure 5.5 IP2 dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ .

the segmented IP2 dataset is given in Figure 5.4. Additionally, the ground-truth of IP2 dataset, randomly distributed train samples, and obtained pseudo-samples affiliated with the segmentation map and chosen train samples, are also presented in Figure 5.4.

There are remaining two parameters to be adjusted called λ and τ , respectively. Figure 5.5 presents the corresponding OA and AA results of the parameter intervals of $\lambda = [1e-3, 1e2]$ and $\tau = [1e-3, 1e2]$, respectively. It is seen from the Figure 5.5(a) that

the AA is starting to fall after $\lambda = 1e-1$, and the OA has similar results except $\lambda = 1e2$. Therefore, the optimum parameter of λ is adjusted as $1e-1$. As for the parameter τ , it is set as $\tau = 1e0$ because the OA results are quite similar for the parameters within the interval, but, the AA hits the maximum value at $\tau = 1e0$ which is also the highest point of OA.

After tuning all of the parameters of GRR, the accuracy results are compared with the results of the other methods. Table 5.1 presents the results with respect to the OA, AA, κ and class accuracies including standard deviations. The running times of methods are also given in the Table 5.1. It is seen that the GRR attains the highest results in terms of all evaluation metrics, and the results are followed by GR-SVM. It means that the efficient use of extracted samples in RCR-based classifier compared to the SVM-based classification improves the results. The GRR obtains nearly 1%, 2% and 1.5% more OA result than the GR-SVM. Besides GR-SVM, the GRR has better results than both RPNNet-based and RCR-based methods. For instance, the JCRC-MTL which utilizes conventional feature extractors and RCR classifier with fixed-size patches, has approximately 2.8% and 8% less OA and AA results than the GRR, respectively. Except the JCRC-MTL, the remaining methods utilized RPNNet-based feature extraction stage followed by various technique to improve the results. Among the all RPNNet-based models, SSRPNNet provides better accuracy results than the others. Compared to the SSRPNNet, the GRR has approximately 1.5% and 9% more OA and AA results than the SSRPNNet. The efficacy of proposed GRR can be observed at 9th class where only single train sample is available. Due to the superpixel-guided RCR classifier, the GRR completely assigns the samples into the true label while the others misclassify these samples. As for the computation cost of methods, the JCRC-MTL takes quite a long time to classify the samples compared to the others because it forms a patch around the test sample each time during RCR-based classification. Among them, GRPC is the fastest algorithm due to the dimension-reduction stage before feature extraction. The computation times of proposed GR-SVM and GRR are within an acceptable range. The required little bit more computation time of GRR is caused by the calculations steps of RCR.

Figure 5.6 displays the corresponding classification maps. Due to the further spatial

Table 5.1 IP2 dataset: Classification accuracy results under 5% labeled samples obtained by RPNNet, RPNNet-RF, RPCC, GRPC, MS-RPNNet, JCRC-MTL, SSRPNNet, GR-SVM and proposed GRR

*The highest result is bolded and the following result is underlined in each line.

Class	Train	Test	RPNNet	RPNNet-RF	RPCC	GRPC	MS-RPNNet	JCRC-MTL	SSRPNNet	GR-SVM	GRR
1	3	43	75.23	90.58	92.44	82.55	92.55	95.81	99.53	95.34	<u>96.04</u>
2	72	1356	91.11	93.19	94.03	91.90	93.57	91.65	97.34	97.33	96.66
3	42	788	87.81	92.60	91.85	92.39	93.73	88.56	95.73	<u>98.85</u>	99.74
4	12	225	60.26	90.31	93.82	97.88	95.55	94.00	97.86	93.24	96.08
5	25	458	89.75	90.25	93.16	92.60	94.93	92.29	<u>94.19</u>	91.96	93.82
6	37	693	97.51	93.51	93.05	93.53	94.52	<u>98.62</u>	95.15	96.65	99.30
7	2	26	85.38	95.00	93.07	90.96	93.84	<u>98.07</u>	80.76	98.46	96.53
8	24	454	96.60	99.57	98.81	98.97	98.98	<u>99.95</u>	99.91	100	100
9	1	19	35.26	49.21	55.00	66.05	51.84	0	24.21	<u>89.47</u>	100
10	49	923	88.55	90.11	89.97	88.62	91.37	90.96	92.97	<u>94.95</u>	95.26
11	123	2332	95.77	94.41	93.91	94.21	94.90	97.37	98.54	98.00	<u>98.39</u>
12	30	563	81.19	87.54	86.59	90.82	88.59	92.62	93.99	<u>95.13</u>	98.54
13	11	194	98.24	94.43	91.08	93.58	93.42	99.17	87.73	99.48	<u>99.27</u>
14	64	1201	98.34	97.33	97.42	98.01	98.46	99.99	99.85	97.36	99.92
15	20	366	89.30	94.19	94.23	95.09	95.72	<u>99.04</u>	99.45	87.97	93.79
16	5	88	64.26	76.64	81.98	87.72	88.92	<u>95.34</u>	59.54	97.04	97.04
OA	–	–	91.61	93.26	93.29	93.51	94.39	95.08	96.39	<u>96.66</u>	97.85
			(± 1.06)	(± 0.63)	(± 0.96)	(± 0.95)	(± 0.86)	(± 0.58)	(± 0.54)	(± 0.33)	(± 0.60)
AA	–	–	83.41	89.30	90.02	90.93	91.31	89.59	88.55	<u>95.70</u>	97.52
			(± 1.82)	(± 1.99)	(± 1.74)	(± 2.45)	(± 2.07)	(± 0.71)	(± 3.69)	(± 1.53)	(± 0.62)
κ	–	–	90.38	92.32	92.35	92.61	93.61	94.38	95.88	<u>96.18</u>	97.55
			(± 1.22)	(± 0.72)	(± 1.10)	(± 1.08)	(± 0.99)	(± 0.66)	(± 0.63)	(± 0.38)	(± 0.69)
Runtime (s)	–	–	2.76	<u>2.39</u>	10.51	2.19	17.21	39.53	22.47	7.72	8.92

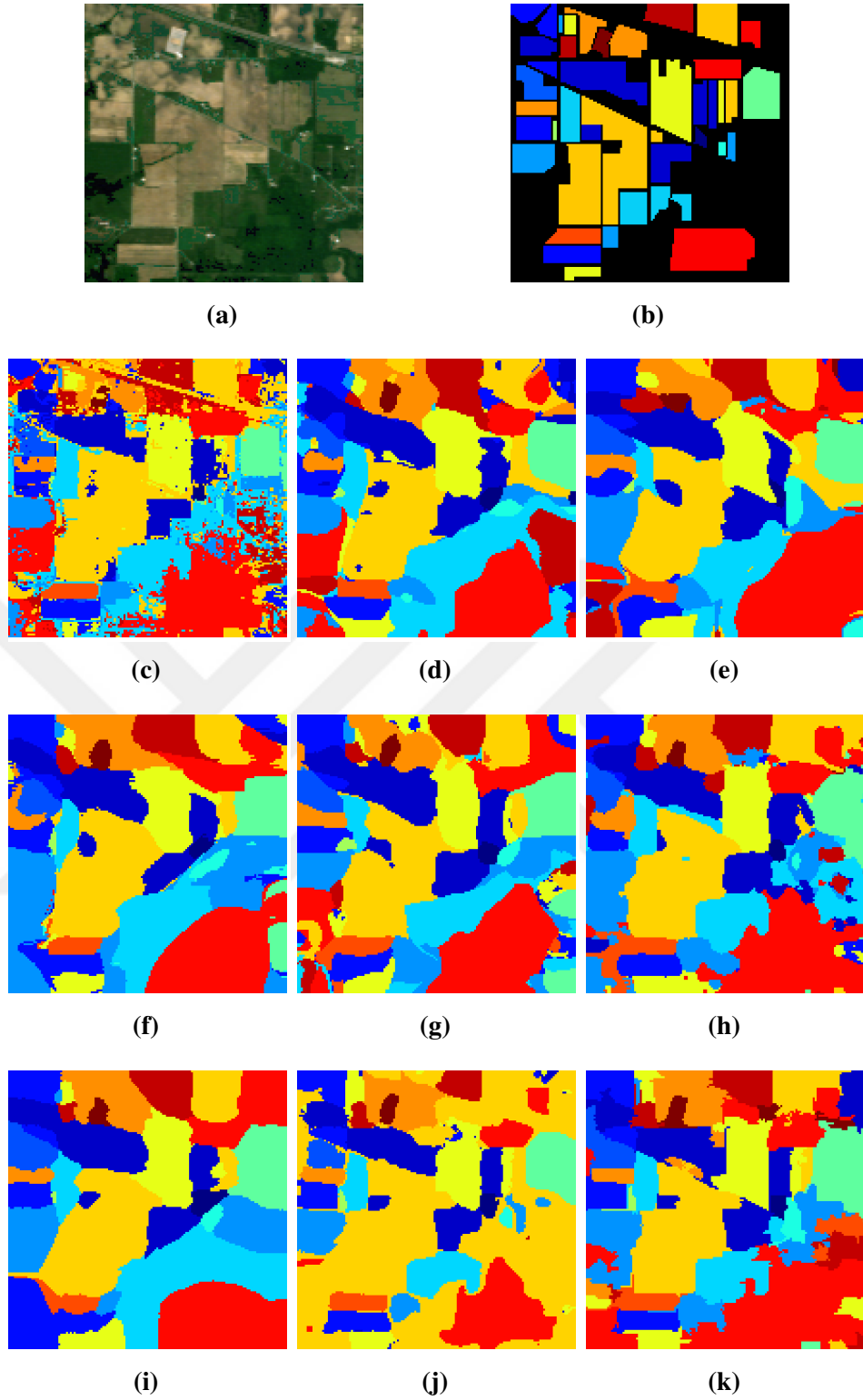


Figure 5.6 IP2 dataset: (a) False-color image, (b) Ground-truth; classification maps with OAs obtained by (c) RpNet (91.97%), (d) RpNet-RF (93.39%), (e) RPCC (93.94%), (f) GRPC (94.51%), (g) MS-RpNet (95.19%), (h) JCRC-MTL (95.31%), (i) SSRpNet (96.71%), (j) GR-SVM (96.92%) and (k) GRR (98.24%).

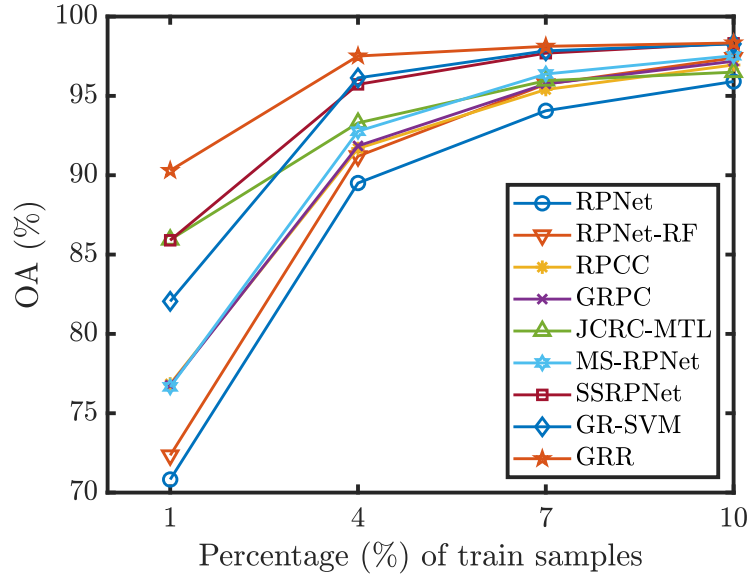


Figure 5.7 IP2 dataset: OA results of the methods RpNet, RpNet-RF, RPCC, GRPC, MS-RpNet, JCRC-MTL, SSRpNet, GR-SVM and proposed GRR under varying training percentages.

filtering phases, the classification maps of the methods out of RpNet exhibit smooth characteristics. This indicates the importance and impact of using additional spatial filters. Compared to the other methods in terms of structural preserving, the GRR provides the best classification map because the class borders and spatial structure are maintained better. The reason is that the usage of GF process in the GRR, and also superpixel-guided RCR classifier. While the GF preserves the border information, superpixels maintain the geometrical structure of the HSI. It is also seen that some of the maps such as RPCC, GRPC, MS-RpNet and especially SSRpNet have quite over-smoothed results. Compared to these, the GRR does not have an over-smoothed output, and has a map providing more closer result to the map of ground-truth.

In addition to presented both qualitative and quantitative results, an experiment is also performed to analyze the effect of available train samples on the OA result by varying the percentages per class. According to the Figure 5.7, the proposed GRR provides 5% more OA than the result of the closest methods as JCRC-MTL and SSRpNet under 1% labeled samples, respectively. Starting from the percentage of 4%, the OA results of other methods approach the result of the GRR. Consequently, it is inferred that the proposed GRR is capable of classifying the pixels accurately even if few train samples are provided.

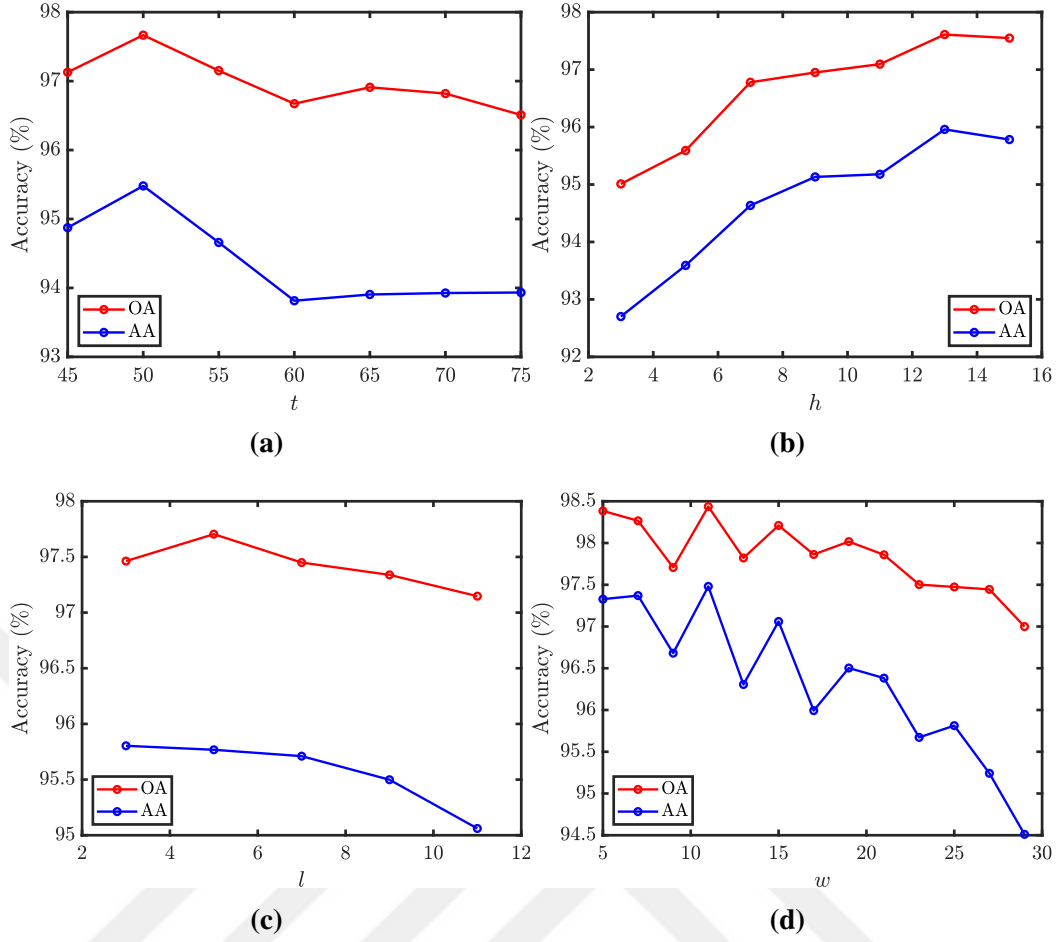


Figure 5.8 PU dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w .

5.3.2 Pavia University Dataset

All of the experiments are implemented by dividing the data as 1% train and 99% test samples in each class of the PU dataset. In order to adjust the optimal parameters of t , h , l and w , the same rule is applied as in the matter of IP2 dataset, where different parameter intervals of the variables are cross-validated. The outputs are monitored in terms of OA and AA results which can be seen from Figure 5.8. The optimal number of t is chosen as $t = 50$ because it is the highest point for both of OA and AA, respectively, as observed from Figure 5.8(a). After this point, the results of OA and AA start to fall significantly. As for the optimum number of principal components h , it is seen from the Figure 5.8(b) that both of OA and AA values trace each other up to the maximum point where $h = 13$. Then, both of them begin to fall slightly. Due to the different optimal parameters of layer number l for the OA and AA results according to the

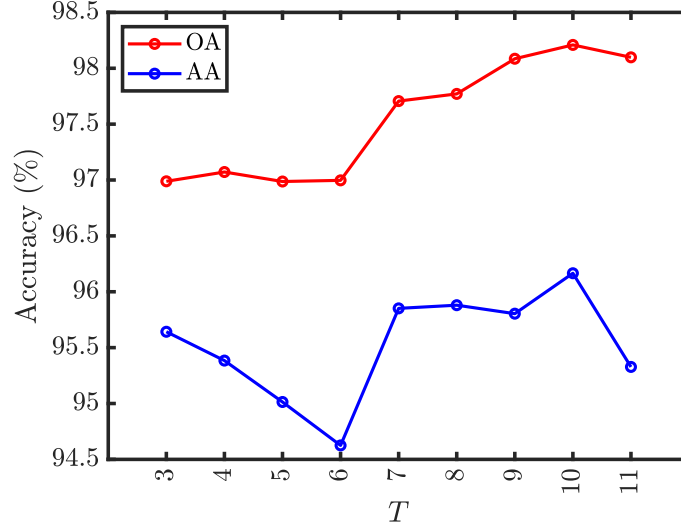


Figure 5.9 PU dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

Figure 5.8(c), the following strategy is applied. While the first and second highest results are obtained at $l = 5$ and $l = 3$ respectively for OA, these results are acquired at $l = 3$ and $l = 5$ respectively for AA. Since the difference between these points is higher for OA compared to AA, the optimum parameter is decided as $l = 5$. Since the PU dataset has a non-uniform textural characteristics, the size of regions enclosed by the classes varies quite different which affects the optimal patch size w as can be analyzed in Figure 5.8(d). It is seen that the value of patch size $w = 11$ is the point where both of OA and AA hit their supreme points. To summarize, the parameters are set as $t = 50$, $h = 13$, $l = 5$ and $w = 11$ to obtain the RPNet features efficiently.

Due to the superpixel-guided production of pseudo-samples and classification of the feature sets, the parameter T representing the superpixel patch size has a significant impact. The OA and AA results obtained by changing T between 3 and 11, is presented in Figure 5.9. It is seen that a gradual rise is observed for OA up to the point $T = 10$, then a slight decrease is occurred. As for the AA, a drastic decrease is monitored between $T = 3$ and $T = 6$, then the results start increasing again until the value $T = 10$, which is the peak point. Therefore, the optimum parameter of T is adjusted as $T = 10$. After this tuning, the segmented PU dataset is provided in Figure 5.10. In addition, the ground-truth of PU dataset, randomly selected train samples, and obtained pseudo-samples associated with the segmentation map and chosen train samples, are also given in Figure 5.10.

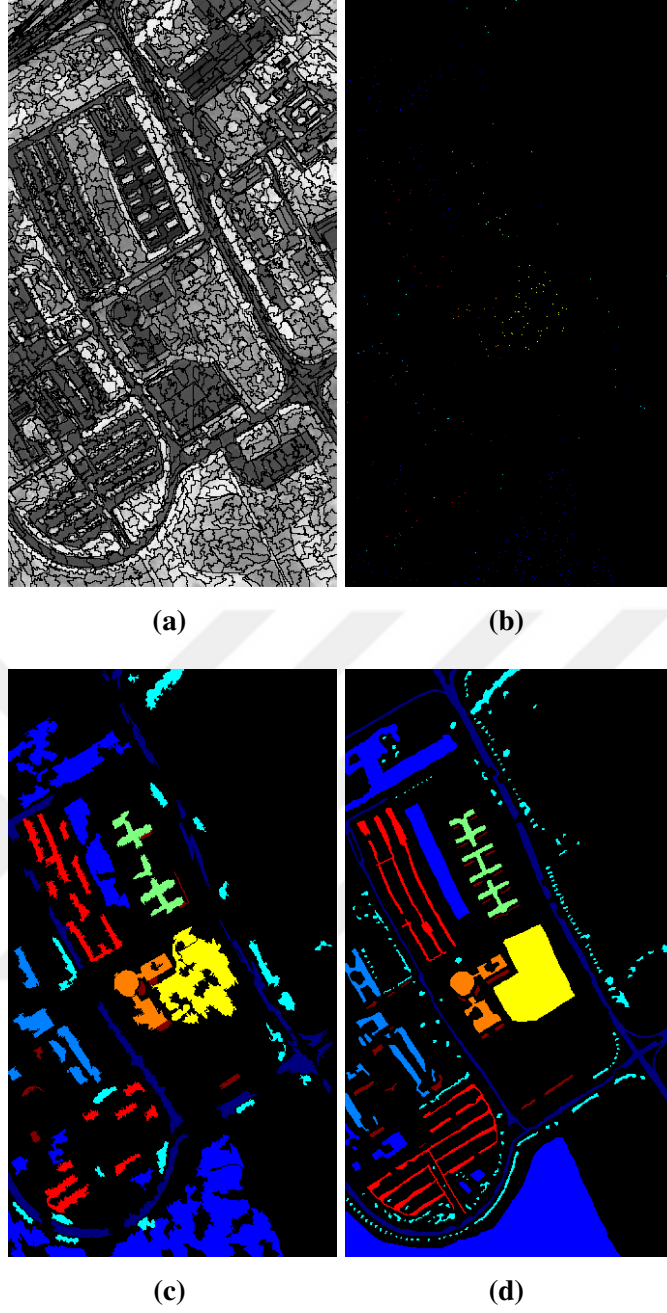
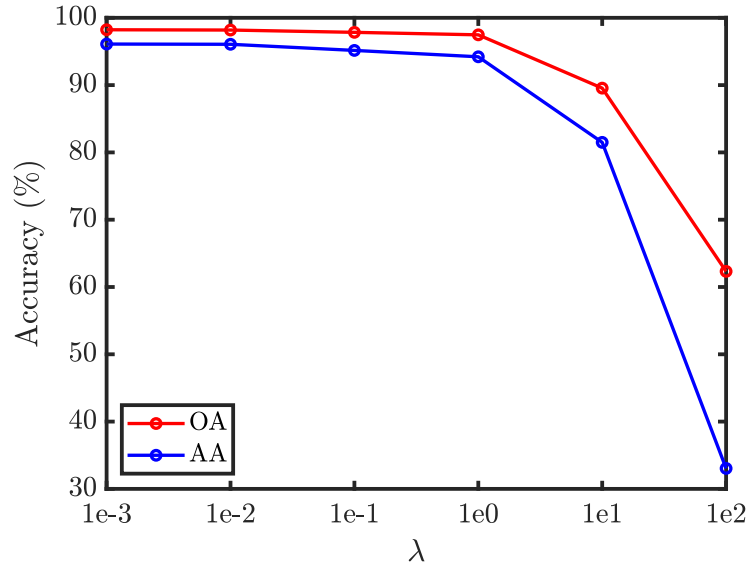
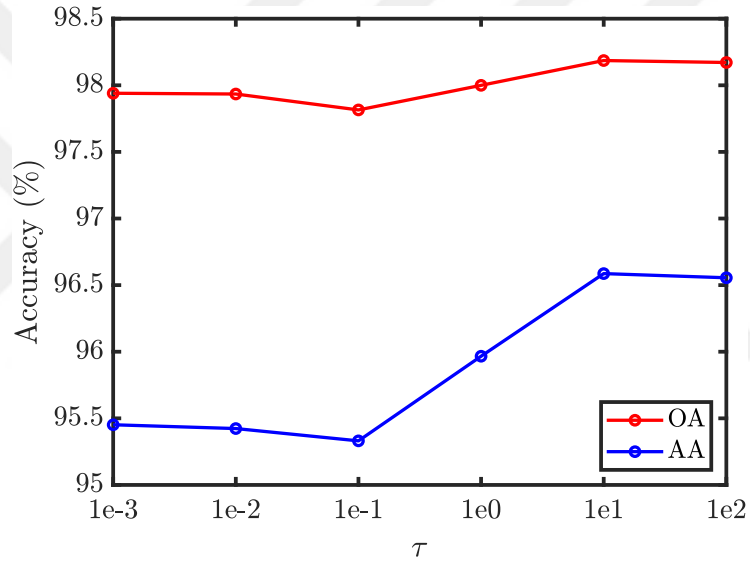


Figure 5.10 PU dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth.

The same parameter intervals are utilized as in the IP2 dataset to find the optimum values of λ and τ respectively, as demonstrated in the Figure 5.11. According to the Figure 5.11(a), the OA and AA results are both close to each other and do not vary much in the intervals of $\lambda = 1e-3$ and $\lambda = 1e0$, where they are both the highest at $\lambda = 1e-3$. However, severe decreases begin after $\lambda = 1e0$ for both of OAs and AAS. It is observed from the Figure 5.11(b) that while OAs do not vary considerably



(a)



(b)

Figure 5.11 PU dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ .

corresponding to the changes in τ , the AA results change especially between $\tau = 1e-1$ and $\tau = 1e1$. Then, a slight decrease starts after $\tau = 1e1$. Hence, the optimum values of λ and τ are tuned as $\lambda = 1e-3$ and $\tau = 1e1$, respectively.

After adjusting the parameters of the GRR to optimum values, the performance is compared to the others. Table 5.2 provides the results in accordance with the class accuracies, OA, AA and κ with their standard deviations under 1% train samples in each class. In addition, the computation times of the methods are also shared in the

Table 5.2. As it is observed, the GRR provides the best results in terms of OA, AA and κ , and also maintains the first or second highest results in most classes. The metrics of GRR is followed by the results of GR-SVM which is around 1.5% less. This indicates that RCR-based classification enables the extracted feature sets to be classified efficiently. Apart from the superiority of GRR on GR-SVM, the GRR also achieves higher results compared to both RPNNet-based and RCR-based methods. Compared to the JCRC-MTL exploiting conventional feature extractors and RCR classifier with fixed-size patches, the GRR obtains approximately 2.5% and 3.5% higher OA and AA metrics, respectively. These results demonstrate that the use of shallow and deep features, and also superpixel-guided classification instead of patch-based classification improve the performance. Besides the JCRC-MTL, the remaining methods utilizing RPNNet-based feature extraction and further spatial processing have lower accuracy results compared to the GRR. For instance, SSRPNet has the highest accuracy results among the all RPNNet-based models. However, the accuracy metrics of GRR exceed the OA, AA and κ results of SSRPNet by almost 2%, 6% and 3%, respectively. As for the runtime of methods, it is seen that all of the methods have higher running time compared to the IP2 dataset due to the larger scale of PU dataset. As an instance, the computation of local covariances in the method RPCC advances the runtime since more local regions exist in the PU, compared to the IP2 dataset. Also, the stages of 2D-SSA and S3-PCA calculations in the MS-RPNNet, the patch-based classification in the JCRC-MTL, graph-based classification of the SSRPNet require much time for the PU dataset than time for the IP dataset. As for GRR, the GF stage requires much time due to the calculation of weights of graph-nodes. The computation time of GRR is almost 4.5s longer in comparison to the GR-SVM because of the RCR-based label assignment. Among all, the lowest runtime belong to RPNNet and RPNNet-RF, respectively.

Figure 5.12 presents the regardful classification maps of the methods. It is seen that the maps except RPNNet's one seem smoothed because of the additional spatial filtering stages. It proves that the further use of spatial filtering process enhances the results. However, it is also observed that the maps RPCC, GRPC, JCRC-MTL and especially SSRPNet seem pretty over-smoothed characteristics. Compared to these, the proposed GRR does not have an over-smoothed output, and has a map providing more closer result to the map of ground-truth. Additionally, the visual outputs indicate that GF-

Table 5.2 PU dataset: Classification accuracy results under 1% labeled samples obtained by RPNNet, RPNNet-RF, RPCC, GRPC, GRPC, MS-RPNNet, JCRC-MTL, SSRPNNet, GR-SVM and proposed GRR

*The highest result is bolded and the following result is underlined in each line.

Class	Train	Test	RPNNet	RPNNet-RF	RPCC	GRPC	MS-RPNNet	JCRC-MTL	SSRPNNet	GR-SVM	GRR
1	67	6564	91.02	91.46	91.76	92.96	89.14	92.69	99.44	98.01	98.51
2	187	18462	99.32	99.17	99.53	99.01	99.01	<u>99.54</u>	99.24	99.43	99.96
3	21	2078	75.70	76.08	87.26	87.68	91.60	<u>96.51</u>	95.44	96.02	96.93
4	31	3033	87.74	84.60	70.84	76.05	91.72	<u>72.97</u>	73.08	88.92	<u>91.58</u>
5	14	1331	50.33	97.55	91.29	92.51	99.19	100	98.44	99.87	99.21
6	51	4978	88.75	95.81	99.62	99.44	88.30	<u>99.84</u>	100	98.78	98.01
7	14	1316	80.16	76.73	95.11	93.29	95.69	<u>99.28</u>	99.52	98.73	99.68
8	37	3645	94.26	85.71	88.00	86.36	92.10	<u>96.59</u>	98.98	95.37	96.26
9	10	937	70.90	77.46	50.53	54.48	<u>82.26</u>	<u>77.40</u>	46.91	77.13	85.84
OA	–	–	91.60 (± 0.76)	93.02 (± 0.78)	93.21 (± 0.34)	93.47 (± 0.93)	94.27 (± 1.34)	95.72 (± 0.52)	96.10 (± 0.51)	97.37 (± 0.44)	98.10 (± 0.51)
AA	–	–	82.02 (± 2.40)	87.17 (± 1.24)	85.99 (± 1.66)	86.87 (± 1.74)	92.11 (± 3.20)	92.76 (± 1.02)	90.12 (± 1.41)	94.70 (± 1.67)	96.22 (± 1.01)
κ	–	–	88.58 (± 1.09)	90.69 (± 1.04)	90.95 (± 0.46)	91.31 (± 1.24)	92.38 (± 1.78)	94.29 (± 0.71)	94.83 (± 0.69)	<u>96.51</u> (± 0.59)	97.47 (± 0.68)
Runtime (s)	–	–	5.98	<u>8.32</u>	81.49	9.20	127.43	267.64	223.87	34.16	38.68

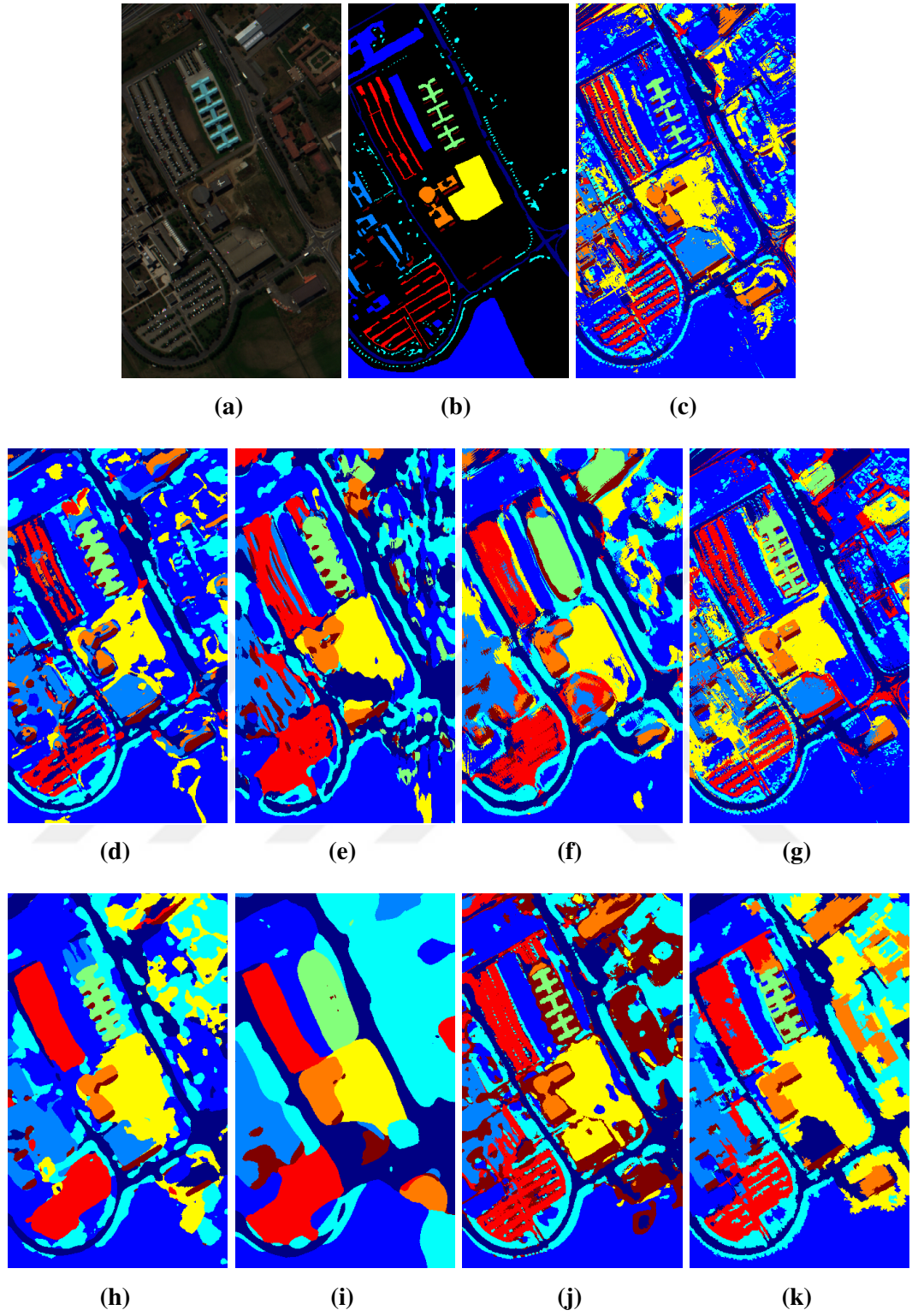


Figure 5.12 PU dataset: (a) False-color image, (b) Ground-truth; classification maps with OAs obtained by (c) RpNet (92.03%), (d) RpNet-RF (93.34%), (e) RPCC (93.53%), (f) GRPC (94.27%), (g) MS-RpNet (95.32%), (h) JCRC-MTL (96.18%), (i) SSRpNet (96.59%), (j) GR-SVM (97.51%) and (k) GRR (98.58%).

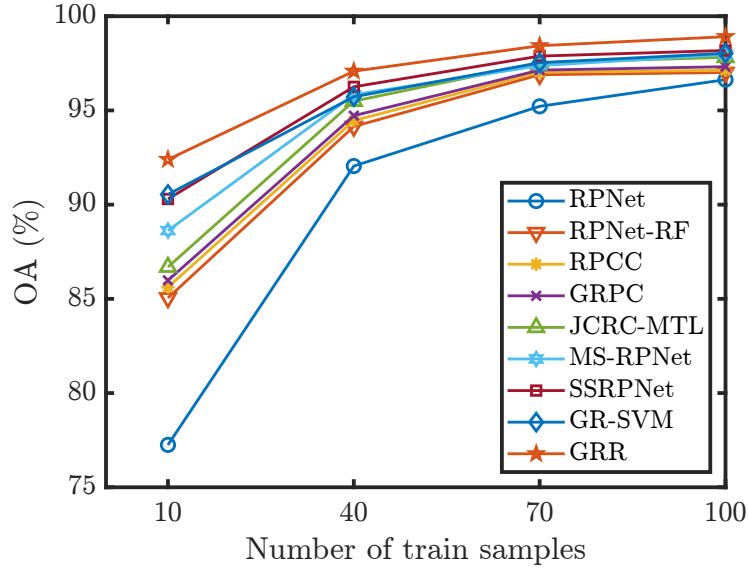


Figure 5.13 PU dataset: OA results of the methods RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, JCRC-MTL, SSRPNet, GR-SVM and proposed GRR under varying training samples per class.

based edge protection and superpixel-based RCR classifier allow the GRR to obtain more correctly classified pixels and more preserved borders, simultaneously.

In order to analyze the performance of methods in terms of OA result by varying the size of train samples, an extra experiment is implemented which simulates each of the methods under small number of available samples given as 10, 40, 70 and 100 in each class. In accordance with the Figure 5.13, the GRR is capable of obtaining approximately 3% more OA than the result of the closest methods as GR-SVM and SSRPNet under 10 samples per class. Starting from 40 samples per class, the OA results of other methods reach the result of the GRR. Therefore, it is reasonable to say that the proposed GRR classifies the samples more accurately, and then obtains higher OA results compared to the others even if few train samples per class are utilized.

5.3.3 WHU-Hi-HongHu Dataset

Throughout the experiments, the WHHH dataset is partitioned into 0.5% train and 99.5% test samples in each class respectively, because of the abundance of samples. The optimum parameters of t , h , l and w are similarly tuned by the rule applied for IP2 and PU datasets, where various ranges of parameters are cross-validated to each other. The results are observed with regards to OA and AA which can be seen from Figure

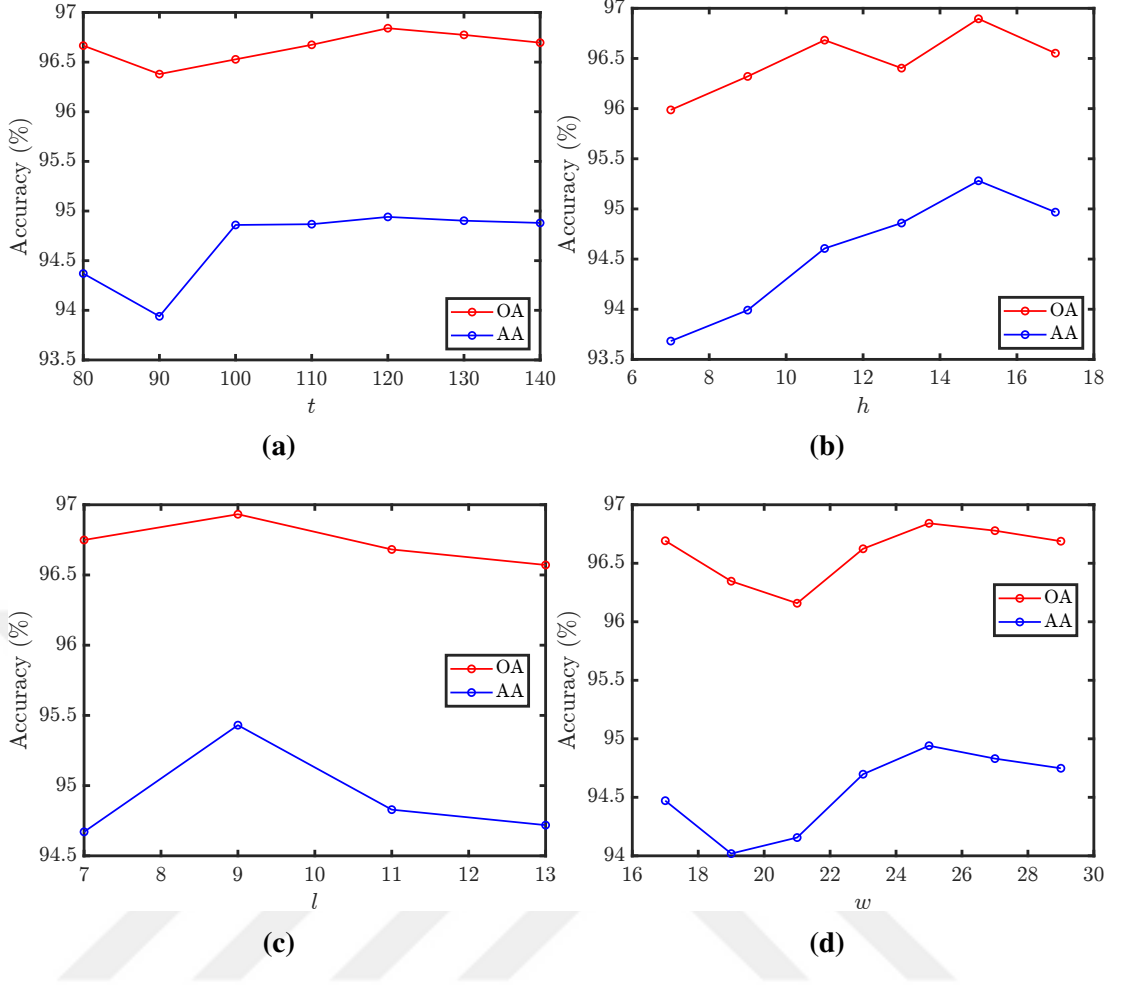


Figure 5.14 WHHH dataset: OA and AA results of the proposed GRR under different (a) random patches t , (b) principal components h , (c) layer depths l and (d) patch sizes w .

5.14. The optimum value of t is determined as 120 since it is the highest point for both of OA and AA, respectively, as investigated from the Figure 5.14(a). After this optimal point, the values of OA and AA start to fall slightly. As for the parameter h , it is observed from the Figure 5.14(b) that both of OA and AA results follow each other until the highest point of $h = 15$. Then, both of them begin to fall considerably. According to the Figure 5.14(c) which shows the variations of OA and AA in accordance with the layer number l , the value $l = 9$ is determined as the optimum because both of OA and AA attain the maximum values. Since the WHHH dataset has many large-scale spatial regions, it is seen from the Figure 5.14(d) that the patch size $w = 25$ is the value where both of OA and AA reach their highest points. Eventually, the parameters are specified as $t = 120$, $h = 15$, $l = 9$ and $w = 25$, respectively.

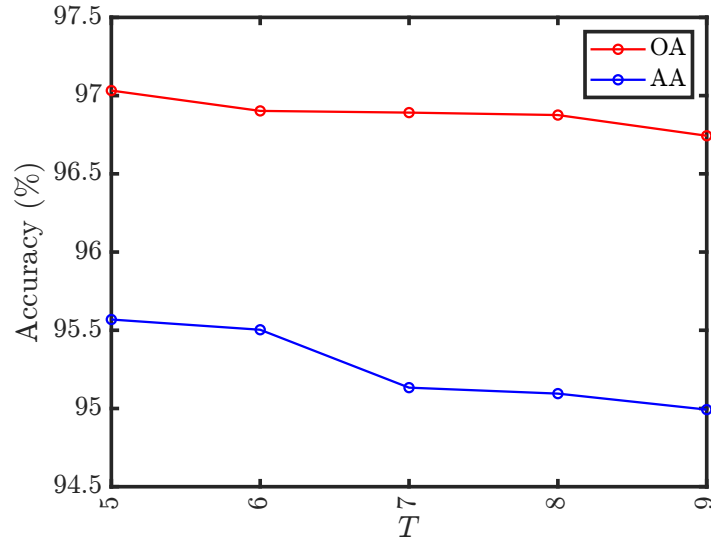


Figure 5.15 WHHH dataset: OA and AA results of the proposed GRR under different superpixel patch size T for the LDA.

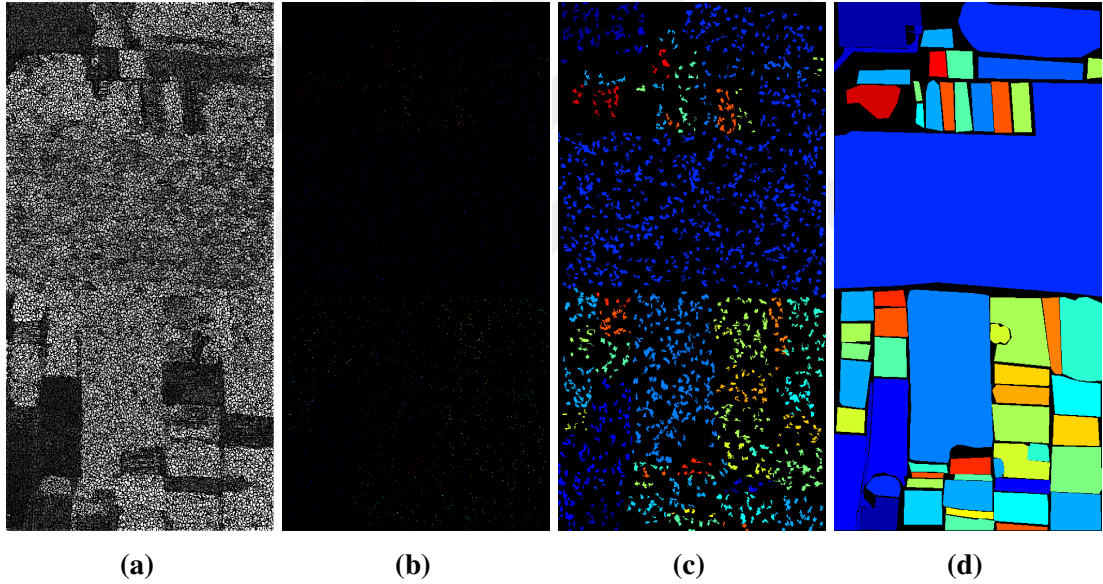
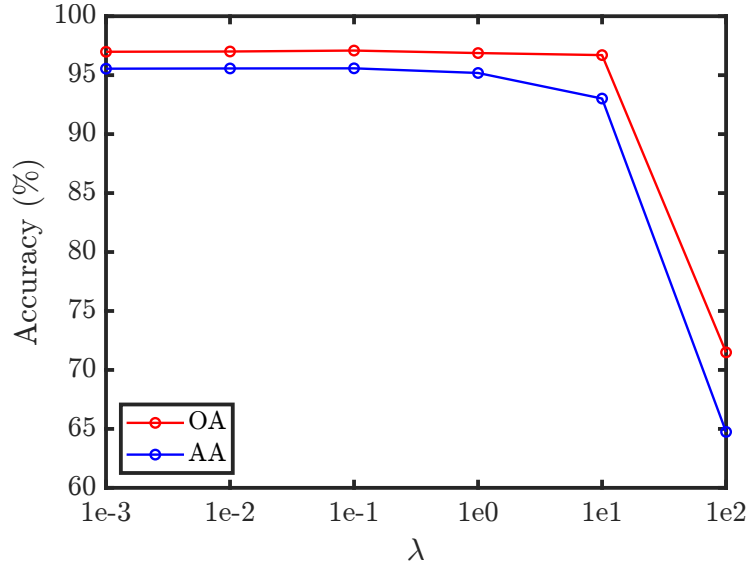
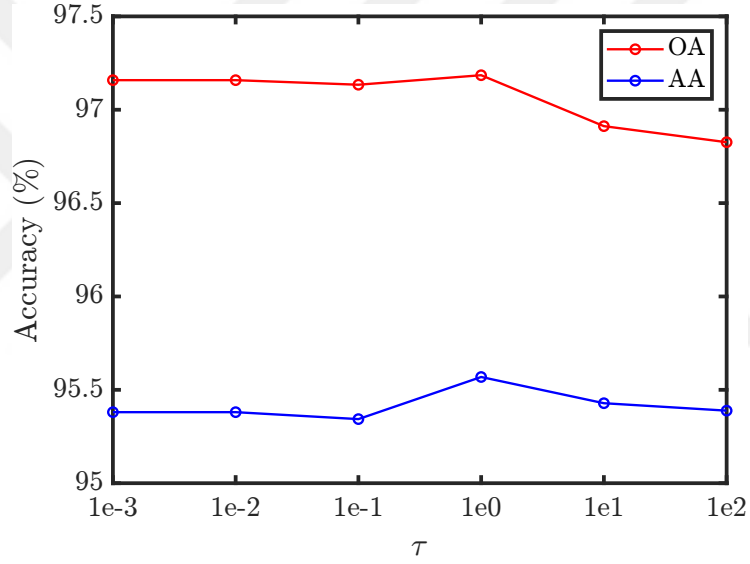


Figure 5.16 WHHH dataset: The maps of (a) Superpixel segmentation, (b) Train samples, (c) Pseudo samples for LDA and (d) Ground-truth.

As mentioned before, the parameter T controlling superpixel patch size has a great effect on the performance of GRR because it directly affects the generation of pseudo-samples, and the RCR-based sample classification. The corresponding OA and AA results acquired by varying T between 5 and 9, is given in the Figure 5.15. It is observed that the results of both OA and AA drop slightly with the increase of T . This can be interpreted as some large areas include a region belonging to a different class, or the dataset has some classes enclosed by small regions. Such these problems can



(a)



(b)

Figure 5.17 WHHH dataset: OA and AA results of the proposed GRR under the different values of the regularizers (a) λ and (b) τ .

cause misclassification as a result of the selection of large patch size. Hence, the optimum value is decided as $T = 5$ because of the highest OA and AA results. After this adjustment, the segmented WHHH dataset is given in Figure 5.16. Additionally, the ground-truth of WHHH dataset, randomly partitioned train samples, and generated pseudo-samples in accordance with the segmentation map and selected train samples, are also given in Figure 5.16.

The ranges of parameters are kept constant to determine the optimal values of λ and τ

respectively, as in the case of IP2 and PU datasets. The effect of parameters λ and τ on the OA and AA results, is provided in Figure 5.17. According to the Figure 5.17(a), the OA values are quite close to each other between the intervals of $\lambda = 1e-3$ and $\lambda = 1e1$. However, there is a serious fall after $\lambda = 1e1$. Meanwhile, a small amount of decrease starts from the value $\lambda = 1e-1$ for the AA results. So, $\lambda = 1e-1$ is determined as the optimum value. It is seen from the Figure 5.17(b) that the value $\tau = 1e0$ is point where both of OA and AA achieve their maximum values. To summarize, the values of $\lambda = 1e-1$ and $\tau = 1e0$ are chosen as the optimum parameters.

After tuning the parameters of the GRR to the optimal values, the classification results are compared to the other methods. Table 5.3 presents the results with respect to the class accuracies, OA, AA and κ with their standard deviations under 0.5% train samples in each class. Additionally, the runtime of the methods are also given in the Table 5.3. It is seen that, the GRR obtains the highest results in terms of OA, AA and κ , and also provides the best or second best results in most classes. The results of GRR are followed by GR-SVM providing approximately 1.5% and 3.5% less OA and AA, respectively. It can be inferred from these results that superpixel and RCR-based label assignment allows the obtained feature sets to be classified more effective. Beside the outperformed results of GRR compared to the GR-SVM, the GRR further provides better results compared to both RpNet-based and RCR-based methods. In comparison with the JCRC-MTL utilizing traditional feature extraction algorithms and fixed-size patch-based RCR classifier, the GRR has almost 3% and 8.5% higher OA and AA results. These results indicate that the use of deep informative features in HSI, and also superpixel-based RCR instead of patch-based RCR improve the results. Apart from the JCRC-MTL, the remainder of methods operating RpNet-based feature extraction and extra spatial filtering have lower accuracy results in comparison with the GRR. It is also observed that the use of additional filtering process on the extracted features advance the accuracy metrics. The validity of GRR is obvious in classes 2 and 15, which have relatively small local areas and few train samples. As seen, the GRR obtains 94.40% class accuracy while the others provides lower than 90% in the 2nd class. Similarly, while the proposed GRR and its SVM-based version GR-SVM obtain class accuracies greater than 90% for class 15, the remainders attain accuracies which are under this percent. As for the runtime of methods, it is monitored that all of the meth-

Table 5.3 Classification accuracy results under 0.5% labeled samples obtained by RPNNet, RPNNet-RF, RPCC, GRPC, MS-RPNNet, JCRC-MTL, SSRPNNet, GR-SVM and proposed GRR

*The highest result is bolded and the following result is underlined in each line.

Class	Train	Test	RPNNet	RPNNet-RF	RPCC	GRPC	MS-RPNNet	JCRC-MTL	SSRPNet	GR-SVM	GRR
1	71	13970	97.95	97.07	91.90	98.73	94.94	92.06	92.76	98.60	97.69
2	18	3494	88.52	87.29	68.05	68.91	58.98	62.97	61.17	88.32	94.40
3	110	21711	95.35	94.71	94.91	97.54	94.76	97.54	98.53	95.60	97.73
4	817	162468	99.20	99.58	99.81	99.48	99.05	99.50	99.47	99.24	99.92
5	32	6186	87.24	86.76	88.57	94.07	91.01	95.59	95.16	99.11	99.21
6	223	44334	95.12	95.38	94.82	96.94	94.52	97.36	97.48	96.96	98.97
7	121	23982	88.16	87.52	85.19	88.01	89.93	94.22	94.59	93.96	93.42
8	21	4033	53.16	63.64	70.74	77.81	89.11	86.55	87.19	78.38	93.18
9	55	10764	81.56	91.24	86.91	86.87	93.30	85.29	85.70	91.43	93.93
10	62	12332	81.30	79.04	90.20	84.17	89.22	89.61	90.21	94.56	92.40
11	56	10959	68.35	75.33	77.31	72.38	71.89	79.20	80.41	91.31	89.40
12	45	8909	68.03	67.10	81.40	78.35	81.27	77.21	82.77	95.33	95.47
13	113	22394	82.21	79.69	83.75	81.97	87.34	90.67	91.87	87.33	91.94
14	37	7319	78.01	86.29	87.47	83.61	93.29	74.68	77.07	91.71	89.73
15	6	996	88.65	70.88	33.03	67.03	53.31	73.73	78.46	90.06	96.78
16	37	7225	90.57	87.28	72.96	87.53	94.20	94.08	94.73	93.23	97.07
17	16	2994	57.08	76.11	81.73	82.01	89.61	88.70	88.79	88.41	99.67
18	17	3200	80.84	80.53	67.81	74.28	78.68	86.04	87.19	78.02	92.06
19	44	8668	75.44	72.30	85.92	71.92	63.64	73.60	74.05	90.62	86.13
20	18	3468	81.02	86.33	52.62	70.41	87.37	80.82	81.03	84.28	97.40
21	7	1321	62.52	64.79	79.93	73.18	99.84	93.37	93.52	92.51	99.93
22	21	4019	80.04	91.71	88.38	85.40	94.30	89.61	92.58	89.97	94.40
OA	-	-	91.25 (± 0.92)	91.96 (± 0.72)	92.07 (± 0.68)	92.54 (± 0.98)	93.12 (± 1.12)	93.98 (± 0.63)	94.46 (± 0.61)	95.73 (± 0.38)	97.08 (± 0.57)
AA	-	-	80.92 (± 2.11)	82.75 (± 1.86)	80.15 (± 1.69)	82.75 (± 1.96)	85.89 (± 2.76)	86.47 (± 0.92)	87.49 (± 2.53)	91.31 (± 1.32)	95.04 (± 0.83)
κ	-	-	88.95 (± 1.10)	89.84 (± 0.88)	89.83 (± 0.78)	90.57 (± 1.19)	91.27 (± 1.53)	92.39 (± 0.69)	92.99 (± 0.65)	94.62 (± 0.43)	96.31 (± 0.61)
Runtime (s)	-	-	103.04	73.22	1355.96	116.51	2074.68	4564.29	3261.71	361.46	520.55

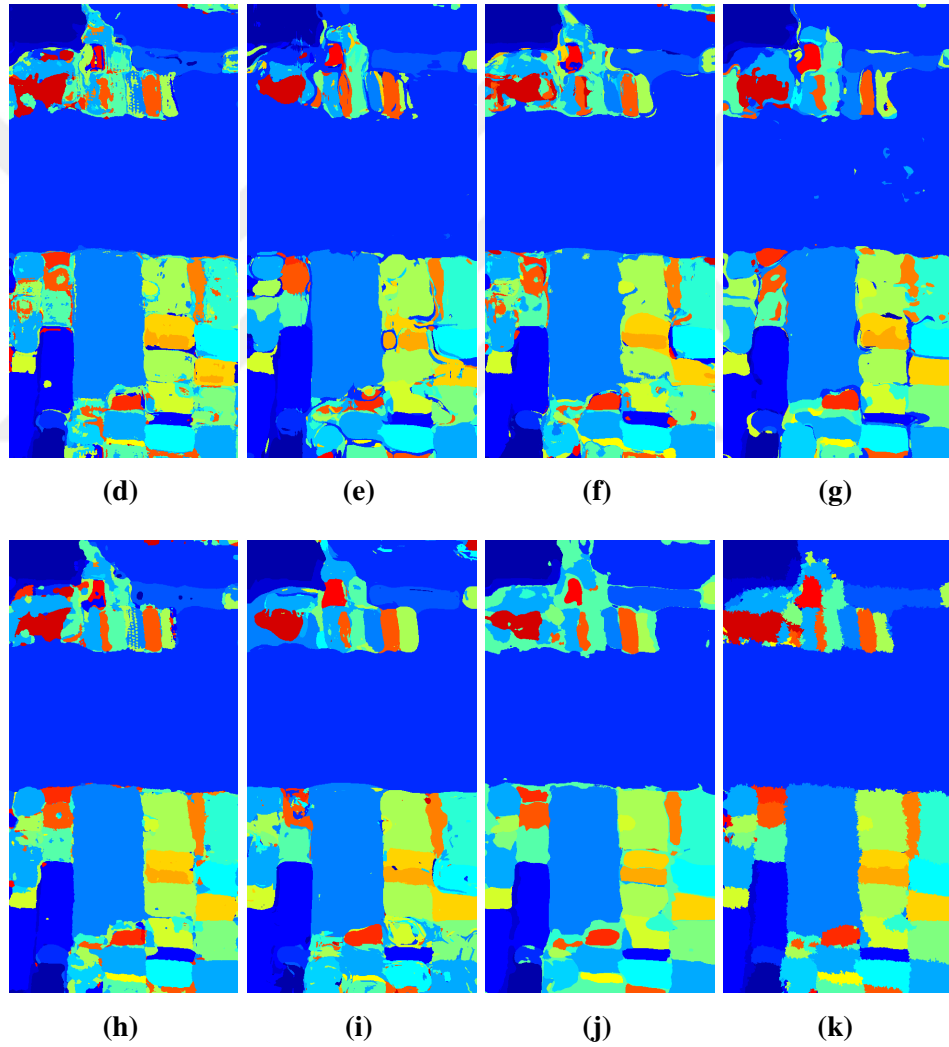
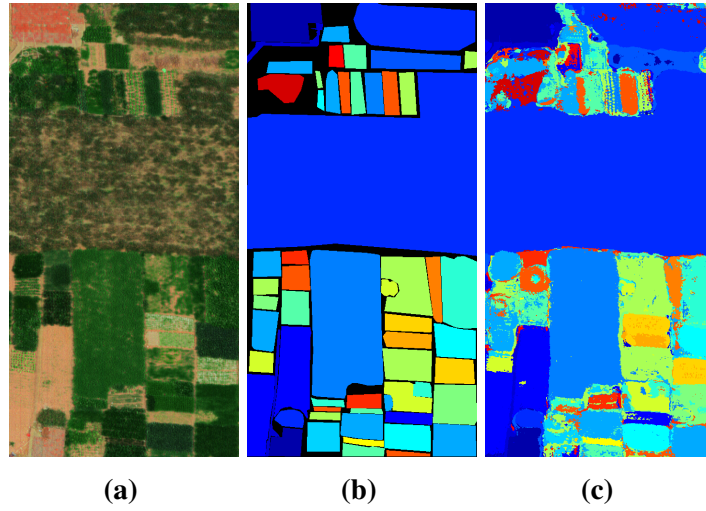


Figure 5.18 WHHH dataset: (a) False-color image, (b) Ground-truth; classification maps with OAs obtained by (c) RpNet (91.78%), (d) RpNet-RF (92.34%), (e) RPCC (92.58%), (f) GRPC (93.18%), (g) MS-RpNet (93.97%), (h) JCRC-MTL (94.51%), (i) SSRPNet (94.92%), (j) GR-SVM (95.83%) and (k) GRR (97.12%).

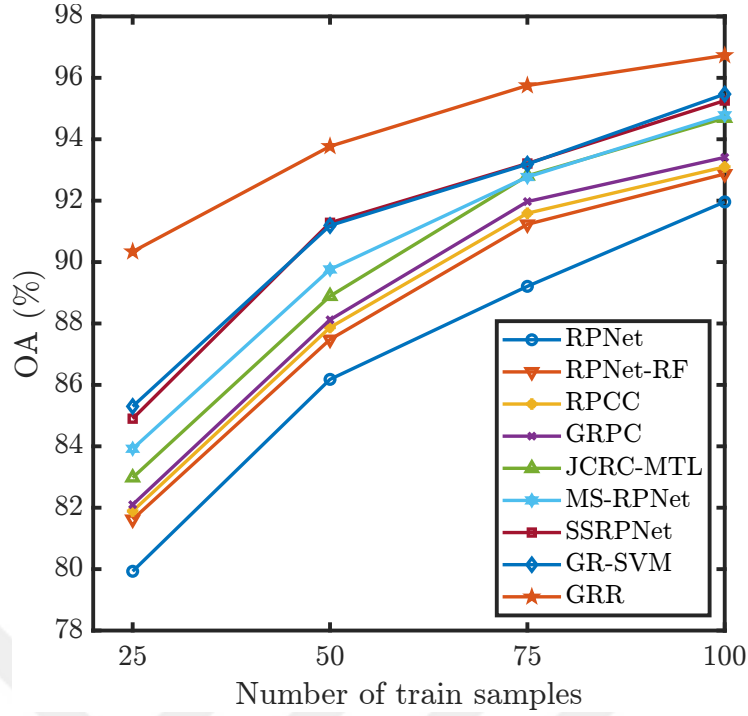


Figure 5.19 WHHH dataset: OA results of the methods RPNet, RPNet-RF, RPCC, GRPC, MS-RPNet, JCRC-MTL, SSRPNet, GR-SVM and proposed GRR under varying training samples per class.

ods have higher computation time in comparison with the IP2 and PU datasets due to the large spatial size of WHHH dataset. For instance, the patch-based RCR classifier of JCRC-MTL consumes much time due to the plenty of test samples. Similarly, the calculation of graph node weights in SSRPNet takes great deal of time. The computation of local covariances in the RPCC increases the runtime due to the many local areas. The GF process spends much time due to the calculation of graph-nodes in the proposed GRR. Among all, the lowest computation time belong to RPNet-RF followed by RPNet, respectively.

Figure 5.18 introduces the corresponding classification maps of the methods. It is seen that the maps outside of RPNet exhibit smoothed characteristics because of the additional spatial filtering steps. It verifies that the supplementary use of spatial filtering stage improves the results. While some of the methods such as RPCC, GRPC, JCRC-MTL and SSRPNet smooth the small regions more than they should, the proposed GRR does not have an over-smoothed output. Moreover, the GRR provides a classification map that is more close to the map of ground-truth. In addition, the visual outputs demonstrate that GF-based edge protection and superpixel-based RCR clas-

sifier enable the GRR to provide more correctly classified pixels and more preserved borders, at the same time.

So as to examine the achievement of methods in terms of OA result by altering the size of train samples, an additional experiment is applied which operates each of the methods under small number of available samples given as 25, 50, 75 and 100 in each class. According to the Figure 5.19, the GRR has the capacity of providing almost 5% more OA than the result of the closest methods as GR-SVM and SSRPNet under 25 samples per class, respectively. It is also observed that the GRR has approximately 3.5%, 2.5% and 1.5% more OA than the others for 50, 75 and 100 samples in each class, respectively. Therefore, it can be obviously said that the proposed GRR assigns the pixels into a class more precisely, and so acquires higher OA in comparison with the others even few train samples in each class are employed.

5.4 Summary

This chapter describes and analyzes the proposed GRR in detail. The extracted and filtered RpNet-based features are classified by superpixel-guided RCR classifier. The extracted feature sets are combined the raw HSI to keep the original information of a given HSI. Processing with GF allows to preserve the border features and details. In order to be classified under few labeled samples, pseudo-samples are produced and then these are utilized by the LDA to increase the class discrimination capability and to reduce the dimension of the feature sets. In addition, the impact of each feature is taken into the consideration by pre-classifying these with SVM where the accuracy results are converted to the weights of feature sets. Superpixel-guided RCR classifier ensures that the segmented homogeneous regions are well classified through the efficient use of feature sets. The proposed GRR is tested on small, medium and large-scale HSIC datasets. The performance of GRR is also compared to the recently proposed methods utilizing RpNet-based feature extraction or RCR-based classification. The comprehensive experiments have proven that the GRR provides higher accuracy results, maintains the borders better, and classifies the samples more accurately under few labeled samples.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

In the content of this thesis, three new spectral-spatial HSIC algorithms have been proposed. The efficacy of methods have been verified on the commonly used HSIC datasets, and also the classification performances are compared to the recent state-of-the-art HSIC methods in the literature.

In Chapter 3, an improved version of the JSRC called WLC-JSC was proposed. The assumption of equal contribution of neighbor samples around the test sample in a fixed-size patch is considered which is suggested by the JSRC. Since the patch may contain pixel/pixels belonging to another classes especially found in the class borders, the contribution of such pixel/pixels should not be assumed equal compared to the pixels with similar spectral characteristics to the test pixel. To overcome this issue, the proposed WLC-JSC applies Gaussian weighting function through each neighbor of the test sample. After calculating the specific weight of each neighbor, it is discarded or selected in accordance with the pre-determined thresholds. Beside the sample weighting, each sub-dictionary consisting train samples in a distinct class, is weighted by the Gaussian weighting function to incorporate the label-consistencies into the residual error computation. On this wise, the possibility of the test sample being assigned to the true label is improved. The experiments have been conducted on the IP1 dataset to determine the optimal parameters, and then to compare the results of proposed WLC-JSC with the different methods. The obtained results have indicated that the proposed WLC-JSC method provides higher accuracy results than the others.

In Chapter 4, the method called LCJS-WLR promoting the joint-sparse and local low-rank structure of the coefficient matrix inside a superpixel, has been introduced. Since a segmented region mostly contains pixels having similar spectral characteristics, the representation matrix of these pixels has joint-sparse structure. It means that these pixels are represented by the same atoms in the dictionary. To increase the probability of

choosing atoms which represent the samples best inside superpixel, locality-constraint has been inserted to the joint-sparse term in the optimization problem. Meanwhile, the possible spectral variations inside superpixel have been suppressed by adding low-rank term into the optimization problem. Then, its capability has been advanced by modifying the pure low-rank term with the weighted low-rank. Additionally, the impact of small sparse noise caused by the sensor measurement conditions has also been considered, and adopted to the optimization function. The proposed minimization problem has been solved by ADMM to obtain the optimal solutions of representation and noise matrices, simultaneously. The experiments carried out on the IP1 dataset have demonstrated that the LCJS-WLR provides higher accuracy results in comparison with the joint-sparse and low-rank based HSIC methods in the literature.

An efficient method called GRR using RpNet-based feature extraction and superpixel-guided RCR classifier have been proposed in Chapter 5. In order to obtain high classification accuracies, it is not sufficient to use only the raw features of HSI sometimes. In such a case, a set of feature extraction algorithms are applied to extract different feature sets. Many proposed HSIC methods implement traditional feature extractors requiring many mathematical computations and much runtime. The remainder of methods utilize deep-learning based feature extraction requiring lots of train samples and quite much computation time. Due to the RpNet-based feature extraction stage of GRR, distinct features have been obtained without any training, and also with low running time. In addition to these features, GF has also been applied to preserve the information of class borders. The extracted different feature sets are then concatenated by the raw HSI to protect original information. Among the available features, the most representatives have been selected by LDA which also increase the class discrimination. Because of the supervised structure of LDA, pseudo-samples are produced by the guidance of superpixel segmentation. In this way, the efficacy of LDA has been improved. Since the RCR classifier can incorporate the impact of feature sets via weighting mechanism, dimension-reduced feature sets are then classified by the SVM to convert the accuracy results to related feature weights. Finally, superpixel-guided RCR has been operated to classify the feature sets effectively. The extensive experiments have been performed on the IP2, PU and WHHH datasets where each has different spectral and spatial dimension such as small, medium and large-scale. The results have demonstrated that the

proposed GRR outperforms the recent methods utilizing RpNet-based feature extraction or RCR-based classification stages. Moreover, the GRR has achieved to classify samples better under few train samples compared to the other methods in the literature.

6.2 Future Work

As future works related to the proposed WLC-JSC, superpixel-guided neighborhood selection can be utilized instead of patch-based neighborhood extraction. Since the superpixel segmentation provides homogeneous regions including pixels with similar spectral characteristics, it is possible to benefit from more samples in the classification stage. While this may increase the accuracy results, meanwhile, the computation time is also reduced. For the label-consistencies of sub-dictionaries, pseudo-samples can be produced to better characterize the averaged spectrum of corresponding sub-dictionary.

A possible future work for the proposed LCJS-WLR can be segmentation of HSI into multiple-scale superpixels instead of single-scale superpixel map. This will assist the use of spatial information better at small or large regions depending on the geometrical structure of classes. Another future research can be exploring global low-rank structure of the HSI with an additional term incorporated into the optimization problem. This can reveal the underlying subspaces, therefore, more discriminative representation matrix could be obtained.

Future studies on the proposed GRR may include as follows. For example, the RpNet exploits patch-based kernels to convolve the HSI, and then to obtain the features. It should be noted that it is reasonable to construct a patch containing samples only in a single class. Because when the patch containing samples of a single class is convolved with another classes in the HSI, more distinctive features will be obtained. Therefore, the use of structure-adaptive patches can be more useful instead the fixed-size kernels. Due to the different geometrical structures of classes including small or large regions, it is also reasonable to segment the HSI into multiple-scale maps providing small or large homogeneous areas. In this way, the spatial information of HSI would be better characterized to classify the samples more accurately.

REFERENCES

- [1] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, June 2013.
- [2] P. Ghamisi, J. Plaza, Y. Chen, J. Li, and A. J. Plaza, "Advanced spectral classifiers for hyperspectral images: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 1, pp. 8–32, March 2017.
- [3] G. Camps-Valls, D. Tuia, L. Bruzzone, and J. A. Benediktsson, "Advances in hyperspectral image classification: Earth monitoring with statistical learning methods," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 45–54, January 2014.
- [4] N. Audebert, B. Le Saux, and S. Lefevre, "Deep learning for classification of hyperspectral data: A comparative review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 159–173, June 2019.
- [5] T. Ince and N. Dobigeon, "Spatial–spectral multiscale sparse unmixing for hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, October 2023, Art. no. 5511605.
- [6] G. Shaw and D. Manolakis, "Signal processing for hyperspectral image exploitation," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 12–16, January 2002.
- [7] D. Wang, Z. Chen, X. Zhang, T. Fu, R. OuYang, G. Bi, L. Jin, and X. Wang, "A high optical throughput spectral imaging technique using broadband filters," *Sensors*, vol. 20, no. 16, pp. 1–16, August 2020, Art. no. 4387.
- [8] R. N. Patro, S. Subudhi, P. K. Biswal, and F. Dell'acqua, "A review of unsupervised band selection techniques: Land cover classification for hyperspectral earth observation data," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 3, pp. 72–111, September 2021.
- [9] Y. Wang, C. M. Albrecht, N. A. A. Braham, L. Mou, and X. X. Zhu, "Self-supervised learning in remote sensing: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 4, pp. 213–247, December 2022.
- [10] P. Ghamisi, E. Maggiori, S. Li, R. Souza, Y. Tarablaka, G. Moser, A. De Giorgi, L. Fang, Y. Chen, M. Chi, S. B. Serpico, and J. A. Benediktsson, "New frontiers in spectral-spatial hyperspectral image classification: The latest advances based on mathematical morphology, markov random fields, segmentation, sparse representation, and deep learning," *IEEE Geosci. Remote Sens. Mag.*, vol. 6, no. 3, pp. 10–43, September 2018.

- [11] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using svms and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 11, pp. 3804–3814, November 2008.
- [12] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, "Spectral–spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2973–2987, August 2009.
- [13] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "A survey of active learning algorithms for supervised remote sensing image classification," *IEEE J. Sel. Top. Signal Process.*, vol. 5, no. 3, pp. 606–617, June 2011.
- [14] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral–spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, March 2018.
- [15] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, "Advances in spectral-spatial classification of hyperspectral images," *Proc. IEEE*, vol. 101, no. 3, pp. 652–675, March 2013.
- [16] M. Ahmad, S. Shabbir, S. K. Roy, D. Hong, X. Wu, J. Yao, A. M. Khan, M. Mazzara, S. Distefano, and J. Chanussot, "Hyperspectral image classification—traditional to deep models: A survey for future prospects," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 968–999, December 2022.
- [17] H. Li, J. Cui, X. Zhang, Y. Han, and L. Cao, "Dimensionality reduction and classification of hyperspectral remote sensing image feature extraction," *Remote Sens.*, vol. 14, no. 18, pp. 1–18, September 2022, Art. no. 4579.
- [18] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 3, pp. 247–259, May 2011.
- [19] J. A. Gualtieri and R. F. Crompt, "Support vector machines for hyperspectral remote sensing classification," in *Proc. SPIE 27th AIPR Workshop*, Washington, DC, USA, vol. 3584, January 1999, pp. 221–232.
- [20] G. Camps-Valls, L. Gomez-Chova, J. Munoz-Mari, J. Vila-Frances, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 1, pp. 93–97, June 2006.
- [21] W. Li and Q. Du, "Support vector machine with adaptive composite kernel for hyperspectral image classification," in *Proc. SPIE Sensing Technology and Applications*, Baltimore, MD, USA, vol. 9501, June 2015, p. 95010O.
- [22] Y. Wang and H. Duan, "Classification of hyperspectral images by svm using a composite kernel by employing spectral, spatial and hierarchical structure information," *Remote Sens.*, vol. 10, no. 3, pp. 1–26, March 2018, Art. no. 441.

- [23] C. Sun, D. Liu, J. G. Han, B. Yang, and Z. Cheng, "Hyperspectral image classification based on composite kernel relevance vector machine," *Journal of Physics: Conference Series*, vol. 1976, no. 1, pp. 1–6, July 2021.
- [24] F. A. Mianji and Y. Zhang, "Semisupervised support vector machine classification for hyperspectral imagery," in *2011 Int. Conf. Commun. and Signal Process. (ICCSP)*, Kerala, India, February 2011, pp. 320–322.
- [25] K. Tan, E. Li, Q. Du, and P. Du, "An efficient semi-supervised classification approach for hyperspectral imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 97, pp. 36–45, November 2014.
- [26] L. Yang, S. Yang, P. Jin, and R. Zhang, "Semi-supervised hyperspectral image classification using spatio-spectral laplacian support vector machine," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 651–655, March 2014.
- [27] M. S. Aydemir and G. Bilgin, "Semi-supervised classification of hyperspectral images with small sample sizes," in *24th Signal Process. Commun. Appl. Conf. (SIU)*, Zonguldak, Turkey, May 2016, pp. 681–684.
- [28] D. Tuia, F. Ratle, F. Pacifici, A. Pozdnoukhov, M. Kanevski, F. Del Frate, D. Solimini, and W. J. Emery, "Active learning of very-high resolution optical imagery with svm: Entropy vs margin sampling," in *IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Boston, MA, USA, vol. 4, July 2008, pp. IV–73–IV–76.
- [29] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, July 2009.
- [30] B. Demir, C. Persello, and L. Bruzzone, "Batch-mode active-learning methods for the interactive classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 1014–1031, March 2011.
- [31] E. Pasolli, F. Melgani, D. Tuia, F. Pacifici, and W. J. Emery, "Svm active learning approach for image classification using spatial information," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 4, pp. 2217–2233, April 2014.
- [32] U. Maulik and D. Chakraborty, "Remote sensing image classification: A survey of support-vector-machine-based advanced techniques," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 1, pp. 33–52, March 2017.
- [33] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3973–3985, October 2011.
- [34] U. Srinivas, Y. Chen, V. Monga, N. M. Nasrabadi, and T. D. Tran, "Exploiting sparsity in hyperspectral image classification via graphical models," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 3, pp. 505–509, May 2013.
- [35] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification via kernel sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 1, pp. 217–231, January 2013.

- [36] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral-spatial hyperspectral image classification via multiscale adaptive sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 12, pp. 7738–7749, December 2014.
- [37] W. Fu, S. Li, L. Fang, X. Kang, and J. A. Benediktsson, "Hyperspectral image classification via shape-adaptive joint sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 2, pp. 556–567, February 2016.
- [38] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral-spatial classification of hyperspectral images with a superpixel-based discriminative sparse model," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4186–4201, August 2015.
- [39] J. Li, H. Zhang, and L. Zhang, "Efficient superpixel-level multitask joint sparse representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 10, pp. 5338–5351, October 2015.
- [40] T. Dundar and T. Ince, "Sparse representation-based hyperspectral image classification using multiscale superpixels and guided filter," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 246–250, February 2019.
- [41] Ren and Malik, "Learning a classification model for segmentation," in *Proc. 9th IEEE Int. Conf. Comput. Vis. (ICCV)*, Nice, France, vol. 1, October 2003, pp. 10–17.
- [42] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, November 2012.
- [43] X. Kang, S. Li, and J. A. Benediktsson, "Spectral-spatial hyperspectral image classification with edge-preserving filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2666–2677, May 2014.
- [44] H. Zhang, J. Li, Y. Huang, and L. Zhang, "A nonlocal weighted joint sparse representation classification method for hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2056–2065, June 2014.
- [45] B. Tu, S. Huang, L. Fang, G. Zhang, J. Wang, and B. Zheng, "Hyperspectral image classification via weighted joint nearest neighbor and sparse representation," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 11, pp. 4063–4075, November 2018.
- [46] T. Dundar and T. Ince, "Adaptif komşuluk seçimi ve ağırlık atama yöntemleri ile hiperspektral görüntülerin sınıflandırılması," *Geomatik*, vol. 4, no. 2, p. 82–91, April 2019.
- [47] T. Dundar and T. Ince, "Spektral eşleştirme yöntemleri kullanarak hiperspektral görüntülerin seyrek gösterim tabanlı sınıflandırılması," *Geomatik*, vol. 4, no. 3, p. 215–226, December 2019.

- [48] B. Song, J. Li, M. Dalla Mura, P. Li, A. Plaza, J. M. Bioucas-Dias, J. A. Benediktsson, and J. Chanussot, "Remotely sensed image classification using sparse representations of morphological attribute profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 5122–5136, August 2014.
- [49] R. Roscher and B. Waske, "Shapelet-based sparse image representation for landcover classification of hyperspectral data," in 8th IAPR Workshop: Pattern Recognit. Remote Sens. (PRSS), Stockholm, Sweden, August 2014, pp. 1–6.
- [50] S. Jia, J. Hu, Y. Xie, L. Shen, X. Jia, and Q. Li, "Gabor cube selection based multitask joint sparse representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3174–3187, June 2016.
- [51] L. Fang, C. Wang, S. Li, and J. A. Benediktsson, "Hyperspectral image classification via multiple-feature-based adaptive sparse representation," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 7, pp. 1646–1657, July 2017.
- [52] M. Zhang, W. Li, and Q. Du, "Joint low rank and sparse representation-based hyperspectral image classification," in 8th Workshop Hyperspectral Image Signal Process. Evol. Remote Sens. (WHISPERS), Los Angeles, CA, USA, August 2016, pp. 1–4.
- [53] L. Pan, H.-C. Li, H. Meng, W. Li, Q. Du, and W. J. Emery, "Hyperspectral image classification via low-rank and sparse representation with spectral consistency constraint," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2117–2121, November 2017.
- [54] Q. Wang, X. He, and X. Li, "Locality and structure regularized low rank representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 911–923, February 2019.
- [55] Y. Ding, Y. Chong, and S. Pan, "Sparse and low-rank representation with key connectivity for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5609–5622, September 2020.
- [56] S. Yang, Y. Zhang, Y. Jia, and W. Zhang, "Local low-rank approximation with superpixel-guided locality preserving graph for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 7741–7754, August 2022.
- [57] Z. Xue, X. Nie, and M. Zhang, "Incremental dictionary learning-driven tensor low-rank and sparse representation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–19, November 2022, Art. no. 5544019.
- [58] S. Yang, Y. Zhang, Y. Ding, and D. Hong, "Superpixelwise low-rank approximation-based partial label learning for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, May 2023, Art. no. 5504905.
- [59] X. Nie, Z. Xue, C. Lin, L. Zhang, and H. Su, "Structure-prior-constrained low-rank and sparse representation with discriminative incremental dictionary for

- hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–19, January 2024, Art. no. 5506319.
- [60] W. Li and Q. Du, “A survey on representation-based classification and detection in hyperspectral remote sensing imagery,” *Pattern Recognit. Lett.*, vol. 83, no. 2, pp. 115–123, November 2016.
 - [61] W. Li, E. W. Tramel, S. Prasad, and J. E. Fowler, “Nearest regularized subspace for hyperspectral classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 1, pp. 477–489, January 2014.
 - [62] W. Li and Q. Du, “Joint within-class collaborative representation for hyperspectral image classification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2200–2208, June 2014.
 - [63] M. Xiong, Q. Ran, W. Li, J. Zou, and Q. Du, “Hyperspectral image classification using weighted joint collaborative representation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 6, pp. 1209–1213, June 2015.
 - [64] X. Chen, S. Li, and J. Peng, “Hyperspectral imagery classification with multiple regularized collaborative representations,” *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7, pp. 1121–1125, July 2017.
 - [65] X. Shen, W. Bao, H. Liang, X. Zhang, and X. Ma, “Grouped collaborative representation for hyperspectral image classification using a two-phase strategy,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, April 2022, Art. no. 5505305.
 - [66] M. Weeks and M. Bayoumi, “3d discrete wavelet transform architectures,” in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Monterey, CA, USA, vol. 4, May 1998, pp. 57–60.
 - [67] S. Jia, L. Shen, and Q. Li, “Gabor feature-based collaborative representation for hyperspectral imagery classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 1118–1129, February 2015.
 - [68] L. Shen and S. Jia, “Three-dimensional gabor wavelets for pixel-based hyperspectral imagery classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 5039–5046, December 2011.
 - [69] J. Yang and J. Qian, “A weighted multiple-feature fusion classifier for hyperspectral images with limited training samples,” *Eur. J. Remote Sens.*, vol. 51, no. 1, pp. 1006–1021, September 2018.
 - [70] B. W. Mauro Dalla Mura, Jon Atli Benediktsson and L. Bruzzone, “Extended profiles with morphological attribute filters for the analysis of hyperspectral data,” *Int. J. Remote Sens.*, vol. 31, no. 22, pp. 5975–5991, December 2010.
 - [71] S. Jia, X. Deng, J. Zhu, M. Xu, J. Zhou, and X. Jia, “Collaborative representation-based multiscale superpixel fusion for hyperspectral image classification,” *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7770–7784, October 2019.

- [72] G. Liu, L. Gao, and L. Qi, "Hyperspectral image classification via multi-feature-based correlation adaptive representation," *Remote Sens.*, vol. 13, no. 7, pp. 1–26, March 2021, Art. no. 1253.
- [73] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3681–3693, July 2015.
- [74] J. Benediktsson, M. Pesaresi, and K. Amason, "Classification and feature extraction for remote sensing images from urban areas based on morphological transformations," *IEEE Trans. Geosci. Remote Sens.*, vol. 41, no. 9, pp. 1940–1949, September 2003.
- [75] M. Yang, L. Zhang, D. Zhang, and S. Wang, "Relaxed collaborative representation for pattern classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Providence, RI, USA, June 2012, pp. 2224–2231.
- [76] Z. Wu, H. Su, X. Tao, L. Han, M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "Hyperspectral anomaly detection with relaxed collaborative representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–17, July 2022, Art. no. 5533417.
- [77] H. Su, H. Zhang, Z. Wu, and Q. Du, "Relaxed collaborative representation with low-rank and sparse matrix decomposition for hyperspectral anomaly detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6826–6842, July 2022.
- [78] J. Li, H. Zhang, L. Zhang, X. Huang, and L. Zhang, "Joint collaborative representation with multitask learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5923–5936, September 2014.
- [79] H. Su, Y. Gao, and Q. Du, "Superpixel-based relaxed collaborative representation with band weighting for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, March 2022, Art. no. 5525416.
- [80] Y. Gao, H. Su, H. Lu, and Q. Du, "Self-balancing dictionary learning for relaxed collaborative representation of hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, September 2022, Art. no. 5539918.
- [81] E. Angelopoulou, S. Lee, and R. Bajcsy, "Spectral gradient: a material descriptor invariant to geometry and incident illumination," in *Proc. 7th IEEE Int. Conf. Comput. Vis. (ICCV)*, Kerkyra, Greece, vol. 2, September 1999, pp. 861–867.
- [82] Y. Xu, B. Du, F. Zhang, and L. Zhang, "Hyperspectral image classification via a random patches network," *ISPRS J. Photogramm. Remote Sens.*, vol. 142, pp. 344–357, August 2018.
- [83] Y. Sun, Z. Fu, and L. Fan, "A novel hyperspectral image classification pattern using random patches convolution and local covariance," *Remote Sens.*, vol. 11, no. 16, pp. 1–21, August 2019, Art. no. 1954.

- [84] C. Cheng, H. Li, J. Peng, W. Cui, and L. Zhang, "Hyperspectral image classification via spectral-spatial random patches network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 4753–4764, April 2021.
- [85] Q. Shenming, L. Xiang, and G. Zhihua, "A new hyperspectral image classification method based on spatial-spectral features," *Sci. Rep.*, vol. 12, pp. 1–16, January 2022, Art. no. 1541.
- [86] D. Uchaev and D. Uchaev, "Small sample hyperspectral image classification based on the random patches network and recursive filtering," *Sensors*, vol. 23, no. 5, pp. 1–20, February 2023, Art. no. 2499.
- [87] H. Chen, T. Wang, T. Chen, and W. Deng, "Hyperspectral image classification based on fusing s3-pca, 2d-ssa and random patch network," *Remote Sens.*, vol. 15, no. 13, pp. 1–22, July 2023, Art. no. 3402.
- [88] A. Green, M. Berman, P. Switzer, and M. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Trans. Geosci. Remote Sens.*, vol. 26, no. 1, pp. 65–74, January 1988.
- [89] L. Fang, N. He, S. Li, A. J. Plaza, and J. Plaza, "A new spatial–spectral feature extraction method for hyperspectral images using local covariance matrix representation," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3534–3546, June 2018.
- [90] C. Rodarmel and J. Shan, "Principal Component Analysis for Hyperspectral Image Classification," *Surv. Land Inf. Sci.*, vol. 62, no. 2, pp. 115–122, June 2002.
- [91] Q. Du and N. H. Younan, "Dimensionality reduction and linear discriminant analysis for hyperspectral image classification," in *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst. (KES)*, Zagreb, Croatia, September 2008, pp. 392–399.
- [92] J. Zabalza, J. Ren, J. Zheng, J. Han, H. Zhao, S. Li, and S. Marshall, "Novel two-dimensional singular spectrum analysis for effective feature extraction and data classification in hyperspectral imaging," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4418–4433, August 2015.
- [93] X. Zhang, X. Jiang, J. Jiang, Y. Zhang, X. Liu, and Z. Cai, "Spectral–spatial and superpixelwise pca for unsupervised feature extraction of hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–10, February 2022, Art. no. 5502210.
- [94] Y. Dong, B. Du, L. Zhang, and L. Zhang, "Dimensionality reduction and classification of hyperspectral images using ensemble discriminative local metric learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2509–2524, May 2017.
- [95] T. Dundar and T. Ince, "Weighted and label-consistent joint sparse hyperspectral image classification," in *Proc. 13th Int. Conf. Elect. Electron. Eng. (ELECO)*, Bursa, Türkiye, November 2021, pp. 215–219.

- [96] T. Dündar and T. İnce, “Locality constraint joint-sparse and weighted low-rank based hyperspectral image classification,” in Proc. 10th Int. Conf. Recent Adv. Space Technol. (RAST), İstanbul, Türkiye, June 2023, pp. 1–6.
- [97] T. Dundar and T. Ince, “Guided filter of random patches network and relaxed-collaborative-representation-based hyperspectral image classification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 6544–6560, March 2024.
- [98] M. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, “Entropy rate superpixel segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Colorado Springs, CO, USA, June 2011, pp. 2097–2104.
- [99] Q. Liu, Z. Wu, L. Sun, Y. Xu, L. Du, and Z. Wei, “Kernel low-rank representation based on local similarity for hyperspectral image classification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 6, pp. 1920–1932, June 2019.
- [100] C. Zhang, C. Liang, L. Li, J. Liu, Q. Huang, and Q. Tian, “Fine-grained image classification via low-rank sparse coding with general and class-specific code-books,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1550–1559, July 2017.
- [101] S. Gu, L. Zhang, W. Zuo, and X. Feng, “Weighted nuclear norm minimization with application to image denoising,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Columbus, OH, USA, June 2014, pp. 2862–2869.
- [102] K. He, J. Sun, and X. Tang, “Guided image filtering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, June 2013.
- [103] Y. Shen, Y. Yu, Y. Wei, and H. Guo, “Hyperspectral image classification based on combining texture and multiscale guided filter feature of principal component,” *Infrared Phys. Technol.*, vol. 127, p. 104415, December 2022.
- [104] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, “220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3,” *Purdue Univ. Res. Repository*, vol. 10, September 2015.
- [105] Y. Tarabalka, J. Chanussot, and J. Benediktsson, “Segmentation and classification of hyperspectral images using watershed transformation,” *Pattern Recognit.*, vol. 43, no. 7, pp. 2367–2379, July 2010.
- [106] Y. Zhong, X. Hu, C. Luo, X. Wang, J. Zhao, and L. Zhang, “Whu-hi: Uav-borne hyperspectral with high spatial resolution (h2) benchmark datasets and classifier for precise crop identification based on deep convolutional neural network with crf,” *Remote Sens. Environ.*, vol. 250, p. 112012, December 2020.
- [107] F. Melgani and L. Bruzzone, “Classification of hyperspectral remote sensing images with support vector machines,” *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, August 2004.

- [108] J. Zou, W. Li, and Q. Du, “Sparse representation-based nearest neighbor classifiers for hyperspectral imagery,” *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2418–2422, December 2015.
- [109] B. Tu, X. Zhang, X. Kang, G. Zhang, J. Wang, and J. Wu, “Hyperspectral image classification via fusing correlation coefficient and joint sparse representation,” *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 3, pp. 340–344, March 2018.
- [110] Y.-f. Jia, Y. Tian, Y.-j. Li, and P.-b. Fu, “Exponential discriminative locality alignment for hyperspectral image classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 1, pp. 33–37, January 2017.
- [111] Q. Liu, Y. Sun, R. Hang, and H. Song, “Spatial–spectral locality-constrained low-rank representation with semi-supervised hypergraph learning for hyperspectral image classification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 9, pp. 4171–4182, September 2017.
- [112] H. Yu, L. Gao, W. Li, Q. Du, and B. Zhang, “Locality sensitive discriminant analysis for group sparse representation-based hyperspectral imagery classification,” *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 8, pp. 1358–1362, August 2017.
- [113] Y. Ding, Z. Zhang, X. Zhao, Y. Cai, S. Li, B. Deng, and W. Cai, “Self-supervised locality preserving low-pass graph convolutional embedding for large-scale hyperspectral image clustering,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, August 2022, Art. no. 5536016.
- [114] P. V. Giampouras, K. E. Themelis, A. A. Rontogiannis, and K. D. Koutroumbas, “Simultaneously sparse and low-rank abundance matrix estimation for hyperspectral image unmixing,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4775–4789, August 2016.
- [115] L. Sun, F. Wu, T. Zhan, W. Liu, J. Wang, and B. Jeon, “Weighted nonlocal low-rank tensor decomposition method for sparse unmixing of hyperspectral images,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1174–1188, March 2020.
- [116] H. Su, C. Jia, P. Zheng, and Q. Du, “Superpixel-based weighted collaborative sparse regression and reweighted low-rank representation for hyperspectral image unmixing,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 393–408, December 2022.
- [117] R. Zhang, L. Yang, G. Liu, and X. Feng, “Weighted nuclear norms of transformed tensors for nonlocal hyperspectral image denoising,” *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, June 2022, Art. no. 6011305.
- [118] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, January 2011.
- [119] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, July 2009.

- [120] L. Pan, H.-C. Li, and X.-D. Chen, "Locality constrained low-rank representation for hyperspectral image classification," in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS), Beijing, China, July 2016, pp. 493–496.
- [121] S. Huang, H. Zhang, W. Liao, and A. Pižurica, "Robust joint sparsity model for hyperspectral image classification," in Proc. IEEE Int. Conf. Image Process. (ICIP), Beijing, China, September 2017, pp. 3130–3134.
- [122] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 7, no. 6, pp. 2094–2107, June 2014.
- [123] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," IEEE Trans. Geosci. Remote Sens., vol. 54, no. 10, pp. 6232–6251, October 2016.
- [124] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," IEEE Geosci. Remote Sens. Mag., vol. 5, no. 4, pp. 8–36, December 2017.
- [125] M. Caywood, B. Willmore, and D. Tolhurst, "Independent components of color natural scenes resemble v1 neurons in their spatial and color tuning," J. Neurophysiol., vol. 91, no. 6, pp. 2859–73, July 2004.
- [126] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," IEEE Trans. Geosci. Remote Sens., vol. 47, no. 3, pp. 862–873, March 2009.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: DÜNDAR, Tuğcan

EDUCATION

Degree	Institution	Department	Graduation Year
Ph.D.	Gaziantep University	Electrical-Electronics Eng.	2024
M.Sc.	Gaziantep University	Electrical-Electronics Eng.	2019
B.Sc.	Gaziantep University	Electrical-Electronics Eng.	2016

WORK EXPERIENCE

Year	Place	Enrollment
2021–Present	Gaziantep University	Lecturer
2017–2019	Gaziantep University	Research Assistant
2016–2017	Adana Science and Technology University	Research Assistant

PUBLICATIONS

- 1) **T. Dündar** and T. Ince, "Guided filter of random patches network and relaxed collaborative representation based hyperspectral image classification," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 17, pp. 6544-6560, March 2024.
- 2) T. Ince, **T. Dündar**, S. Kaçmaz and H. Karci, "Superpixel weighted low-rank and sparse approximation for hyperspectral unmixing," in Proc. 10th Int. Conf. Recent Adv. Space Technol. (RAST), Istanbul, Türkiye, June 2023, pp. 1-5.
- 3) **T. Dündar** and T. Ince, "Locality constraint joint-sparse and weighted low-rank based hyperspectral image classification," in Proc. 10th Int. Conf. Recent Adv. Space Technol. (RAST), Istanbul, Türkiye, June 2023, pp. 1-6.

4) **T. Dundar** and T. Ince, "Weighted and label-consistent joint sparse hyperspectral image classification," in Proc. 13th Int. Conf. Elect. Electron. Eng. (ELECO), Bursa, Turkey, November 2021, pp. 215-219.

5) **T. Dünder** ve T. İnce, "Spektral eşleştirme yöntemleri kullanarak hiperspektral görüntülerin seyrek gösterim tabanlı sınıflandırılması", Geomatik, c. 4, sy. 3, ss. 215–226, Aralık 2019.

6) T. Ince and **T. Dundar**, "Simultaneous nonconvex denoising and unmixing for hyperspectral imaging," IEEE Access, vol. 7, pp. 124426-124440, August 2019.

7) **T. Dünder** ve T. İnce, "Adaptif komşuluk seçimi ve ağırlık atama yöntemleri ile hiperspektral görüntülerin sınıflandırılması", Geomatik, c. 4, sy. 2, ss. 82–91, Ağustos 2019.

8) **T. Dundar** and T. Ince, "Sparse representation-based hyperspectral image classification using multiscale superpixels and guided filter," IEEE Geosci. Remote Sens. Lett., vol. 16, no. 2, pp. 246-250, February 2019.