University of Strathclyde Glasgow

Multi-modality Feature Fusion and Unsupervised Hyperspectral Band Selection for Effective Classification of Remote Sensing Images

by

He Sun

Center for Signal and image processing

Department of Electronic and Electrical Engineering

University of Strathclyde, Glasgow

A thesis submitted in the fulfillment for the degree of

Doctor of Philosophy

May 28, 2020

I would like to dedicate this thesis to my greatest motherland, China, my dearest parents, and my beloved her.

Declaration of Authorship

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

He Sun

March. 28, 2020.

Acknowledgements

Finally, it comes to the end of my PhD study. During the last three years, many people have helped me from both personal and academic views. I would like to use this opportunity to thank them.

Firstly, I would like to thank my first supervisor Dr Jinchang Ren. I could never forget the first phone call I made with him, when he showed great trust on me and gave me the opportunity to pursue a PhD. He taught me how to do research and introduced me into the academic community. Most importantly, I really thank him to show me that always treat the research with a positive attitude. I have learnt a lot from him, not only in the academic field, but also in the real life. I would also like to thank Dr Ren and the University of Strathclyde for giving me financial support, which makes my PhD easier. Besides, I would also like to thank my second supervisor, Prof. Stephen Marshall, all my publication co-authors, and my viva examiners, for their useful comments and suggestions.

I wish to thank all the colleagues, staffs, and visitors in the CeSIP group for their help. In particular, I would like to thank Mr Zhenyu Fang for his support in deep learning, I will never forget those interesting discussions we have had during lunch and coffee breaks. I would also like to thank Dr Yijun Yan for introducing me into the group when I started my PhD and taking care of me as a senior. Besides, I would like to thank all the Chinese members in our group for

numerous gathering parties together. Thanks must go to the group in the HSI Lab, TIC building, including Dr Jaime Zabalza, Dr Julius Tschannerl, Dr Andrew Young, Mr Fraser Macfarlane, Mr Ha Viet Khang, and Mr Calum Maclellan. I thank you all for helpful discussions within the lab and enjoyable moments in many occasions.

I would also like to thank all my friends for their kindly support during my PhD. I would like to thank my best friend in Glasgow, Mr Jinkai Zhang. He always gives me the best suggestion whenever I have problems in my study or life. I wish you have a wonderful life in London. Special thanks to all my friends in Daxing, for treating me warmly every time whenever I go back home.

Although most of my relatives in my large family, especially elders, could not read these English words, I would like to thank all of them, including my grandparents, my uncles, my aunts, my cousins, and my two lovely nieces on both my parents' sides, for all their love during my PhD. Particularly, I would like to dedicate this thesis to my grandfather on my father's side, who has passed away during my PhD, and I will always miss him.

My deepest gratitude goes to my family. I would like to thank my parents for their constant love and trust. I really thank them to send me to U.K. even they could not speak English. I wish to use this opportunity to thank my girlfriend' parents, for their support and trust during my PhD. Last, but not least, I want to express my gratitude to my girlfriend, Chufei. I could not finish my PhD without your encouragement in the very beginning, and your distant love and motivation throughout my PhD. I love you, now and always.

Abstract

In recent years, Hyperspectral image (HSI) has been widely applied in a range of applications due to its contained rich spectral information. As a fundamental topic in HSI analysis, HSI classification has attracted increasing attention. An effective classification algorithm without too much computational cost is always desired, especially under the circumstance of insufficient training samples. As a result, this thesis aims to design and implement novel techniques to reduce the high dimensionality of the HSI data and improve the classification performance with limited training samples.

In this thesis, first, a superpixel-based feature specific sparse representation framework (SPFS-SRC) is proposed for spectral-spatial classification of HSI at superpixel level, which can improve the classification performance with less training samples and better efficacy. The proposed online learning strategy can better reflect the effect of each extracted feature. Second, a superpixel-based multiple feature fusion framework has been developed to generate an effective fused feature with a reduced dimension.

In addition, two novel methods are also proposed for unsupervised band selection for dimensionality reduction in HSI. First, an adaptive distance enabled tree-based band hierarchy framework (ADBH) has been developed to obtain desired band subset of the HSI, which can help to avoid the noisy bands. With

the proposed tree hierarchy-based framework, any number of band subset can be acquired. By introducing a novel adaptive distance into the hierarchy, the similarity between bands and band groups can be computed straightforward whilst reducing the effect of noisy bands. Furthermore, a deep learning-based framework has been designed to determine the optimal band subset by utilizing the concrete autoencoder (CAE). The band subset with the most information can be chosen as the desired result. For performance evaluation, several remote sensing HSI datasets have been utilized to evaluate the proposed algorithms, where improved performance has proved the superiority of proposed methodologies.

In summary, the outcome of this thesis make contributions in the HSI community by proposing two multi-modality feature fusion algorithms and two unsupervised band selection methods for the effective dimensionality reduction and data classification in HSI, the novelty and robustness of the proposed technologies have been fully demonstrated by extensive experiments. Relevant approaches also have great potential to be applied in other signal and image analysis tasks, especially dimensionality reduction, data fusion and data classification.

Contents

Abbreviations

AA–Average accuracy

ADBH–Adaptive Distance based Band Hierarchy

AE–Autoencoder

ASPS–Adaptive Subspace Partition Strategy

AVIRIS–Airborne Visible Infrared Imaging Spectrometer

CAE-UBS–Concrete AE-UBS

CCA–Canonical Correlation Analysis

CBS–Constrained Band correlation Strategy

CK-SVM–Composite Kernel SVM

CNN–Convolutional Neural Network

DCCA–Dual-clustering based band selection by Context Analysis

DSEBS–Band Selection with Dominant Set Extraction

EDBH–Euclidean Distance based Band Hierarchy

E-FPDC–Enhanced FPDC

EMP–Extended morphological profile

FPDC–fast-peak-based clustering

HSI–Hyperspectral Image

ICA–Independent Component Analysis

JSRC–Joint SRC

JKSRC–Jont Kernelized SRC

KSC–Kennedy Space Center

KNN–K-Nearest Neighbourhood

LEGO–LogDet Extract Gradient Online

LiDAR–Light Detection and Ranging

LSC–Linear spectral clustering

LMCCA–Localized MCCA

MASR–Multiscale Adaptive SRC

MCCA–Multiple CCA

MFASR–multiple feature adaptive SRC

MNF–Maximum Noise Fraction

MRF–Markov Random Field

MTSP–Multitask sparsity pursuit

MSE–Mean Squared Error

MVPCA–Maximum-variance PCA

NP–Non-deterministic polynomial-time

OA–overall accuracy

OCF–Optimal clustering framework

OMP–orthogonal matching pursuit

PCA–Principal Component Analysis

PaviaU–Pavia University

ROSIS–Reflective Optics System Imaging Spectrometer

Salinas–Salinas scene

SC-MK–superpixel-based classification framework with multiple kernels

SLIC–simple linear iterative clustering

SRC–Sparse Representation Classification

SPSRC–Superpixel-based SRC

SPFS-SRC–superpixel-based Feature Specific SRC

SMFF-SRC–Superpixel-based Multiple Feature Fusion SRC

SVM–Support Vector Machine

UBS–Unsupervised Band Selection

UH–University of Houston

VGBS–volume gradient band selection

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Research Motivation

With rich spectral information contained in tens or hundreds of spectral bands, hyperspectral images (HSI) have been successfully applied in a wide range of applications [1–3], especially in the remote sensing area, such as land cover analysis [4–6], military surveillance [7, 8], object detection [9], image enhancement [10, 11], and precision agriculture [12–19], etc. Among these applications, image classification is an active topic, which aims to assign each pixel in the HSI into one unique semantic category or class. Generally, the performance of HSI classification is determined by several important issues: the extracted features, the number of training samples, the number of dimensions of the extracted features, and the classifier structure, etc. In this thesis, the number of training samples and the number of dimensions of the extracted features are taken into consideration. With training samples, the designed classifier can better detect the discriminative ability between different semantic classes. However, the number of training samples is generally not enough in practical applications [20].

To address the problem of insufficient training samples, many researchers have found that the classification accuracy can be significantly improved by extracting spatial features from the HSI [21–24], such as the morphological features [21–24], the texture features [25], etc. Furthermore, the availability of light detection and ranging (LiDAR) data could provide elevation information, which helps to distinguish objects with different height. For the classification task, it is necessary to fuse these above mentioned features in an effective way.

During the past few years, a number of techniques have been proposed to fuse the spectral features from the HSI and other features [26–28], such as spatial feature, texture feature, etc. Basically, the simplest strategy is to stack the spectral feature and other extracted features into one feature, and then apply some effective classifiers, such as support vector machine (SVM) [29], random forest [30], Markov random field and its variants [31, 32], etc. To improve the classification performance, Li et al. [29] have proposed a generalized composite kernel framework based on the SVM. By combining both the spectral and spatial information contained in the HSI, the generalized composite kernels have been constructed to investigate the flexibility between the different features without introducing any weight parameters. In [33], a spectral-spatial framework was built by jointly applying the loopy belief propagation and active learning strategy. The loopy belief propagation has been employed to calculate the conditional marginal. With the aid of the estimated marginal, the active learning algorithm [33] is utilized to exploit more spectral and spatial information from the HSI data. Therefore, it is necessary to extract more useful features from the HSI data to address the problem of insufficient training samples.

For the HSI image, the numerous bands contain sufficient spectral information,

which enable material identification and object detection, the processing of HSI suffers from the "curse of dimensionality" [34]. Besides, there are redundant bands in the HSI, which may lower the efficiency of data analysis. Moreover, due to the high dimensionality of the HSI, the computational burden is huge. To tackle these problems, it is crucial to reduce the dimensionality of the HSI data whilst preserving the useful spectral information.

Basically, there are two kinds of dimensionality reduction methods for HSI: feature extraction and feature selection. With the feature space transform, feature extraction can project the original data into a lower dimensional space, using approaches such as the principal component analysis (PCA) [35, 36], independent component analysis (ICA) [37], wavelet transform [16], the manifold learning [38], and the maximum noise fraction (MNF) [39], etc. The resulting data can be assumed to contain most of the spectral and spatial information from the original HSI data. Although the feature extraction methods successfully reduce the dimensionality of HSI whilst keeping the discriminative ability, the feature transform itself relies on the whole set of original data and often has poor correspondence to the process of optical acquisition of the data. In contrast, the feature selection method, which is also called band selection in the HSI context, can select an optimised subset from the HSI data, based on their dominant contributions to certain tasks. Since the band selection methods can maintain the physical acquisition characteristic of raw data and solve the high dimensionality problem simultaneously, an efficient band selection method is often preferred.

Generally, based on the availability of the class label information, existing band selection methods can be divided into two groups: i.e. supervised [40–42] and unsupervised ones [48–59]. Supervised methods can construct a criterion with

the label information of pixels aiming to improve the class separability. In [40], the desired band subset is chosen based on the class-based spectral signatures. By extracting two most distinctive bands whose dissimilarity is the largest among all bands, other bands can be chosen iteratively by minimizing the estimated abundance covariance from each pixel along with the class information. Cao *et al.* [41] proposed another wrapper-based supervised band selection method, where the chosen band subset is determined based on minimizing the defined local smoothness with the aid of the classification map from a Markov random field (MRF) classifier. To improve the reliability of the local smoothness generated from the classification map, the wrapper method is utilized to initialize the designed method. In [42], Patra *et al.* developed a rough-set-based supervised band selection method. The rough-set theory is applied to compute the relevance and significance of each band by using the class information as a prior knowledge, and bands with higher relevance and significance are chosen to form the band subset.

Although the band subset acquired by the supervised methods can achieve better classification performance, the selected bands are often affected by the chosen training samples where different training samples may lead different band selection results. Furthermore, these approaches can become less effective in practical applications if sufficient training samples with label information are not approachable. Even though some supervised band selection methods only choose few training samples, the classification performance with less band and less training samples are not reliable as a criterion for band selection. Therefore, we will focus on the unsupervised band selection (UBS) methods in this thesis.

1.2 Original Contributions

In this thesis, several novel methods for hyperspectral remote sensing feature fusion with insufficient training samples and unsupervised band selection are proposed. By utilizing less training samples or choosing less spectral bands, these proposed methods aim to improve the classification performance with an efficient computational complexity, which has the potential to employ in real applications. More specifically, the contributions of this thesis are listed as follows:

1) To improve the efficacy of the sparse representation classification (SRC), a superpixel-based feature specific sparse representation framework (SPFS-SRC) has been proposed for spectral-spatial classification of hyperspectral images (HSI) at superpixel level. The classification is significantly improved by utilizing the online metric learning strategy whilst the computational burden is reduced with the proposed superpixel framework. The contribution is summarized in Chapter 3 and Paper 1 in Appendix A.1

2) A superpixel-based multiple feature fusion SRC (SMFF-SRC) approach is proposed to improve the classification performance with insufficient training samples. Multiple features are combined by the canonical correlation analysis (CCA)-based fusion mechanism, where both the efficacy and efficiency are improved with the superpixel preprocessing step. The contribution is summarized in Chapter 3 and is under preparation, which can seen seen in Paper 1 in Appendix A.3.

3) An adaptive distance based band hierarchy (ADBH) clustering framework is proposed for unsupervised band selection in HSI, which can help to avoid

the noisy bands whilst reflecting the hierarchical data structure of HSI. With a tree hierarchy-based framework, we can acquire any number of band subset. By introducing a novel adaptive distance into the hierarchy, the similarity between bands and band groups can be computed straightforward whilst reducing the effect of noisy bands. The contribution is summarized in Chapter 4 and Paper 2 in Appendix A.2.

4) A concrete autoencoder(AE)-based UBS framework is proposed (CAE-UBS) for the HSI, which enables effective learning by introducing the concrete random variables and the reconstruction loss. With a trained autoencoder, the framework can provide potential band subsets and the optimal one can be determined through the trained decoder. The idea of the CAE-UBS is quite straightforward without designing any complicated strategies or metrics. By utilizing the reconstruction loss, the whole training process can be supervised in an unsupervised manner. The contribution is summarized in Chapter 5 and is under preparation, which can be seen in the Paper 2 in Appendix A.3.

1.3 Thesis Organisation

The remaining part of this thesis are divided into the following five chapters.

Chapter 2 reviews the related work in HSI and provides the theoretical background of proposed methods.

Chapter 3 first introduces the motivation of the SPFS-SRC and SMFF-SRC, followed by the detail discussions of the proposed frameworks. With the implemented SPFS-SRC algorithm, the classification result can be obtained by com-

bining the extracted spatial and spectral features. Experimental results of SPFS-SRC on two popular HSI datasets have demonstrated the efficacy of the proposed methodology. By proposing a novel multiple CCA(MCCA)-based method in the SMFF-SRC, the extracted features can be fused into one discriminative feature, which increase the classification accuracy and reduce the dimensions of features. The efficacy of the proposed SMFF-SRC methodology has been demonstrated on three HSI datasets.

Chapter 4 proposes the ADBH framework for the UBS task in HSI. The motivation of the designed framework is introduced first. Then, the proposed framework is presented. The performance of this framework is validated by applying the band selection result on classification of four publicly available HSI datasets.

Chapter 5 introduces a novel UBS framework with the autoencoder (CAE-UBS) based deep learning. The motivation of the designed framework is introduced first. Afterwards, the proposed deep learning model is presented in detail. Similarly, the performance of this framework is validated by applying the band selection result on classification of four HSI datasets.

Chapter 6 gives the conclusion of this thesis and also discusses the future perspectives as about how to further improve the work.

Chapter 2

Background and Related Work

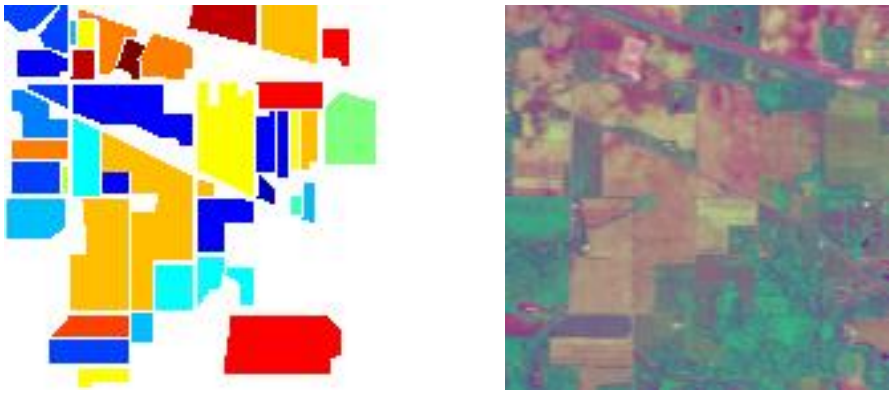
2.1 Introduction

In this chapter, first, a brief introduction of HSI and some commonly used remote sensing datasets are given. Afterwards, a literature review of the SRC-based HSI classification is presented in Section 2.3. Then, the relevant work of UBS of HSI is reviewed in Section 2.4. Section 2.5 gives detailed descriptions of utilized techniques, including the SRC, CCA, and AE.

2.2 HSI remote sensing datasets

The HSI remote sensing dataset is usually acquired by collecting the information of ground objects in distance. The acquired HSI datasets can be utilized for a range of applications, such as classification, detection, recognition, super-resolution, etc.

To evaluate the performance of the proposed methods in this thesis, five HSI datasets from three imaging systems have been used. The first one is the Indian

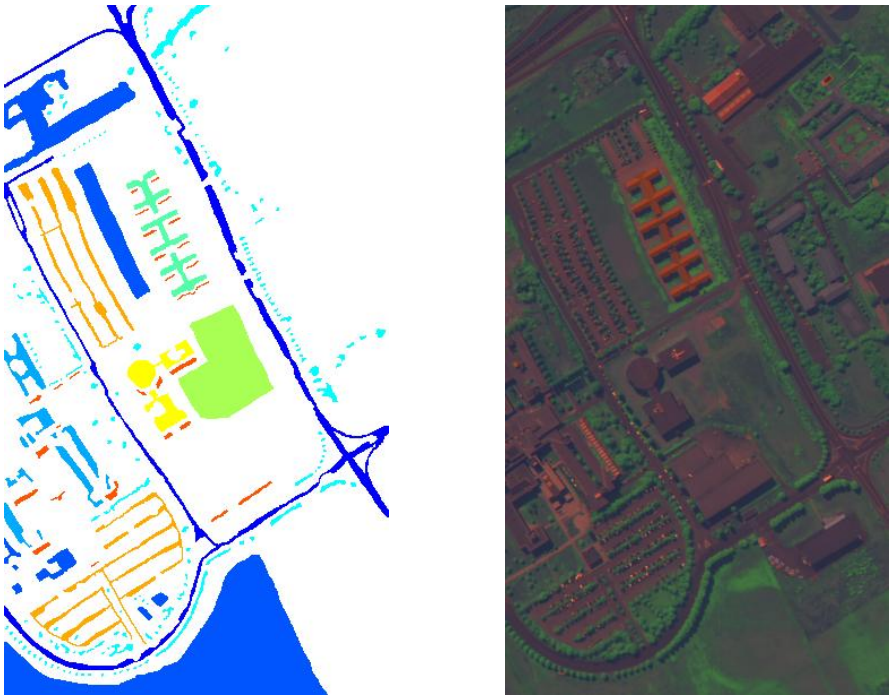


(a) (b)

Figure 2.1: (a) the ground truth of the Indian Pine dataset, (b) a false-color image of the Indian Pine dataset generated from PCA.

pines dataset [43], which was collected by the Airborne Visible Infrared Imaging Spectrometer (AVIRIS) sensor over the agricultural experimental field located at North-Western, Indiana, USA in 1992. The original dataset has 224 spectral bands ranging from 0.4 to 2.5 μm with 16 manually labelled class, and its spatial size is 145×145 pixels with 10249 labelled pixels. After the removal of 24 water absorption bands, the rest 200 bands are utilized for band selection and data classification. The ground truth of the Indian Pine dataset and its corresponding false-color image are shown in Figure 2.1.

The second dataset is the Pavia University (PaviaU), which was captured by the Reflective Optics System Imaging Spectrometer (ROSIS) system over the campaign of the university of Pavia, Italy in 2002 [44]. The PaviaU dataset has a spatial size of 610×610 pixels and 103 spectral reflectance bands with the spectral range from 0.43 to 0.86 μm. A cropped image of 610×340 pixels are employed after discarding pixels with no information. In the PaviaU dataset, 42776 pixels from 9 semantic classes are labelled. The ground truth of the PaviaU dataset and its corresponding false-color image are shown in Figure 2.2.

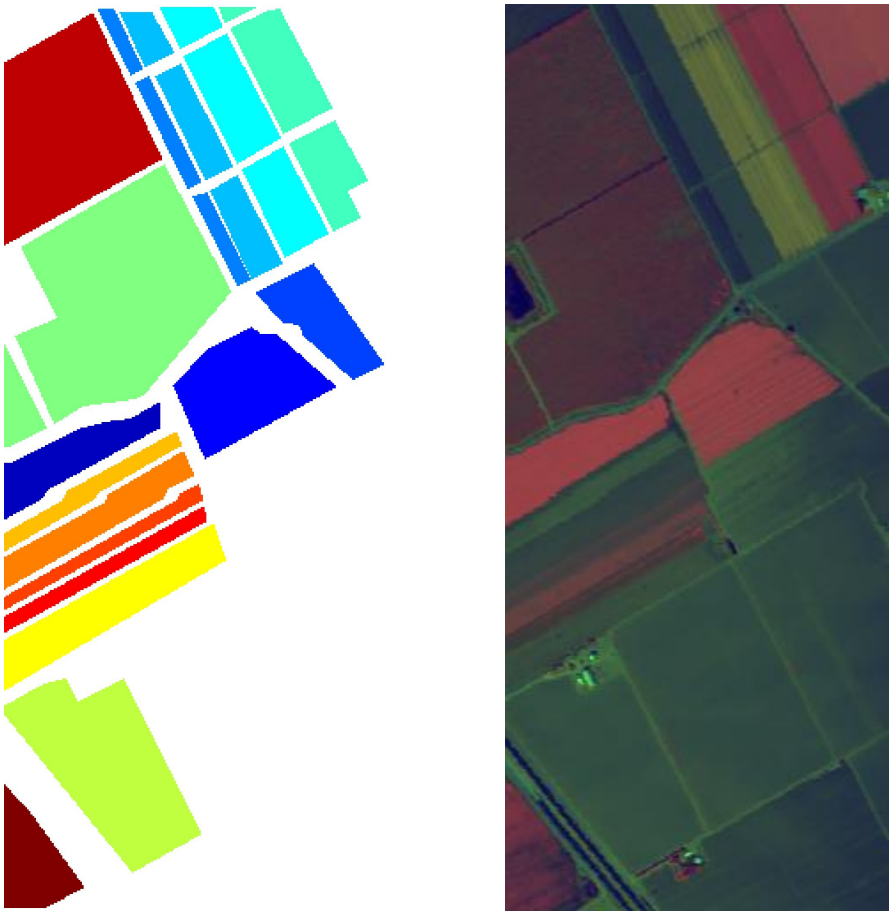


(a) (b)

Figure 2.2: (a) the ground truth of the PaviaU dataset, (b) a false-color image of the PaviaU dataset generated from PCA.

The third dataset is the Salinas scene (Salinas), which was also captured by the AVIRIS in Salinas Valley, California, USA in 1998 [45]. Same as the Indian pines dataset, the Salinas dataset collects spectral information within 0.4-2.5 μm in 224 bands. Its ground truth data also has 54129 labelled pixels from 16 classes and its image spatial size is 512×217 pixels. Similar to the Indian pines dataset, the Salinas dataset in experiments also has 20 water absorption bands removed with the rest 204 bands for analysis. The ground truth of the Salinas dataset and its corresponding false-color image are shown in Figure 2.3.

The fourth dataset is the Kennedy Space Center (KSC) dataset [46], which was obtained using the same AVIRIS sensor in Florida, USA, 1996. By removing the water absorption and low SNR bands, only 176 bands are used with 13 labelled

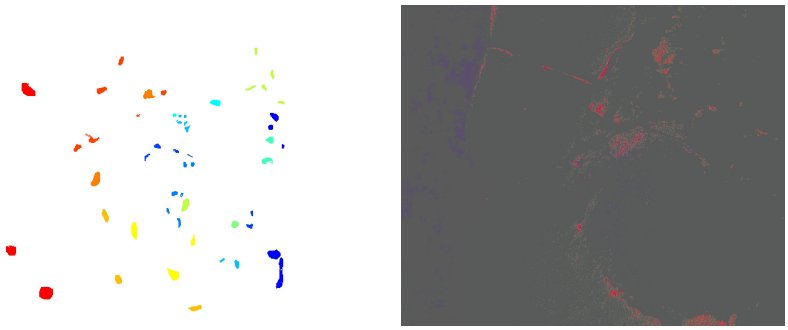


(a) (b)

Figure 2.3: (a) the ground truth of the Salinas dataset, (b) a false-color image of the Salinas dataset generated from PCA.

classes, and the spatial size of this dataset is 512×614 pixels and 19035 pixels are manually labelled. The ground truth of the KSC dataset and its corresponding false-color image are shown in Figure 2.4.

The last one is the University of Houston dataset (UH) [47], which includes two source data, an HSI image and pseudo waveform LiDAR. The UH dataset was captured over the campus of the University of Houston and its neighbouring area by the ITRES-CASI 1500 sensor. The HSI data consists of 144 spectral bands ranging from 380 to 1050 nm and its spatial size is 349×1905. The LiDAR data



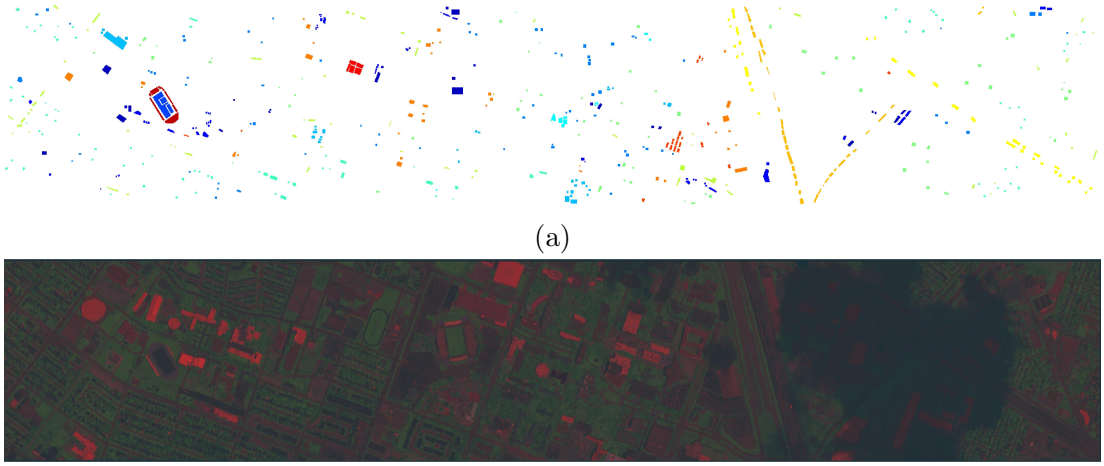
(a) (b)

Figure 2.4: (a) the ground truth of the KSC dataset, (b) a false-color image of the PaviaU dataset generated from PCA.

were acquired by an Optech Gemini 280 sensor and then coregistered to HSI. This dataset includes 15 semantic classes and 15029 sparsely labelled samples. The ground truth of the UH dataset and its corresponding false-color image are shown in Figure 2.5.

2.3 Sparse Representation-based HSI classification

Due to its simple mathematical principle, the sparse representation classification (SRC)-based method has become a powerful tool in the computer vision community [60–62], especially in the HSI classification task [26–28, 63–67]. Firstly, Wright et al. applied the sparse representation on the facial recognition task [60], where the estimated sparse representation coefficients can represent the discriminative ability. For the HSI classification, each test pixel of the HSI can be reconstructed by choosing the given number of training samples from a built dictionary. The correlation between the test pixel and the selected training samples can be determined by the sparse coefficients. After that, the test pixel can be



(a)

(b)

Figure 2.5: (a) the ground truth of the UH dataset, (b) a false-color image of the UH dataset generated from PCA.

assigned into the class with the minimum reconstruction error. In [63], a Joint SRC (JSRC) has been proposed to incorporate more spatial information with the spectral features. In JSRC, a fixed-size local window around the test pixel is predefined, all the pixels within this window can be assumed to be in one semantic class and thus have a common sparse representation. With the developed JSRC model, the spectral-spatial information can be explored by adding the spatial correlation with the test pixel. To address the non-linear discriminative ability of the proposed model, the corresponding kernel-based JSRC (JKSRC) has also been proposed [64]. Similar to JSRC, a nonlocal weight between the neighbouring pixel and the text pixel is applied as a regularization term of the SRC [65], where the spatial correlation can be better explored. Furthermore, the multiscale adaptive SRC (MASR) [66] has been proposed to combine the spatial information from different sized regions. Although the above JSRC-based methods can achieve a good performance, the optimal sized region is difficult to find, and the computational burden is rather high as a result of the spatial correlation.

For the HSI data, the high dimensionality of the pixel vector often leads to huge computational burden. For SRC-based framework, the computational complexity can be even higher due to the large size of the dictionary to be constructed from the training samples in most circumstance. Thus, it is crucial to improve the efficiency of the SRC while maintaining the classification accuracy. Within the aforementioned methods, the spatial information of HSI is usually extracted from a fixed-size window or multiscale square windows, which also increases the computational burden. Recently, the utilization of superpixel [68] and other shape-adaptive filters [69] are used to find the homogeneous regions instead of square windows. In [70], Superpixel-based classification framework with multiple kernels (SC-MK) has been designed, and the experimental results indicate the efficacy of the approach. The superpixel-based SRC method [71] has also shown the superiority in terms of high classification accuracy and efficient computational speed.

2.4 HSI band selection

In the last two decades, a number of approaches have been proposed for unsupervised band selection (UBS) in HSI. In this section, some typical USB approaches from the aforementioned two groups, i.e. the ranking-based and the clustering-based methods, will be reviewed, and relevant analysis to motivate the proposed work is also given.

As mentioned in the last section, the goal of the ranking-based UBS methods is to find the most significant bands among the HSI data. To fulfil this purpose, an effective criterion for estimating the importance of each band is essential. With

the aid of the designed criterion, most representative bands can be determined. In [48], a PCA-based band selection criterion was proposed. By applying the maximum-variance PCA (MVPCA), the band prioritization can be estimated according to the eigenanalysis. A defined load factor of each band can be obtained from the consolidation of eigenvalue and eigenvector. For each band, a variance-based band power ratio is utilized to represent its discriminative ability, which is accessed by using the variance of each band to divide that of all bands. By finding the bands with higher ratio, a band subset is determined. Although the chosen bands are more representative and more discriminative, the correlation between those bands are ignored in the MVPCA. The robustness of the selected bands is not guaranteed as they are with higher variance. Chang and Wang [49] have presented a constraint band correlation strategy (CBS), which is derived from the idea of constrained energy minimization. By defining a finite impulse filter between each band and the whole dataset, the correlation can be represented by a minimized vector. After discarding bands with high correlation, the remaining bands are selected, which can be more robust to the noisy band.

Different from the ranking-based methods, clustering-based methods can naturally reduce the correlation between chosen bands. In these approaches, the HSI bands are sequentially grouped into different clusters by a defined criterion. Afterwards, typical bands from each cluster are selected to form the desired band subset. Since the band subset comprises bands from different clusters, the high correlation between bands can be avoided. In [50], a hierarchical clustering (WaLuDi/WaLuMi) is applied to divide bands of whole dataset into segments. Two metrics, mutual information and K-L divergence, have been utilized to measure the distances between bands. In terms of the Ward's linkage theory [72],

partitions with minimum variance can be achieved, and the band which is most identical to the rest bands is selected in each cluster. By considering the contextual information of the HSI dataset, Yuan *et al.* have proposed a novel clustering method, i.e., dual-clustering-based band selection by context analysis (DCCA), for UBS [51]. Along with the input raw HSI data, the DCCA has designed a new pairwise hyperspectral angle descriptor to exploit the contextual information of each pixel in HSI. With the dual clustering framework, the contextual feature of the HSI and the raw HSI are grouped simultaneously and the mutual effect of these two features determine the clustering result. Similar to other clustering-based methods, the most representative band from each cluster is selected based on a groupwise strategy.

Nowadays, it has become a trend to combine the ranking-based and the clustering-based methods. For the ranking-based methods, most representative bands can be easily found. Meanwhile, the clustering-based methods can restrict the correlation within the obtained subset of bands. Therefore, the merits from these two methods can enhance the performance of UBS. Inspired by the fast-peak-based clustering (FDPC) [73], Jia *et al.* have proposed the enhanced FDPC (E-FDPC) [52] where the characteristic of each band can be determined by its local density and its distance to the nearest high density band. The significance of each band can be determined by considering these two factors jointly. Based on the assumption that the band with a higher local density and maximum nearest neighbour distance is the cluster centre, top ranked bands are chosen to form the band subset, which is still similar to most ranking-based methods. Different from the E-FDPC which combines the clustering-based methods into the ranking-based methods, Wang *et al.* has further developed an optimal cluster-

ing framework (OCF) for HSI band selection [53]. With two defined objective functions, the normalized cut and top-rank cut have been used to partitioned the whole dataset into several clusters by an optimal way. Three ranking strategies, including E-FDPC, MVPCA, and Information Entropy, are utilized to find the most important band from each cluster. The performance of OCF has validated the successfully cooperation between ranking-based and clustering-based UBS methods. In [54], the adaptive subspace partition strategy (ASPS) has been proposed for UBS in HSI. By applying a coarse to fine strategy, the bands are grouped into different subcubes. By estimating the noise information for each band, the band with the minimum noise is considered as the most representative one for that subcube and added to the subset of selected bands. The experimental reults have further emphasized the importance of removing the noisy band from the selected band subset.

Recently, in addition to the ranking-based and clustering-based methods, optimization-based UBS methods have attracted increasing attention as the iterative process seems more controllable to obtain the number of the selected bands. The volume gradient band selection method (VGBS) is introduced by deriving the 'volume' information from the covariance matrix of all bands [55]. Instead of calculating any measurements between a single band and all other bands, VGBS removes the most redundant band by the assumption that it usually has the maximum gradient in the dataset. Different from the VGBS algorithm, the multitask sparsity pursuit (MTSP) [56] attempts to find an optimal solution by iteratively updating the chosen band subset. In MTSP, a constructed data descriptor based on the compressive sensing theory is firstly utilized to reduce the original HSI data, and a band subset with the desired number of bands can be obtained ran-

domly. Afterwards, a multitask sparse representation-based criterion is utilized to examine the potential band groups. By updating the preliminary band subset using the immune clonal strategy, the optimized result can be obtained. Under the consideration of structure information from both band informativeness and independence, Zhu *et al.* developed a greedy-search-based UBS approach by tackling a graph-based clustering problem with dominant set extraction (DSEBS) [57]. The DSEBS takes the advantage of the first-order statistic of local spatial-spectral consistencies and structure correlation for quantifying band information and independence. After that, the band selection task is transformed to a dense subgraph discovery problem, where the dominant set extraction can provide an optimal solution. In DSEBS, the interdependencies between bands determine the reliability of each band and its contribution to the final result. By choosing the optimal band subset iteratively, the optimization-based UBS methods have comparable achievement. However, two major drawbacks restrict the performance of this kind of methods. Foremost, the iterative process usually focuses more on each individual bands, which fails to filter the contributions from noisy bands. Secondly, there is a trade-off between the computational complexity and performance in the iterative process, hence some valuable information may be compromised for reducing the complexity.

Recently, AE and its extended work have proved its superiority in extracting more effective features [74, 75]. Different from other deep learning-based neural networks, the basic idea of AE-based feature selection is to learn hidden representations that can effectively reconstruct the input data. Due to its strong ability to investigate both linear and nonlinear information among features, the AE has been utilized in the high dimensional data feature selection in an unsupervised

manner [74]. In the UBS for HSI, the AE-based methods are not as popular as the above summarized methods. In [75], the input weights of the AE are utilized to select most significant bands in an unsupervised way and the selected band subsets have achieved a good performance. Cai et al [76] have proposed another similar end-to-end CNN for band selection, where the final band subset is determined by ranking the average of the learned weight of each band. Although the above two methods introduce the AE into the UBS work, the principle of ranking the band with the learned weight from AE is not reasonable enough as the statement in the last section.

2.5 Theoretical background

2.5.1 Sparse Representation Classification

The SRC was firstly extended to the HSI classification in [63], which is based on the observation that each spectral pixel can be approximately presented by a combination of training samples from the same semantic class. For an HSI image, one test pixel can be presented as $y \in R^{m*1}$ with m indicates the number of bands. By randomly selecting the training samples from each semantic class, a structural dictionary $D = D_1, ..., D_C \in R^{m*N}$ can be built, where the $D_c \in R^{m*N_c}$ represents the sub-dictionary of class $c = 1, ..., C$ and N_c is the number of atoms in the sub-dictionary. Besides, the number of whole training samples is given by $N = N_1 + N_2 + ... + N_c + ... + N_C$. With the built dictionary, the test pixel can be linear approximated as:

$$y = D * \alpha \tag{2.1}$$

where $\alpha \in R^{N*1}$ is the sparse coefficient vector with the dimension equalling to the number of atoms in the dictionary. In the SRC, the sparse coefficient vector only has defined non-zero entries. Based on that, the coefficient vector α can be recovered by solving the following problem:

$$\hat{\alpha} = \arg\min_{\alpha} ||y - D * \alpha||_2, ||\alpha||_0 \leq L \tag{2.2}$$

where L corresponds to the number of non-zero coefficients within $\hat{\alpha}$, which is also called the sparsity level. The above problem is also known as a non-deterministic polynomial-time hard (NP-hard) problem, which can be approximately solved by greedy search algorithms, such as the orthogonal matching pursuit (OMP). After estimating the sparse coefficient vector $\hat{\alpha}$, the class label of the test pixel y can be determined by the criterion of minimum reconstruction error:

$$\hat{c} = \arg\min_{c=1,...,C} ||y - D_c * \hat{\alpha}_c||_2 \tag{2.3}$$

To improve classification performance, many SRC-based methods have attempted to combine more spatial information [63–66]. The common way is to define a fixed-size square window and assume all the neighbouring pixels within this window to be in the same class of the test pixel [63–66]. The pixels within the window, including the test pixel and its neighbouring pixel, can be stacked into a matrix Y first. Then, the corresponding sparse coefficient matrix A can be estimated:

$$\hat{A} = \arg\min_{A} ||Y - D * A||_2, ||A||_0 \leq L \tag{2.4}$$

and the class of the test pixel can be determined by:

$$\hat{c} = \arg \min_{c=1,\ldots,C} ||Y - D_c * \hat{A}_c||_2 \tag{2.5}$$

2.5.2 Canonical Correlation Analysis

The canonical correlation analysis (CCA) has become powerful in many computer vision applications [77–79], where the purpose of CCA is to discover the intrinsic association between different modalities and preserve the useful information for data fusion. Mathematically, it is implemented by maximizing the projections of two sets of variables on the basis vectors, which is considered as the maximization of mutual information.

Let F_1 and F_2 denote two sets of variables. There is a pair of vectors w_1 and w_2 to project these two variables into a pair of canonical variables $w_1^T F_1$ and $w_2^T F_2$,. The process of maximizing their correlation is given by:

$$\arg \max_{w_1, w_2} w_1^T F_1 F_2^T w_2 \tag{2.6}$$

where w_1 and w_2 are constrained by:

$$w_1^T F_1 F_1^T w_1 = w_2^T F_2 F_2^T w_2 = 1 \tag{2.7}$$

To obtain the projected vectors, the following relationship can be satisfied by applying the Lagrange multipliers [77]:

$$\begin{bmatrix} 0 & F_1 F_2^T \\ F_2 F_1^T & 0 \end{bmatrix} w = \mu \begin{bmatrix} F_1 F_1^T & 0 \\ 0 & F_2 F_2^T \end{bmatrix} w \tag{2.8}$$

where the $w = [w_1, w_2]$ and μ is the canonical correlation value.

As an extension of the CCA, the objective of multiple CCA (MCCA) is to find a set of projection vectors $w = [w_1^T, w_2^T, ..., w_p, ..., w_P^T]^T$ for handling more than two sets of variables $F_1, F_2, ..., F_p, ..., F_P$. The formulation of MCCA is presented as:

$$\arg \max_{w_1, w_2, ..., w_p, ..., w_P} \frac{1}{P-1} \sum_{q,r=1}^{P} w_q^T F_q F_r^T w_r (q \neq r) \tag{2.9}$$

and the projection vectors satisfy the following equation:

$$\sum_{q=1}^{P} w_q^T F_q F_q^T w_q = P \tag{2.10}$$

Similar to CCA, the formulation of MCCA can be transformed into (2.11) by applying the Lagrange multipliers:

$$\frac{1}{P-1}(G - H)w = \mu H w \tag{2.11}$$

where

$$G = \begin{bmatrix} F_1 F_1^T & ... & F_1 F_P^T \\ \vdots & \ddots & \vdots \\ F_P F_1^T & ... & F_P F_P^T \end{bmatrix} \tag{2.12}$$

$$H = \begin{bmatrix} F_1 F_1^T & ... & 0 \\ \vdots & \ddots & \vdots \\ 0 & ... & F_P F_P^T \end{bmatrix} \tag{2.13}$$

where μ refers to the multiple canonical correlation values, which can be computed by the generalized eigenvalue method.

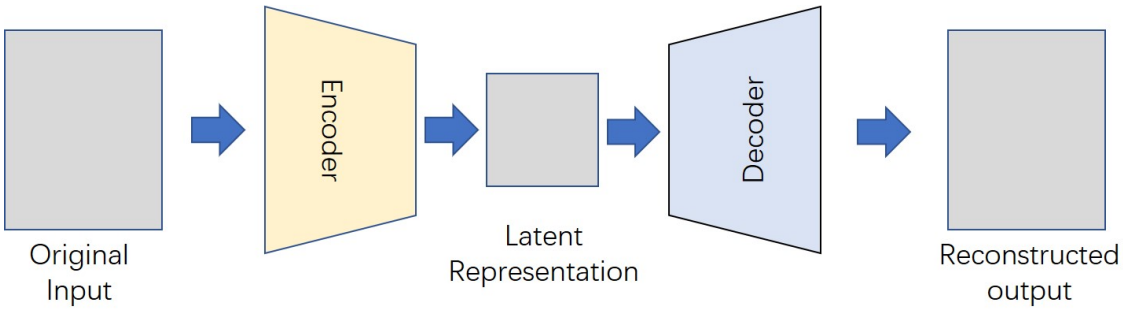


Figure 2.6: The flowchart of the AE

2.5.3 Autoencoders

Different from most of current deep learning models, AE enables to learn an efficient hidden data representation in an unsupervised manner, which has become a useful tool in many applications [74, 75, 80–83]. The purpose of AE is to learn a low dimensional representation of the original data, which is often implemented by mapping the data into a latent representation and then reconstructing the data based on the data coding. An AE usually consists of two module: an encoder for compressing the data into latent space and a decoder for reconstructing the input data. From the flowchart of an AE shown in Figure 2.1, the unsupervised training process of the AE is by considering the residual between original input and reconstructed output as the loss.

Most of the deep learning models aim to generate an desired output like classification map, segmentation map, or object detection bounding box. However, the AE attempts to extract a more valuable latent representation, which has lower dimensions than the original input. In this way, the AE can be used to choose useful features among the orignial data whilst reducing the data dimensions in an effective way [74, 75, 80].

Chapter 3

Superpixel-based Sparse Representation for Spectral-Spatial Classification of Hyperspectral Images

3.1 Introduction

The lack of sufficient training samples is a common problem in practical applications, which is also addressed in the proposed framework. For improving the classification accuracy, effective fusion of spectral and spatial features in the SRC-based classification framework have attracted increasing attention. Most of current SRC-based methods [26–28, 67] utilize adaptive strategies to estimate the sparse coefficients and determine the label of the test pixel by the sum of residuals from all extracted features. In [67], a collaborative representation-based multitask

learning framework is introduced for fusion of multiple extracted features, where the significance of each feature is represented by an adaptive weight. Zhang *et al* have built a joint SRC-based multisource classification framework [26], where a locality adaptive weighting strategy is employed to improve the feature fusion from different data. In [27], a multiple feature adaptive SRC framework (MFASR) has been proposed, where the generated sparse coefficients are obtained adaptively to keep the feature-specific pattern for multiple feature learning and the classification performance has been improved. Moreover, the similar kernel version of the multiple feature SRC [28] has also been introduced and shown the significance of the non-linear separability, which can significantly improve the classification accuracy. Although these approaches have shown relative good performance, the mechanism for fusion of multiple features needs be further analysed to derive a more robust strategy.

To improve the efficiency and maintain the classification accuracy under the circumstance of insufficient training samples, a superpixel-based feature specific sparse representation framework (SPFS-SRC) is proposed in this chapter for the classification of HSI. First, the PCA analysis [84] is used to reduce the dimension of HSI. Second, the extended morphological profiles (EMPs) [85] are extracted as spatial features from the 1st principle component. Afterwards, the linear spectral clustering (LSC) oversegmentation approach [86] is applied on the first three principle components to generate superpixels of the HSI. Pixels in each superpixel is assumed to share similar spatial-spectral characteristics. Before the classification, an online metric learning step is used for weighting each atom in the dictionary. With the kernel-based sparse regularization, the sparse coefficients are obtained. Finally, instead of labelling each pixel in the superpixel, the recovered sparse co-

efficients can be jointly utilized to calculate the reconstruction residual and assign the class label for the whole superpixel, which can reduce the computational cost.

As descriped in last section of this chapter, many algorithms have attempted to incorporate more spatial information in the classification framework, for example, the JSRC [63]. Although the JSRC has a better classification accuracy than the SRC, it has several drawbacks: first, with the fixed (size and shape) window strategy, many unrelated pixels may be chosen to the test pixel whilst correlated pixels may be missed. Second, with unlabelled neighbouring pixels used for estimation, this may increase the computational time of the classifier. Besides, only the spectral information within a neighborhood is utilized in the classification framework, for which more robust spatial features are required. To address these issues, the designed superpixel-based feature specific SRC framework is proposed. According to the superpixel of the HSI, the spatial neighbouring region around each test pixel can be determined. During the classification, all the pixels within the superpixel are regarded from the same class and labelled simultaneously, which can significantly improve the efficiency of the SRC. To better exploit the spatial information, spatial features and the online metric learning strategy are applied for obtaining shared sparse matching from multiple features whilst maintaining the feature-specific sparse pattern.

Although the above SRC-based methods [26–28,63,67] have achieved relatively good performance in HSI classification, there are still some drawbacks. Firstly, most of these methods employ the raw features or the extracted features in the SRC framework directly, where the total dimensions of the raw features or the extracted features are quite high. As a result, it is easy to overfit and also the method suffers from the large computational burden. Secondly, the existing

mechanisms of fusion of multiple features focus mainly on the equal utilization of each individual feature, where the correlation between different features has not been properly analysed. Besides, the adaptive weighting strategy sometimes cannot choose weights to reflect the separability of each feature.

Therefore, a superpixel-based multiple feature fusion SRC framework (SMFF-SRC) has been proposed to improve the efficiency and efficacy especially when there is insufficient training samples. Firstly, the principal component analysis (PCA) and the extended morphological profiles (EMPs) are utilized to extract the spectral feature and the spatial feature, respectively, along with the elevation feature if the LiDAR data is available. Secondly, the simple linear iterative clustering (SLIC)-based oversegmentation method [90] is employed to extract the superpixels from the first three PCA components of the HSI. Afterwards, a localized multiple canonical correlation analysis (MCCA) algorithm has been implemented to produce a fused feature-based on the extracted multiple features. The fused feature can represent the discriminative ability of all features in a low-dimensional space. With the kernel-based sparse coding algorithm, the sparse coefficients can be obtained. To reduce the computational burden, the proposed framework assumes that all pixels within one superpixel belong to the same labelled class. In this way, the estimated sparse coefficients can be jointly utilized to obtain the reconstruction residual for effective classification.

The rest of this chapter is organized as follows: the second section will describe the SPFS-SRC method, including the experimental results. After that, the SMFF-SRC method will be introduced. In the last section, a brief summary of this chapter will be given.

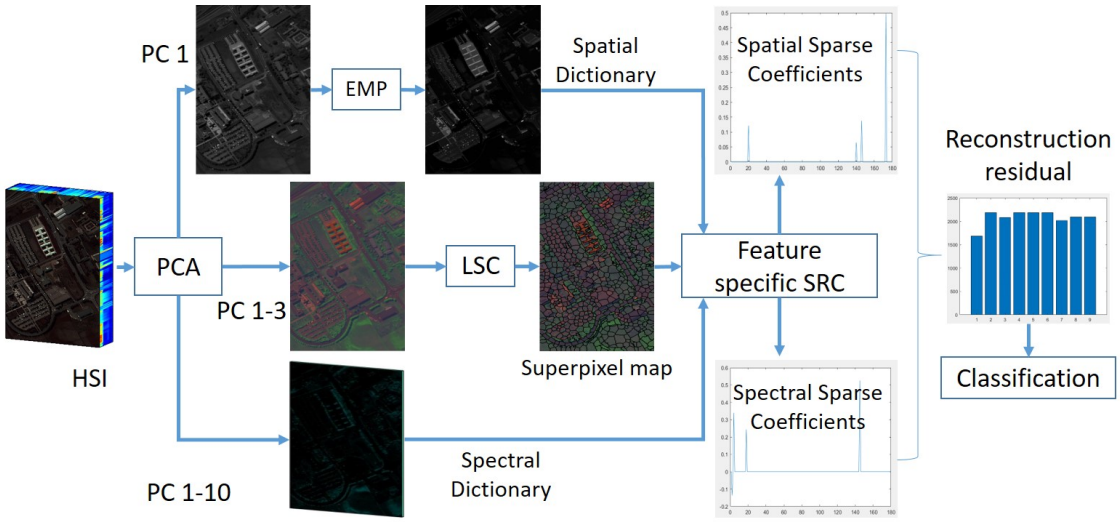


Figure 3.1: The flowchart of the proposed SPFS-SRC framework

3.2 Superpixel-based Feature Specific Sparse Representation for Spectral-Spatial Classification of Hyperspectral Images

Inspired by the aforementioned challenges and the success of SRC in HSI classification, an improved SRC-based framework for the HSI classification has been proposed in this section. The framework consists of two components: superpixel generation and kernel-based SRC with proximity constraint using the online metric learning. Figure 3.1 shows the flowchart of the proposed framework, with the details discussed as follows.

3.2.1 Superpixel generation

For obtaining the superpixel map of the HSI, an efficient oversegmentation approach is applied firstly [86]. Since the HSI usually contains hundreds of bands,

it is unrealistic to perform the segmentation on the raw data. For saving the computation cost, the PCA is applied to the HSI and the first three principal components are extracted and taken as a false-colored image for segmentation using the LSC algorithm [86]. The LSC algorithm runs efficiently in linear complexity, which can optimize the segmentation cost function of normalized cut by applying the weighted k-means clustering strategy. The LSC algorithm proposes a novel relationship between the objective functions of the normalized cut and weighted k-means. Both objective functions can be equivalently optimised when the similarity between two points is equal to the weighted inner product between the two corresponding vectors [86]. During the superpixel generation process in the proposed framework, the seeds are initialized with fixed spacing intervals in the false-colored image, and each seed is moved to its lowest neighbour. For each cluster, a weighted mean and a search center are calculated iteratively until the weighted means converge for all clusters. After grouping tiny superpixels, a superpixel map of HSI can be generated.

3.2.2 Superpixel-based SRC

After creating the superpixel map, the original HSI can be divided into many spatial regions. Similar to the JSRC, pixels in each superpixel can be stacked into a matrix $Y_i = [y_{i,1}, ...y_{i,n_i}]$, where i represents the index of the superpixel and n_i denotes the number of pixels it contains. Because it can be assumed that pixels in one superpixel share the same spectral characteristics, all those pixels are considered jointly in the SRC framework.

With a 3-D cube of the HSI, rich spatial information is contained along with the spectral information. The EMPs are extracted to represent the spatial infor-

mation. With the EMPs and the raw spectral data, fusion of the spectral-spatial features can be applied into the HSI classification. Although it is very straightforward to stack the spatial feature and spectral feature together, the derived high dimensional data may lead the overfitting. In [67], a simple weighted strategy is applied into the calculation of the reconstruction residual, where the weights of all extracted features are defined empirically. In [26], an adaptive weight strategy is designed, which sets a high penalty to the zero sparse coefficients based on the previous iteration. In the Multiple Feature Adaptive SRC [27] approach, the label of the test pixel is also determined by the sum of residuals from all extracted features. However, for the above methods, the significance of each extracted feature is not used, as most of them consider each feature equally. It is crucial to consider the difference among extracted features and preserve the regulariztion between the test and training samples.

To this end, a learned distance on the joint sparse representation constraint has been imposed. The SRC problem can be modified accordingly as:

$$\hat{A^k} = \arg\min_{A^k} \sum_{1}^{K} ||Y^k - D^k * A^k||_2 + \lambda * \sum_{1}^{K} ||B^k \bigodot A^k|| \tag{3.1}$$

where $k = 1, ..., K$ is the index of the extracted features, and Y^K, D^k and A^k are the test pixel matrix, dictionary and sparse coefficients matrix in the kth feature. The B^k is the learned distance in the kth feature and $\bigodot$ represents the element-wise multiplication.

With the learned distance between test samples and training samples, the training samples which are closer to the test ones would be used for the reconstruction, which is corresponding to the fact that similar samples are more likely

to be in the same class. Therefore, an online metric learning strategy has been introduced to preserve the locality of data between the test sample and training samples. Generally, the predefined Euclidean distance is employed to measure the data similarity. In this chapter, an Mahalanobis-based distance has been applied to find the matching between the test and training samples with multiple features, which guarantees to obtain more accurate sparse coefficients [87, 88]. The distance function between two samples x_1 and x_2 in the kth feature is defined as follows:

$$\hat{B^k}(x_1^k, x_2^k) = \sqrt{\omega^k g^k(x_1^k, x_2^k)} \tag{3.2}$$

where ω^k is a nonnegative weight for the kth feature, and it is constrained by $\sum_1^K \omega^k = 1$; $g^k(x_1^k, x_2^k) = (x_1^k - x_2^k)^T M^k (x_1^k - x_2^k)$ is the distance function for the kth feature with the Mahalanobis metric M^k.

Inspired by the LogDet Extract Gradient Online (LEGO) algorithm [87] and its application in visual tracking [88], the proposed method aims to learn the feature weight ω and distance metric M^K iteratively. By acquiring the training sample pairs from the built dictionary for classification, two determination statements ϕ_1 and ϕ_2 are defined for the training sample pairs. If the two samples in one training sample pair come from the same class, it can be assumed that the ground truth of this training sample pair is "similar", the condition ϕ_1 is described as:

$$\phi_1 = \begin{cases} True, & if \quad g_{p-1}^k(x_1^k, x_2^k) = (x_1^k - x_2^k)^T M_{p-1}^k (x_1^k - x_2^k) \geq \theta_1 \\ Flase, & otherwise \end{cases} \tag{3.3}$$

where $p = 1, ..., P$ represents the number of iteration and P is equal to the number

Algorithm 1 Online metric learning

Input: initialize feature weight $\omega_0^k = \frac{1}{K}$; Metric $M_0^k, k = 1, ..., K$; β; η;

Initialisation: Generate P training sample pairs randomly: $(x_1^{k,p}), (x_2^{k,p})$, where $p = 1, ..., P$ and $k = 1, ...K$.

for $p = 1$ to P **do**

  for $k = 1$ to K **do**

    case 1:% similar pairs

    if ϕ_1 holds true **then**

      $\xi = 1$, update weight $\omega_{k,p}$ by Eq. (3.5) and M_p^k using Eq. (3.7) - (3.10);

    else

      $\xi = 0$

    end if

    case 2:% dissimilar pairs

    if ϕ_2 holds true **then**

      $\xi = 1$, update weight $\omega_{k,p}$ using Eq. (3.5) and M_p^k using Eq. (3.7) - (3.10);

    else

      $\xi = 0$

    end if

  end for

  Update weight by (3.6);

end for

Output: $M^k, k = 1, ..., K$ and $\omega_k, k = 1, ...K$;

of training sample pairs; θ_1 is the threshold for determining the similarity of training sample pair. M_{p-1}^k is the learned metric from the last iteration and g_{p-1}^k is the related distance function. If the ϕ_1 holds true, the two samples in this part does not match the ground truth, otherwise this pair matches the ground truth. Likewise, if the two samples in one training sample pair comes from different classes, it can be considered that the ground truth of this training sample pair is "dissimilar", the statement is depicted as:

$$\phi_2 = \begin{cases} True, & if \quad g_{p-1}^k(x_1^k, x_2^k) = (x_1^k - x_2^k)^T M_{p-1}^k (x_1^k - x_2^k) \leq \theta_2 \\ Flase, & otherwise \end{cases} \tag{3.4}$$

If the ϕ_2 holds true, the two samples in this pair does not match the ground truth. Otherwise, the determination is corresponding to the ground truth.

Although the training sample pairs can be acquired from the built dictionary before classification, an online-based metric learning process has been designed to fully exploit the correlation between the training sample pairs. The training sample pairs of each feature are selected randomly to learn the Mahalanobis metric M^k. For the feature weight ω^k, it is updated by using the Hedge algorithm [89]. With all selected training sample pairs, the weight and metric for each feature are obtained as follows:

(1) Weight updating

The weight for each feature can be estimated using the Hedging algorithm as follows [89]:

$$\omega^{\hat{k},p} = \omega^{k,p-1}\beta^{\xi} \tag{3.5}$$

$$\omega^{k,t} = \frac{\hat{\omega^{k,p}}}{\sum_1^K \omega^{k,p}} \tag{3.6}$$

where $\beta \in (0,1)$ is a penalty coefficient; if the training sample pair for the kth feature meets the conditions in Eqs. (8) or (9), then $\xi = 1$, otherwise it is 0. if the determined result of the training sample pair for the kth feature is against the ground truth, the feature weight is penalized.

(2) Metric Updating

According to the LEGO algorithm [87], if the training sample pair for the kth feature is punished based on the judgement, the Mahalanobis metric M^k is updated by:

$$M^{k,p} = M^{k,p-1} - \frac{\eta(v-td)M^{k,p-1}(x_1^{k,p} - x_2^{k,p})(x_1^{k,p} - x_2^{k,p})^T M^{k,p-1}}{1 + \eta(v-td)(x_1^{k,p} - x_2^{k,p})M^{k,p-1}(x_1^{k,p} - x_2^{k,p})} \tag{3.7}$$

$$v_{uppleft} = \eta td(x_1^{k,p} - x_2^{k,p})M^{k,p-1}(x_1^{k,p} - x_2^{k,p}) - 1 \tag{3.8}$$

$$v_{uppright} = \sqrt{v_{uppleft})^2 + 4\eta((x_1^{k,p} - x_2^{k,p})M^{k,p-1}(x_1^{k,p} - x_2^{k,p}))^2} \tag{3.9}$$

$$v = \frac{v_{uppleft} + v_{uppright}}{2\eta(x_1^{k,p} - x_2^{k,p})M^{k,p-1}(x_1^{k,p} - x_2^{k,p})} \tag{3.10}$$

where td denotes the target distance measured by using the Euclidean distance between two sample points instead of a fixed value. On the other hand, if the evaluation of the training sample pair is exactly the same as defined in the ground truth, the metric is maintained.

The proposed weight and metric updating algorithm is summarized in Algorithm 1. With the obtained metric and feature weights, the distance B^k can be calculated. During the training process, the obtained Mahalanobis-based metric

can be more discriminative to reflect the importance of each feature.

Since the EMPs have been extracted as the spatial features, the kernel-based SRC is utilized to estimate the sparse coefficients for improving the non-linear separability of SRC:

$$\hat{A}_{\phi}^{k} = \arg\min_{A_{\phi}} \sum_{1}^{K} ||Y_{\phi}^{k} - D_{\phi}^{k} * A_{\phi}^{k}||_2 + \lambda * \sum_{1}^{K} ||B^{k} \bigodot A_{\phi}^{k}|| \tag{3.11}$$

where the radial basis function (RBF) is used as the operated kernel function, and ϕ represents the kernel domain. With the estimated sparse coefficients in each feature, the label for all pixels of the superpixel can be assigned to the class with the minimum sum of residuals from multiple features:

$$\hat{c} = \arg\min_{c=1,\ldots,C} \sum_{1}^{K} ||Y_{\phi}^{k} - D_{\phi,c}^{k} * \hat{A}_{\phi,c}^{k}||_2 \tag{3.12}$$

The whole SPFS-SRC algorithm is summarized in Algorithm 2.

Algorithm 2 SPFS-SRC

1: **Input:** raw HSI data

2: **Feature extraction:** Utilize PCA to extract principle components and then use the first component extracted from the PCA to generate EMPs as the spatial feature. The first ten components extracted from the PCA are utilized as the spectral feature.

3: **Superpixel generation:** Apply the LSC algorithm to create superpixel map by using the first three components extracted from the PCA.

4: **Metric learning:** Learn the weight of each feature and update the distance between training samples and test samples according to Algorithm 1.

5: **Superpixel classification:** Classify each superpixel based on kernel-based SRC and learned metrics.

6: **Output:** Classification map;

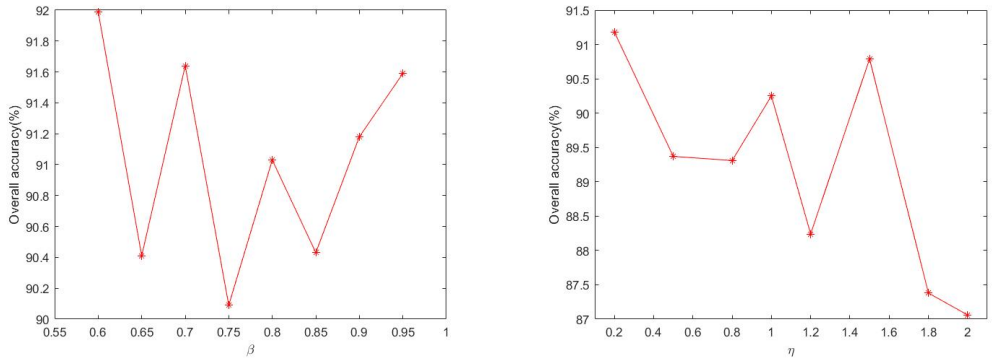
3.2.3 Experimental results

Experimental Settings

In this chapter, three common metrics have been used for quantitative performance evaluation, including the overall accuracy (OA), the average accuracy (AA) and the Kappa coefficient. OA reflects the percentage of correctly classified pixels, whilst AA denotes the mean of the class based classification accuracy. The Kappa coefficient represents the consistency of the classification result, which is estimated based on the confusion matrix. For the utilized PaviaU and Indian Pine datasets, the training data are selected randomly from all samples and the rest are used for testing. All experiments are completed with a 16 GB Intel i5-6500 CPU on the MATLAB 2017b.

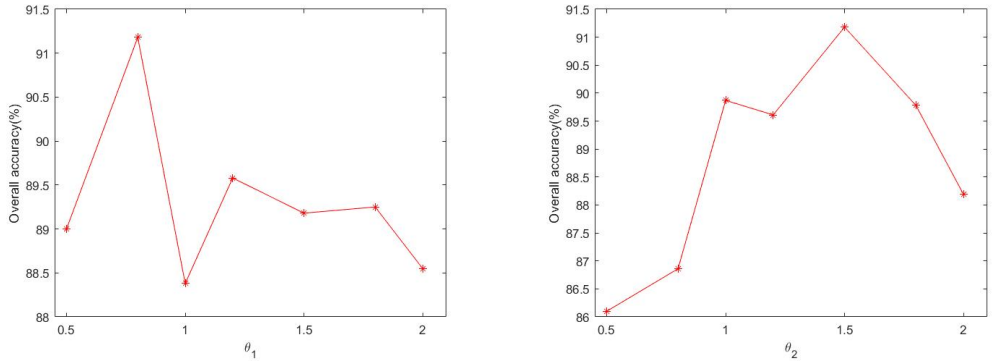
In the designed online metric learning strategy, there are four predefined parameters, which include: two thresholds θ_1, θ_2, the discounting parameter β and one regularization parameter η. To validate the effect of those four parameters on the OA, related experiments are carried out on both datasets. For the PaviaU dataset, 20 randomly selected training samples per class are used to train the classifier. In total 180 training sample pairs from 9 classes are formed, including 90 "similar" pairs and 90 "dissimilar" training pairs, to guarantee the metric learning approach. The experimental results are shown in Figures 3.2-3.3, and all experiments are repeated 10 times, where the OA is the average value on the 10 experiments.

As seen in Figure 3.2, the discounting parameter β and the regularization parameter η have limited effect on the OA, from which it can be assumed that the proposed approach is insensitive to these two parameters. In experiments, the



(a) (b)

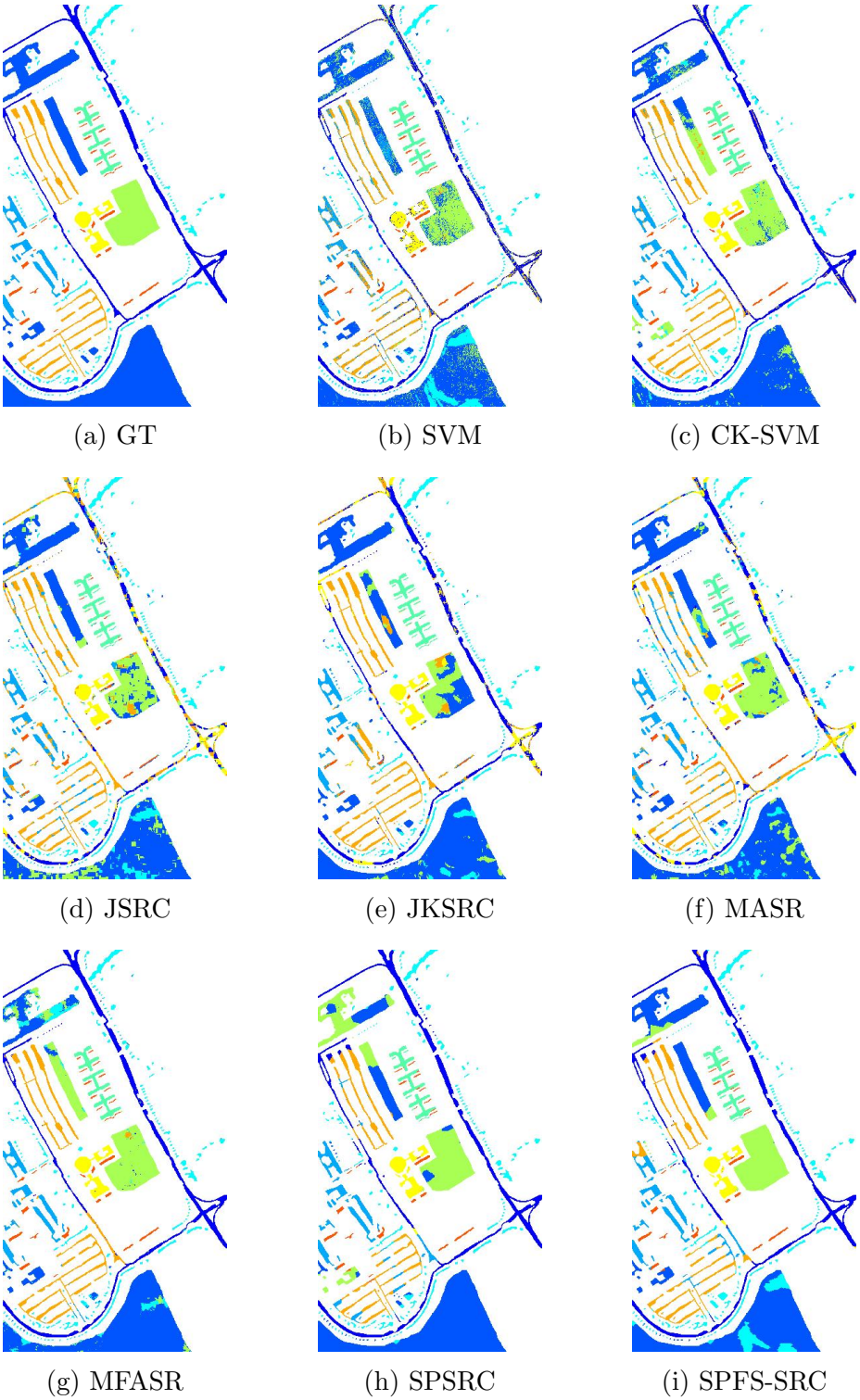
Figure 3.2: The effect of β and η on OA(%). (a) β, (b) η.



(a) (b)

Figure 3.3: The effect of θ_1 and θ_2 on OA(%). (a) θ_1, (b) θ_2.

penalization parameter is chosen as 0.9, and η is set to 0.2 as suggested in [87,88]. For two thresholds θ_1 and θ_2, they were set to 0.8 and 1.5 according to Figure 3.3, respectively. For the Indian Pine dataset, same parameters are adopted while 16 "similar" sample pairs and 16 "dissimilar" sample pairs have randomly chosen from the training samples.



(a) GT (b) SVM (c) CK-SVM

(d) JSRC (e) JKSRC (f) MASR

(g) MFASR (h) SPSRC (i) SPFS-SRC

Figure 3.4: The classification map of the PaviaU dataset. (a)the Ground Truth (GT), (b) SVM, (c) CK-SVM, (d) JSRC, (e) KSRC, (f) MASR, (g) MFASR, (h) SPSRC, (i) SPFS-SRC.

Table 3.1: Number of training and testing samples in each class for the PaviaU dataset.

PaviaU dataset							
Class		Sample		Class		Class	
Label	Name	Train	Test	Label	Name	Train	Test
1	Asphalt	20	6611	6	Bare Soil	20	5009
2	Meadows	20	18629	7	Bitumen	20	1310
3	Gravel	20	2079	8	Self-blocking bricks	20	3662
4	Trees	20	3044	9	Shadows	20	927
5	Painted metal sheets	20	1325	Total		180	42596

Table 3.2: Number of training and testing samples in each class for the Indian Pine dataset.

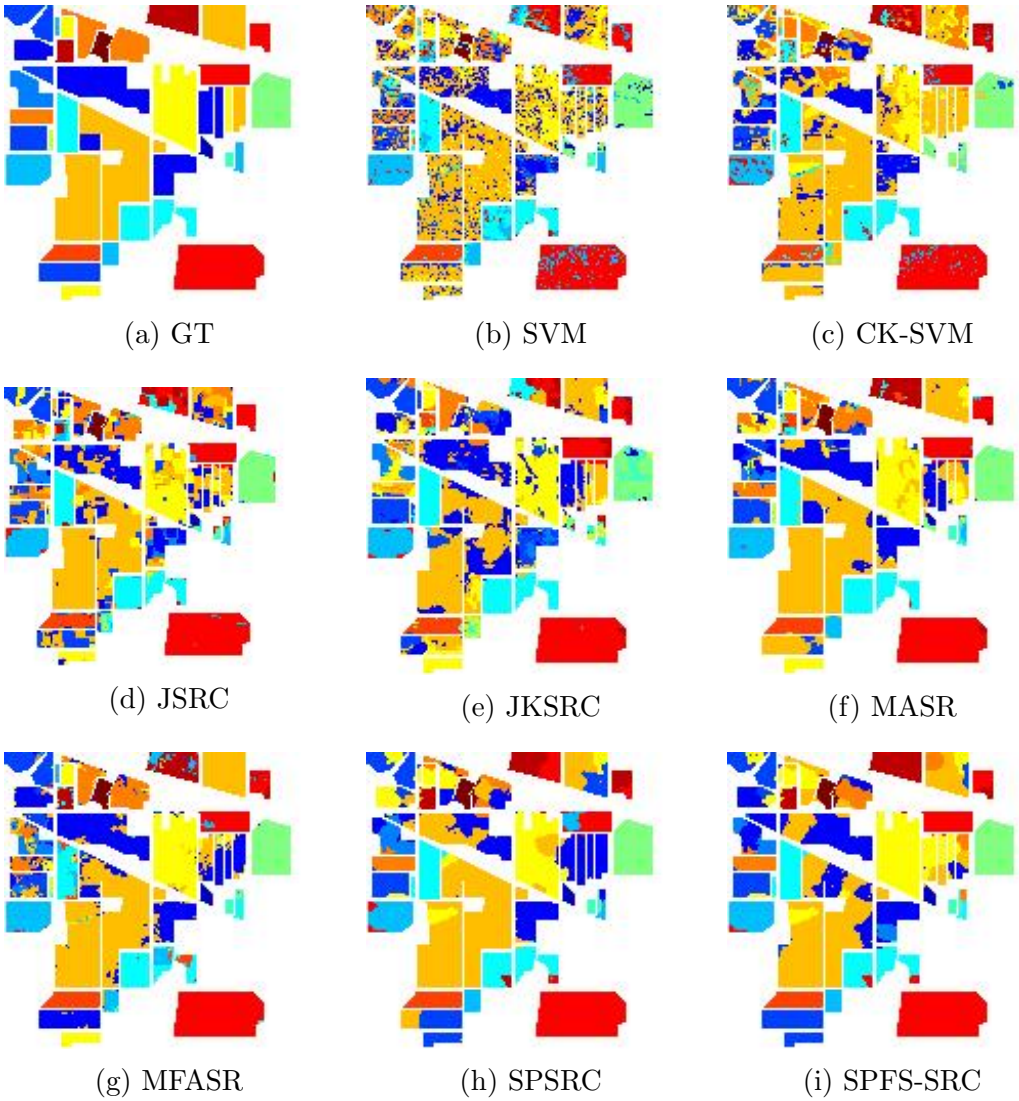
Indian Pine dataset							
Class		Sample		Class		Class	
Label	Name	Train	Test	Label	Name	Train	Test
1	Alfalfa	2	44	9	Oats	2	18
2	Corn-notill	14	1414	10	Soybeans-notill	10	962
3	Corn-min	9	821	11	Soybeans-min	25	2430
4	Corn	3	234	12	Soybeans-clean	7	586
5	Grass/pasture	5	478	13	Wheat	3	202
6	Grass/trees	8	722	14	Woods	13	1252
7	Grass/pasture-mowed	2	26	15	Bldg-gass-tree drives	4	382
8	Hay-windowed	5	473	16	Stone-steel towers	2	91
Total						114	10135

Table 3.3: Classification results from different approaches for the PaviaU dataset with 20 training samples per class (Best result of each row is marked in bold type)

Methods	SVM	CK-SVM	JSRC	JKSRC	MASR	MFASR	SPSRC	SPFS-SRC
OA(%)	78.04±0.04	89.05± 0.03	64.12±0.04	73.81±0.04	78.97±0.03	84.16±0.02	88.98±0.03	**91.51±0.01**
AA(%)	81.64±0.01	94.03±0.01	53.40±0.05	66.53±0.04	73.00±0.03	**95.65±0.01**	85.80±0.03	88.92±0.02
Kappa	0.68±0.04	0.86±0.04	0.61±0.04	0.75±0.02	0.82±0.01	0.86±0.02	0.91±0.01	**0.92±0.01**
Time(s)	6.12±0.01	11.12±0.03	61.99±0.01	57.80±0.01	331.87±12.57	266.05±10.87	**5.77±0.02**	12.1±0.01

Comparison Experiments

To evaluate the performance of the proposed framework under the situation of a small number of training samples, the developed framework with some state-of-the-art algorithms has been compared, including the support vector machine



(a) GT (b) SVM (c) CK-SVM

(d) JSRC (e) JKSRC (f) MASR

(g) MFASR (h) SPSRC (i) SPFS-SRC

Figure 3.5: The classification map of the Indian Pine dataset. (a)the Ground Truth (GT), (b) SVM, (c) CK-SVM, (d) JSRC, (e) KSRC, (f) MASR, (g) MFASR, (h) SPSRC, (i) SPFS-SRC.

(SVM), the composite kernel support vector machine (CK-SVM) [29], the JSRC and the JKSRC [63, 64], the MASR [66], the MFASR [27]. To better detect the effect of the proposed online metric learning approach, a superpixel-based SRC model (SPSRC) without metric learning is also applied.

The parameter settings for the proposed approach and other compared meth-

Table 3.4: Classification results from different approaches for the Indian Pine dataset with 1% training samples (Best result of each row is marked in bold type).

Methods	SVM	CK-SVM	JSRC	JKSRC	MASR	MFASR	SPSRC	SPFS-SRC
OA(%)	54.90±0.02	62.35±0.02	65.20±0.02	70.37±0.04	80.21±0.02	81.79±0.04	82.38±0.03	**83.71±0.01**
AA(%)	55.71±0.02	58.47±0.07	60.15±0.03	65.98±0.05	77.27±0.02	**82.71±0.02**	79.82±0.03	81.36±0.01
Kappa	0.48±0.02	0.57±0.03	0.66±0.02	0.68±0.03	**0.81±0.02**	0.79±0.02	**0.81±0.03**	0.80±0.04
Time(s)	1.60±0.02	6.42±0.02	7.65±0.12	16.43±0.75	137.57±2.56	13.45±0.52	**0.32±0.02**	1.22±0.02

ods are summarized as follows. The parameters of the proposed SPFS-SRC method, the SPSRC method, and the SVM-based algorithm, including kernel parameters and the reguralization parameters, are all determined via cross-validation. For JSRC and JKSRC, the default parameters suggested in [15] are adopted yet based on self-implementation of the algorithms. For other methods including CK-SVM, MASR and MFASR, experiments are tested on original codes with the default parameters. For CK-SVM and MFASR, the same spatial and spectral features are utilized for consistency. In addition, for all SRC-based methods, the sparsity level is set to 3 for efficiency.

For the PaviaU dataset, 20 samples per class are randomly selected for training, whilst the rest samples are utilized for testing. For the Indian Pine dataset, the number of samples in each class are rather unbalanced, for example, there are only 26 samples and 20 samples in the class "Grass-pasture-mowed" and "Oats". Hence, for each class 1% of the samples or 2 is selected if the total number of samples in that class is below 200 for training, and the rest samples are used for testing. The number of samples used for training and testing in each class in the two datasets are listed in Tables 3.1 and 3.2, respectively. After random selection of training samples in both datasets, the chosen training samples are excluded in each superpixel to avoid the inaccurate estimation of classification accuracy. The

experimental results are shown in Tables 3.3-3.4 and Figures 3.4-3.5.

As seen in Tables 3.3 and 3.4, the proposed framework achieves the best performance in the PaviaU dataset with only 20 training samples per class. Many algorithms cannot gain satisfactory classification result even with the aid of the spatial information. It can be noticed that the proposed method performs better than our baseline approach, where the OA is improved about 2.5% after the utilization of the online metric learning strategy. This has clearly demonstrated the efficacy of this strategy and the Mahalanobis-based distance. In the Indian Pine dataset, the designed approach also achieves the hightest OA among all compared algorithms. With the aid of the weight from the online metric learning strategy, the OA is also improved from the baseline approach. The MFASR and MASR have achieved second and third best performance on both datasets. However, both methods suffer from huge computational burden. Although the SVM classifier has the best efficiency, its classification performance is rather poor.

In this chapter, an online Mahalanobis-based metric learning strategy has been designed to acquire better matching between the training and test samples. With this mechanism, the best performance in both datasets have been achieved, and the OA has been improved in the PaviaU dataset from the baseline method's 88.98% to 91.51%, and in Indian Pine dataset from 82.38% to 83.71%. From the experimental reuslts, the learned metric can improve the discriminative ability of the SRC without high computational burden. For the learned metric, four determined parameters are discussed. For the discounting parameter β and regularization parameter η, corresponding results show that these two parameters are robust to the OA. As for the two thresholds θ_1 and θ_2 that determined the defined statement, they were set by empirically. By searching the value from

0.5 to 2, optimal parameters were chosen. From the confusion matrix of PaviaU dataset,it can be found that quite a number of samples from class 6 'bare soil' have been misclassified into the class 2 as 'meadows'. However, less samples from class 2 'meadows' are misclassified as 'bare soil' in class 6. This is possibly due to inaccurate ground truth caused by spectral mixing as there can be grasses grown in regions labeled as 'bare soil'. On the other hand, there may be also small regions of 'bare soil' in labelled 'meadows' regions. This explains the high error rate from class 6 to class 2, yet the low error rate from class 2 to class 6.

3.3 Superpixel-based Multiple Feature Fusion Sparse Representation for Spectral-Spatial Classification of Hyperspectral Images

Although the above method has achieved a good performance in HSI classification, the correlation between different features has not been fully investigated. To generate a more robust framework by fusing different features into a low-dimensional feature, the superpixel-based multiple feature fusion sparse presentation framework is proposed in this section. The flowchart of the proposed framework is shown in Figure 3.6, and the details are described below.

3.3.1 Preprocessing

As the HSI usually contains hundreds of highly-correlated spectral bands, it is essential to reduce the dimensionality of the data for more effective and more efficient analysis and classification. By applying the PCA on the raw HSI data, the

first ten principle components of HSI data are extracted as the spectral feature. To incorporate more contextual information, the EMPs have been extracted to represent the spatial feature. If the LiDAR data is available, the EMPs of the LiDAR data are also extracted as the elevation feature.

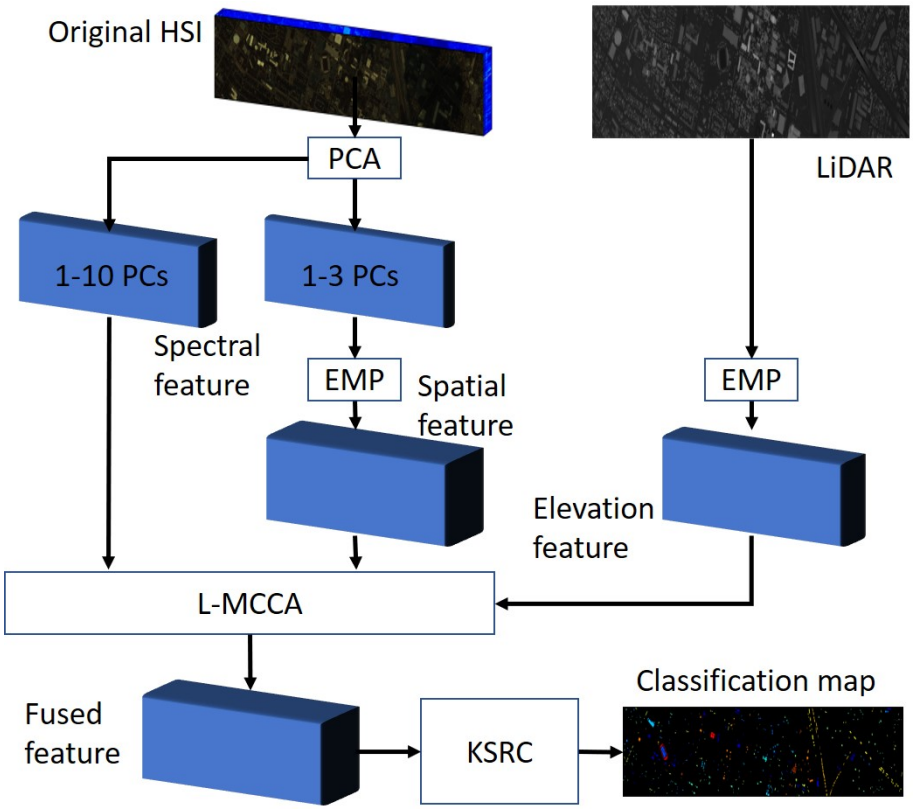


Figure 3.6: The flowchart of the proposed SMFF-SRC framework.

To obtain the superpixel map of the HSI, the SLIC [90] algorithm has been implemented on the first three components which forms a false-colored image to represent the characteristics of the raw data. As an efficient tool in computer vision, the SLIC adapts a k-means method to generate superpixels. By defining a desired number of superpixels in prior, the SLIC has initialized the cluster center by considering the size of the produced superpixels and gradient information jointly. The neighbouring pixels are assigned into the cluster center based on

their locations and distance to their neighbouring centers. Accordingly, the SLIC is more efficient than the k-means method [90]. Due to its low computational complexity, the SLIC has been employed in many applications to generate the superpixel map. In this framework, the SLIC is performed on the false-colored image of the first three principal components of the HSI to obtain the superpixel map.

3.3.2 SMFF-SRC

With the generated superpixel map, the HSI data is divided into numerous spatial regions. Similar to other SRC-based methods [63, 66], it can be assumed that the pixels within each superpixel have similar spectral characteristics thus share the same semantic class. Therefore, each superpixel is classified jointly in the proposed SMFF-SRC framework.

To address the issue of insufficient training samples, multiple features have been extracted and fused together. Although it is straightforward to normalize and stack those features together, the high dimensions of the stacked feature can result in the problem of overfitting. In [26], an adaptive weighting strategy is implemented to penalize the zero entries of the sparse coefficient vector. For the MFASR framework [27], the class of the test pixel is determined by summing the residuals from all kinds of features. In the previous subsection, an online weighting strategy is employed to evaluate the significance of the aforementioned feature. However, the performance of above methods relies heavily on the designed weighting strategy and the associations between different features have not been fully analysed. In the SRC framework, it is hard to decide the semantic label if different features are inconsistent or even contrary to each other. Even if

the classification result can be acquired by either assigning adaptive weights to features or summing the residuals from all the features, the values of adaptive weights or residuals are hardly to be well justified. To this end, the conflicts between different features in the SRC framework is solved by maximizing the mutual correlation between different features in one pixel. The extension of MCCA, the localized MCCA, is proposed to generate a fused feature-based on the extracted features, where the dimension of the fused feature is also reduced.

Let $F_{spe} \in R^{se*V}$, $F_{spa} \in R^{sa*V}$, and $F_{ele}R^{el*V}$ be the extracted spectral, spatial, and elevation features sets, respectively, where V represents the number of all labelled samples within the dataset. The dimensions of each feature vector are shown as se, sa, and el. The extracted feature can be stacked as $F = [F_{spe}; F_{spa}; F_{ele}]$. As discussed in the previous chapter, MCCA can be transformed into (2.11) and the multiple canonical correlation values μ can be computed through the generalized eigenvalue method. We can introduce the extracted features into the (2.11) and consider the estimated multiple canonical correlation values as the fused feature. However, the MCCA usually ignores the discriminative information, and the fused feature estimated from (2.11) may lack the separability, which is important for data classification in HSI. Therefore, the localized MCCA has been designed to introduce more discriminative information as detailed below.

In linear discriminative analysis-based methods [91] and the labelled MCCA model [77], the training samples are utilized to maximize the difference between different classes. By minimizing the within-class scatter matrix and maximizing the between-class scatter matrix, a more discriminative feature can be derived. Nevertheless, those strategies are not realistic with insufficient training samples.

As a result, the local information has been introduced to improve the performance of MCCA (LMCCA), which is implemented in an unsupervised manner as follows.

First, the k nearest neighbours of all pixels in the HSI are found out, where each pixel and its k nearest neighbours are considered to be likely in the same class. After stacking all extracted features of pixel v and its neighbours into a matrix $\tau_v = [F_v, F_{v1}, ..., F_{vk}]$, the within-class matrix is estimated by:

$$S = \sum_{v=1}^{V} S_v = \sum_{v=1}^{V} \theta(v)\tau_v\tau_v^T \tag{3.13}$$

where S is the combined within-class matrix, $\theta(v)$ is a distance regularization computed from the distance between v and its neighbours, and $v_1, .., v_k$ are the k neighbours of pixel v. In Equation (3.13), it can be considered that the estimated within-class matrix S can preserve the discriminative information of the extracted features, which can be also divided into three sub-matrices of dimensions S_{se}, S_{sa}, and S_{el} to represent the spectral, spatial and the elevation features separately.

Different from the labelled MCCA, the estimated within-class matrix should combine with the original one. Therefore, it simplifies (2.11) as:

$$Ow = \mu Iw \tag{3.14}$$

where

$$O = \begin{bmatrix} F_{spe}F_{spe}^T & F_{spe}F_{spa}^T & F_{spe}F_{ele}^T \\ F_{spa}F_{spe}^T & F_{spa}F_{spa}^T & F_{spa}F_{ele}^T \\ F_{ele}F_{spe}^T & F_{ele}F_{spa}^T & F_{ele}F_{ele}^T \end{bmatrix} \tag{3.15}$$

$$I = \begin{bmatrix} F_{spe}F_{spe}^T + S_{se} & \dots & 0 \\ \vdots & F_{spa}F_{spa}^T + S_{sa} & \vdots \\ 0 & \dots & F_{ele}F_{ele}^T + S_{el} \end{bmatrix} \tag{3.16}$$

By solving the generalized eigenvalue problem, the eigenvectors derived from (3.14) are used to form the transformation matrix $W = [w_{spe}; w_{spa}; w_{ele}]$ between the original feature and the fused feature. The fused feature can be determined by:

$$z = W^T F \tag{3.17}$$

The process of the proposed LMCCA is shown in Figure 3.7.

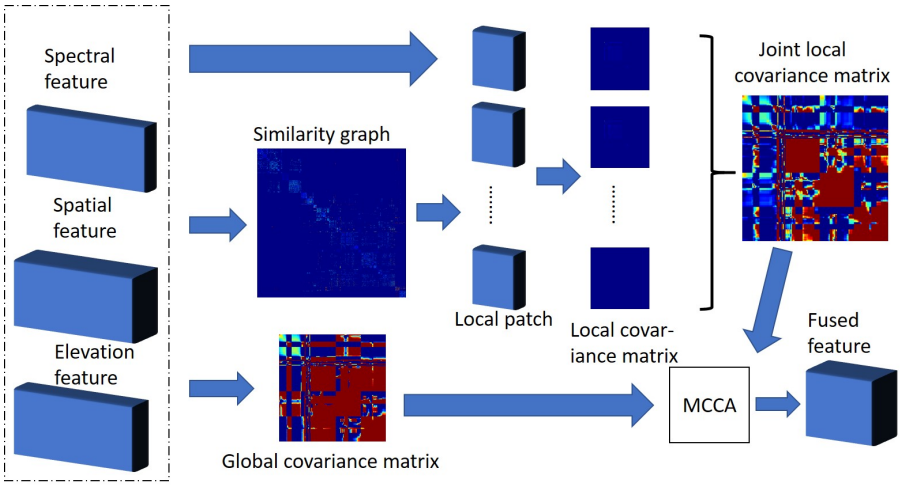


Figure 3.7: The flowchart of the proposed LMCCA

For the superpixel-based SRC, the fused feature of pixels in each superpixel are stacked into a matrix $Z_i = [Z_{i,1}, ...Z_{i,n_i}]$, where i represents the superpixel index and n_i depicts the number of pixels within the ith superpixel. Although the superpixel generation is performed in an unsupervised way, the training samples are excluded before the stacking. Similar to (3.11), the recovery of kernel-based

sparse coefficient matrix can be given by:

$$\hat{A_\phi^Z} = \arg\min_{A_\phi} ||Z_\phi - D_\phi^Z * A_\phi^Z||_2 \tag{3.18}$$

where the ϕ denotes the kernel domain along with the radial basis function as the kernel function. After the sparse coefficient matrix is estimated, the label of all pixels within the superpixel is assigned to the class with the minimum reconstruction residuals:

$$\hat{c} = \arg\max_{c=1,\ldots,C} ||Z_\phi - D_{\phi,c}^Z * A_{\phi,c}^Z||_2 \tag{3.19}$$

The whole process of the proposed SMFF-SRC is summarized in Algorithm 3.

Algorithm 3 SMFF-SRC

1: **Input:** raw HSI data and LiDAR data (if available).
2: **Feature extraction:** Apply the PCA to extract the principal components; from the first component to generate EMPs as the spatial feature; and take the first ten components as the spectral feature. Generate EMPs on the LiDAR data as the elevation feature if available.
3: **Superpixel generation:** Implement the SLIC algorithm to create superpixel map by using the first three components of HSI.
4: **LMCCA:** Fuse the extracted features into a new low-dimensional feature by combining the local information and MCCA, which is implemented by Eq. (3.13) - (3.17)
5: **Superpixel classification:** Assign the semantic label to each superpixel by using the fused feature as the input of the kernelized SRC, which is implemented by Eq. (3.18) - (3.19)
6: **Output:** Classification result.

Table 3.5: Number of training and testing samples in each class for the UH dataset.

UH dataset							
Class		Sample		Class		Class	
Label	Name	Train	Test	Label	Name	Train	Test
1	Grass Healthy	20	1231	9	Road	20	1232
2	Grass Stressed	20	1234	10	Highway	20	1207
3	Grass Synthetic	20	677	11	Railway	20	1215
4	Tree	20	1224	12	Parking-Lot 1	20	1213
5	Soil	20	1222	13	Parking-Lot 2	20	449
6	Water	20	305	14	Tennis Court	20	408
7	Residential	20	1248	15	Running Track	20	640
8	Commercial	20	1224	Total		300	14779

3.3.3 Experimental Results

Experiment Settings

Generally, the performance of HSI classification can be evaluated by three common metrics as the description in the previous chapter, including the overall accuracy (OA), the average accuracy (AA), and the Kappa coefficient. The OA denotes the percentage of corrected labelled pixels, and the AA represents the means of the classification accuracy over each class. To better reflect the reliability of the classification result, the Kappa coefficient is estimated from the confusion matrix. In this section, the proposed SMFF-SRC framework has been implemented on three datasets, the UH, PaviaU, and Indian Pine dataset.

Table 3.6: Number of training and testing samples in each class for the PaviaU dataset.

PaviaU dataset							
Class		Sample		Class		Class	
Label	Name	Train	Test	Label	Name	Train	Test
1	Asphalt	20	6611	6	Bare Soil	20	5009
2	Meadows	20	18629	7	Bitumen	20	1310
3	Gravel	20	2079	8	Self-blocking bricks	20	3662
4	Trees	20	3044	9	Shadows	20	927
5	Painted metal sheets	20	1325	Total		180	42596

During all the experiments, the training samples are randomly selected and the rest samples are considered as the test samples. For the UH dataset, three kinds of features, including the spectral and spatial features from the HSI and the elevation feature from the LiDAR data, are extracted. For other two datasets, only the spectral and spatial features are acquired as the LiDAR data is not available. In the proposed framework, there are several predefined parameters, including the number of superpixels, the dimension of fused feature, and the number of neighbours in the construction of within-class matrix. For the number of superpixels, with the consideration of the size of each HSI image and the computational efficiency, the numbers of superpixels are set to be 50000, 1000, and 250 for the UH (spatial size 349 $\times$ 1905), PaviaU (spatial size 610 $\times$ 340), and Indian Pine dataset (spatial size 145 $\times$ 145), respectively. The generated superpixels are shown in Figure 3.8-3.10. From the generated superpixel map, it can be seen that pixels are grouped into superpixels with arbitrary shape and size. The dimensions of fused features are set to be 100 for the UH dataset and 50 for the rest two datasets. The numbers of neighbours are set to be 300, 50, and 150 for the UH, PaviaU, and Indian Pine dataset, respectively. The kernel parameters of the utilized RBF function are determined by cross-validation. All the experiments are repeated 10 times and the mean values are reported. For the hardware and software, all the experiments are conducted on a 16 GB Intel i5-6500 CPU on the MATLAB 2017b.

Comparison Experiments

To demonstrate the performance of the proposed framework with insufficient training samples, the proposed framework is compared with some state-of-the-art

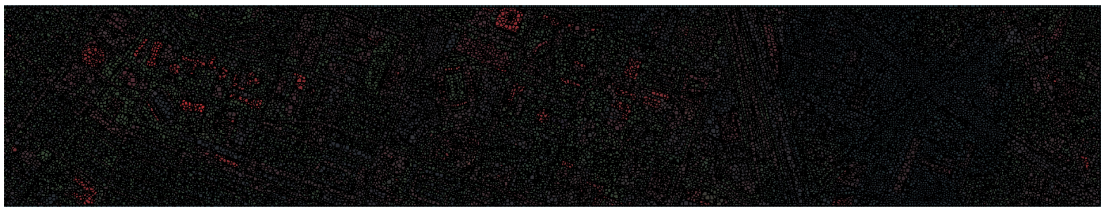


Figure 3.8: The produced superpixel map of UH dataset.

Table 3.7: Number of training and testing samples in each class for the Indian Pine dataset.

Indian Pine dataset							
Class		Sample		Class		Class	
Label	Name	Train	Test	Label	Name	Train	Test
1	Alfalfa	2	44	9	Oats	2	18
2	Corn-notill	14	1414	10	Soybeans-notill	10	962
3	Corn-min	9	821	11	Soybeans-min	25	2430
4	Corn	3	234	12	Soybeans-clean	7	586
5	Grass/pasture	5	478	13	Wheat	3	202
6	Grass/trees	8	722	14	Woods	13	1252
7	Grass/pasture-mowed	2	26	15	Bldg-gass-tree drives	4	382
8	Hay-windowed	5	473	16	Stone-steel towers	2	91
Total						114	10135

algorithms, including the support vector machine (SVM), the composite kernel support vector machine (CK-SVM) [29], the JSRC [63] and its kernelized extension JKSRC [64], the MASR [66] and the MFASR [27]. For all the SRC-based algorithms, the sparsity level is set to be 3 for saving computational burden. For the CK-SVM, MASR, and MFASR, the experiments are done with the open-source codes. For the JSRC and JKSRC, the algorithms have been implemented with the suggested parameters. For the SVM algorithm, the parameters are determined via cross-validation.

For the UH dataset and the PaviaU dataset, 20 training samples are randomly chosen from each semantic class and the rest are utilized for testing. As the number of samples in the Indian Pine dataset is rather imbalanced, 1% training samples are chosen for each class or 2 training samples if the number of samples

Table 3.8: Class specific accuracies (%) for the UH dataset.

Class	SVM	CK-SVM	JSRC	JKSRC	MASR	MFASR	SMFF-SRC
1	88.79	84.97	96.34	92.53	95.29	90.98	**98.05**
2	92.63	97.89	96.27	**99.27**	96.11	85.33	94.89
3	99.56	**100.00**	95.27	**100.00**	98.82	**100.00**	**100.00**
4	96.32	89.79	96.57	97.47	98.04	83.25	**98.77**
5	99.18	96.48	99.43	99.10	**100.00**	92.80	99.84
6	93.44	91.15	95.08	92.13	88.52	**98.36**	98.03
7	**90.22**	83.33	62.82	66.83	63.94	82.69	89.46
8	57.11	**86.76**	68.55	68.14	71.90	61.27	73.28
9	68.91	78.98	81.90	65.75	81.01	77.84	**85.15**
10	71.25	77.22	92.05	89.23	**100.00**	98.76	93.70
11	79.59	88.89	72.43	70.86	**93.33**	80.16	88.64
12	70.57	65.87	80.96	59.52	**91.34**	82.69	85.64
13	44.10	58.13	45.43	53.67	56.35	**89.53**	61.69
14	97.79	98.53	97.55	**100.00**	99.75	**100.00**	99.75
15	98.75	99.69	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA	82.64	86.05	85.23	82.51	89.25	85.97	**91.10**
AA	83.21	86.51	84.02	81.09	88.37	88.25	**90.37**
Kappa	81.23	84.91	85.38	83.63	88.96	84.85	**91.13**
Time(s)	64.23	46.78	25.10	**20.43**	367.85	159.47	32.07

Table 3.9: Class specific accuracies (%) for the PaviaU dataset.

Class	SVM	CK-SVM	JSRC	JKSRC	MASR	MFASR	SMFF-SRC
1	75.43	83.03	32.22	65.94	43.58	83.71	**85.47**
2	74.99	76.23	81.64	**90.26**	85.71	76.42	87.54
3	71.33	85.74	80.47	75.85	89.75	**95.85**	93.31
4	**94.35**	90.90	91.66	87.84	84.56	91.71	86.53
5	99.47	99.40	99.62	99.92	**100.00**	99.79	93.58
6	76.24	82.79	72.87	50.59	85.21	84.96	**94.41**
7	87.40	93.77	97.56	**100.00**	99.47	99.57	70.50
8	75.29	83.10	85.77	**97.30**	76.87	94.29	89.71
9	99.68	**99.90**	50.27	27.62	62.03	89.67	96.33
OA	78.12	81.93	74.32	80.79	78.82	84.16	**88.26**
AA	83.80	88.32	67.09	74.55	72.70	**95.65**	84.80
Kappa	72.21	76.97	76.90	77.26	80.80	79.80	**88.54**
Time(s)	**5.78**	7.18	62.78	57.82	432.35	331.87	247.66

Table 3.10: Class specific accuracies (%) for the Indian Pine dataset.

Class	SVM	CK-SVM	JSRC	JKSRC	MASR	MFASR	SMFF-SRC
1	24.44	40.91	55.56	68.18	85.68	**97.50**	93.18
2	43.07	41.94	51.56	58.49	61.12	70.12	**79.70**
3	26.92	45.55	60.41	51.04	67.69	**69.25**	66.44
4	14.96	39.32	**62.39**	24.36	61.07	39.66	**62.39**
5	**86.61**	64.64	67.36	28.24	82.26	69.08	66.11
6	80.06	88.50	92.94	**97.92**	97.42	83.47	81.86
7	77.78	84.62	77.78	92.31	99.62	96.54	**100.00**
8	87.74	68.71	97.46	97.89	**98.73**	96.41	92.60
9	31.58	22.22	52.63	44.44	86.67	**91.67**	66.67
10	43.35	49.17	70.79	74.22	78.19	**87.06**	73.60
11	66.67	71.11	74.69	70.74	85.02	**89.65**	87.41
12	30.03	17.75	57.85	56.14	53.72	73.79	**82.59**
13	96.04	89.11	84.65	98.51	96.19	**99.50**	**99.50**
14	83.31	90.42	98.00	**98.08**	96.89	92.96	97.60
15	16.75	51.05	30.63	38.22	52.71	71.28	**77.75**
16	67.39	81.32	82.61	84.62	92.93	**96.48**	71.43
OA	58.04	61.79	72.07	69.88	80.21	81.79	**82.45**
AA	54.79	59.15	68.08	65.41	77.27	**82.71**	80.04
Kappa	51.84	56.21	69.83	67.71	**81.42**	80.17	81.18
Time(s)	**1.35**	6.42	13.36	14.98	137.57	13.45	6.12

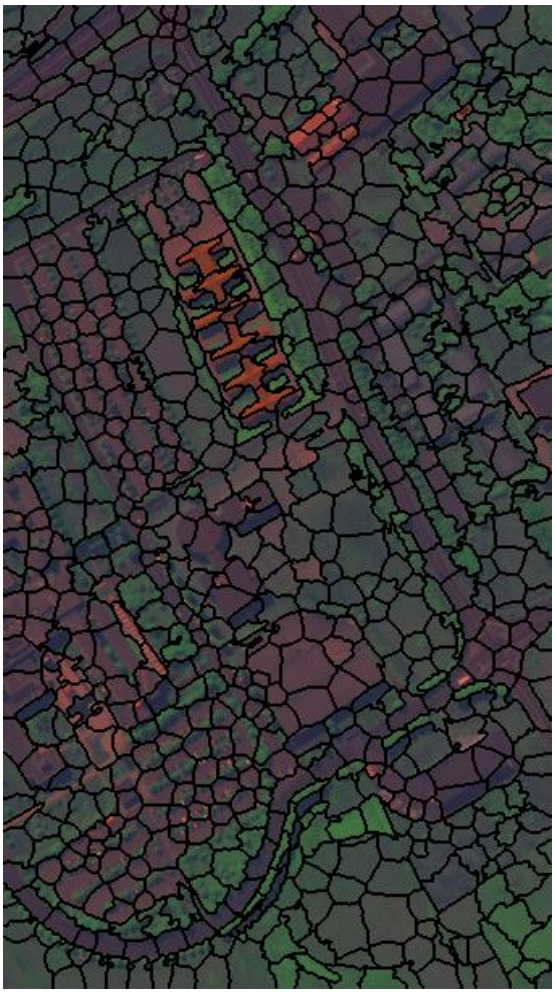


Figure 3.9: The produced superpixel map of the PaviaU dataset.

in that class is below 200. The distributions of training and testing samples for three datasets are shown in Table 3.5-3.7. The comparison experiments results are shown in Figure 3.11-3.14 and Table 3.8-3.10, respectively.

As seen in Figure 3.11, 3.12 and Table 3.8, the proposed SMFF-SRC method has achieved the best performance among other state-of-the-art algorithms with only 20 training samples per class in the UH dataset. Although the MASR has obtained a fairly good OA, its huge computational burden is inevitable. By applying the superpixel-based SRC, the proposed method is more efficient. For the

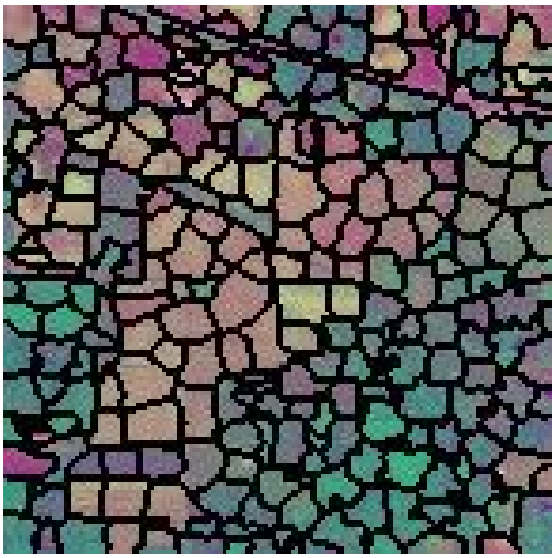
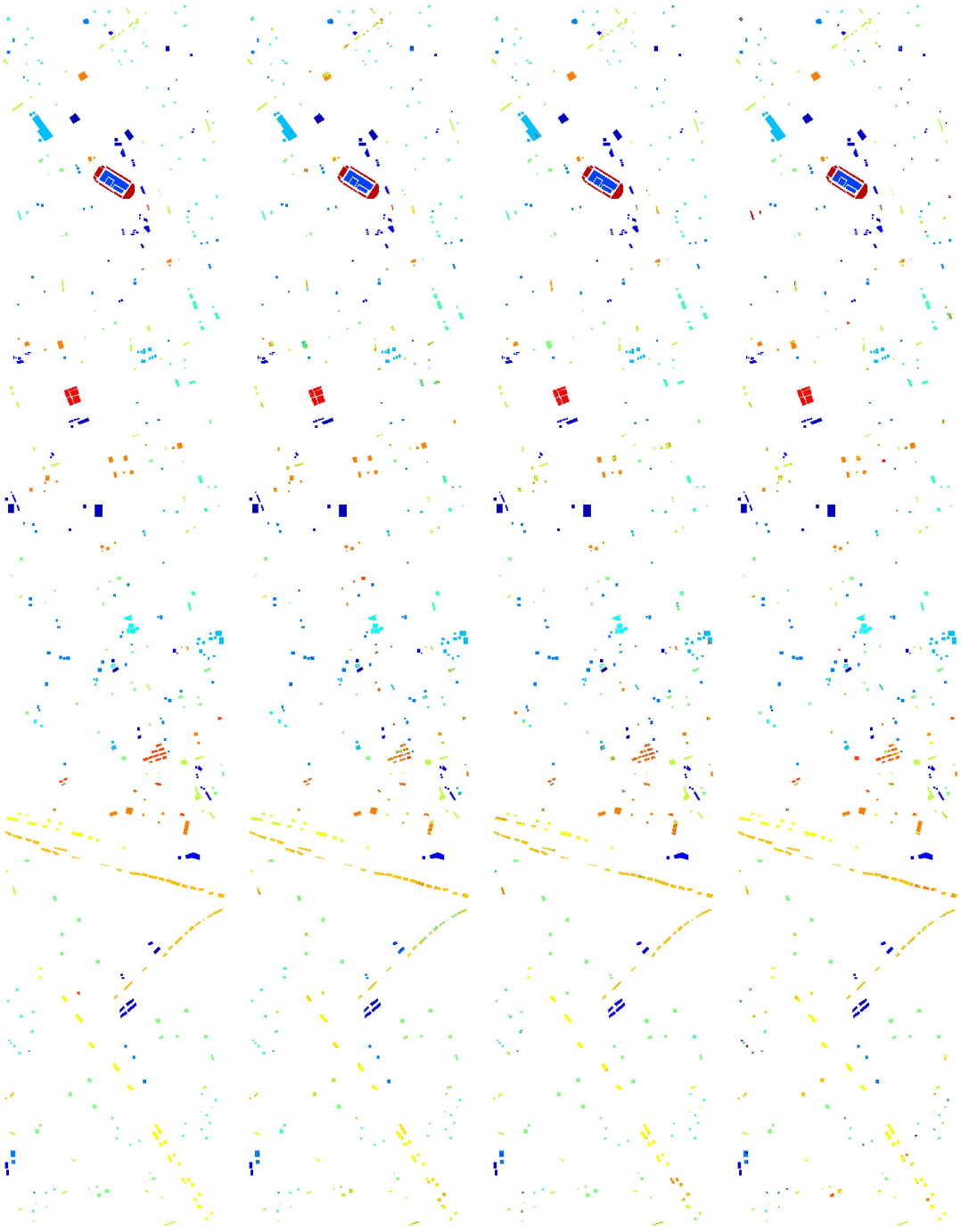


Figure 3.10: The produced superpixel map of the Indian Pine dataset.

JKSRC method, its computational time is the shortest, but its classification accuracy is the worst among the compared methods. For the class-specific accuracies, the proposed method has a more robust performance even if not achieved the best accuracy in class 8 'commercial' and 13 'Parking Lot 1'.

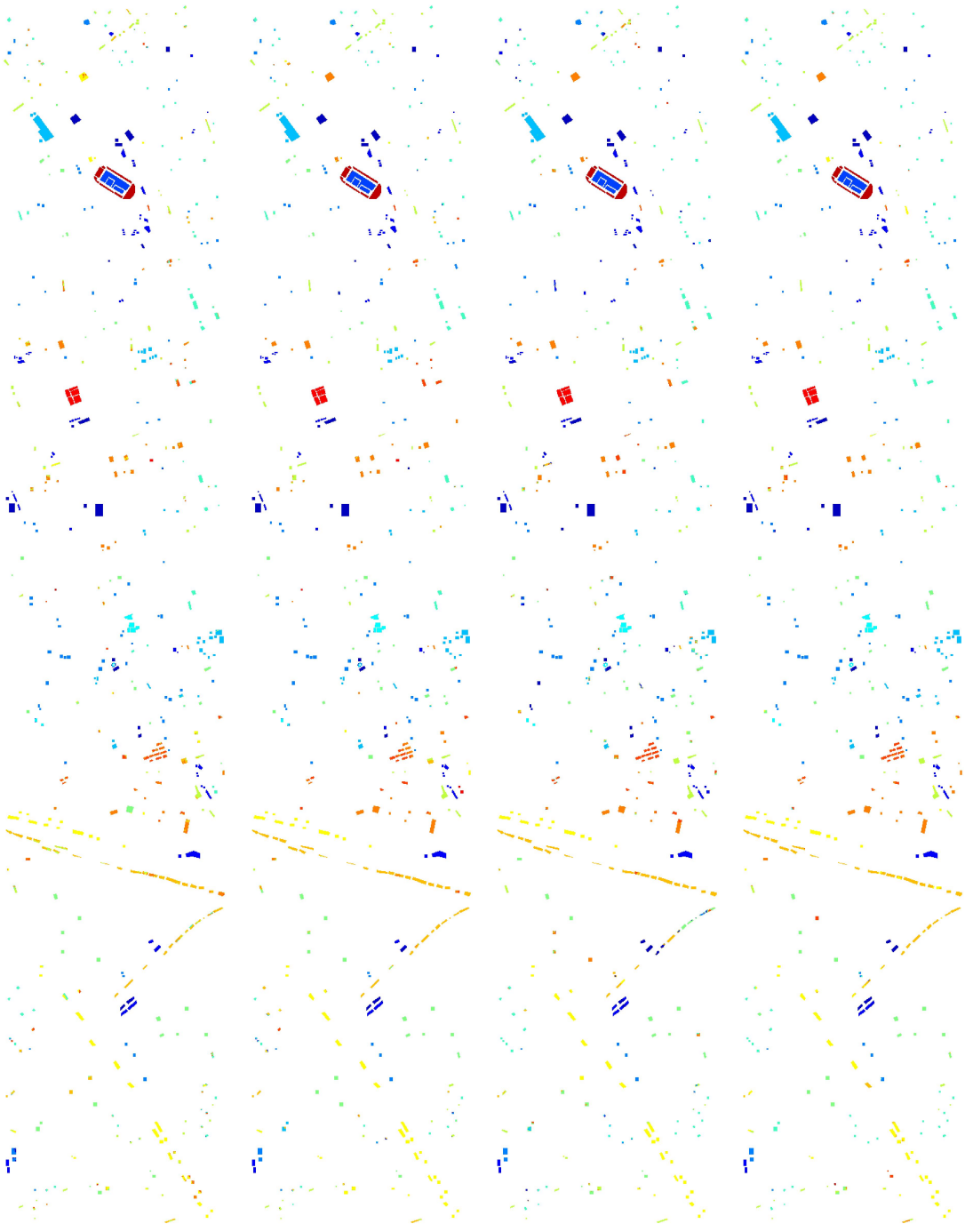
The performance of the comparison results on the PaviaU dataset with 20 training samples per class is shown in Figure 3.13 and Table 3.9. We have achieved the best performance among all compared methods. Although the MFASR has the best AA, its poor performance on the second class 2 'Meadows' cannot be ignored, which determines the OA of MFASR is not robust as class 2 has the most testing samples. SVM-based methods have the best efficiency compared to other methods. Although Our proposed SMFF-SRC framework has a larger computation burden than the SVM-based methods, the designed framework have an obvious leading in the classification accuracy.

Fig. 3.14 and Table 3.10 show the classification performance on the Indian Pine dataset. Our proposed method has achieved good performance with nearly



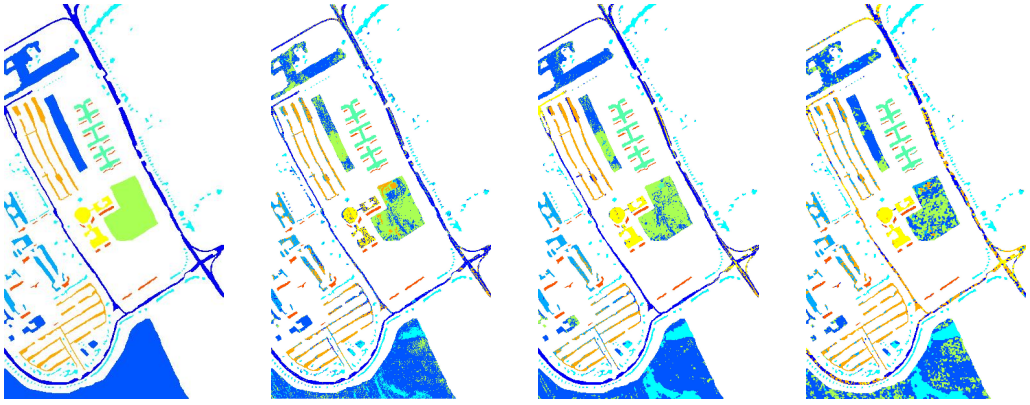
(a) GT (b) SVM (c) CK-SVM (d) JSRC

Figure 3.11: The classification map of the UH dataset. (a)the Ground Truth (GT), (b) SVM, (c) CK-SVM, (d) JSRC

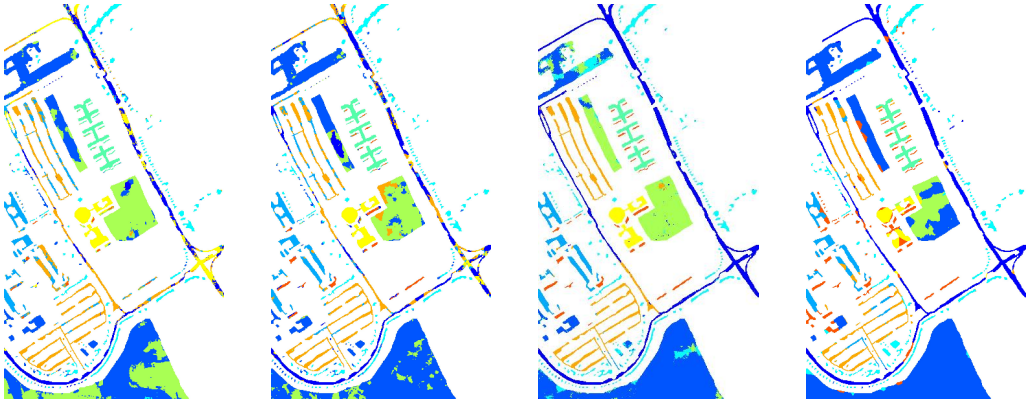


(a) JKSRC (b) MASR (c) MFASR (d) Ours

Figure 3.12: The classification map of the UH dataset. (a) KSRC, (b) MASR, (c) MFASR, (d) Ours(SMFF-SRC).

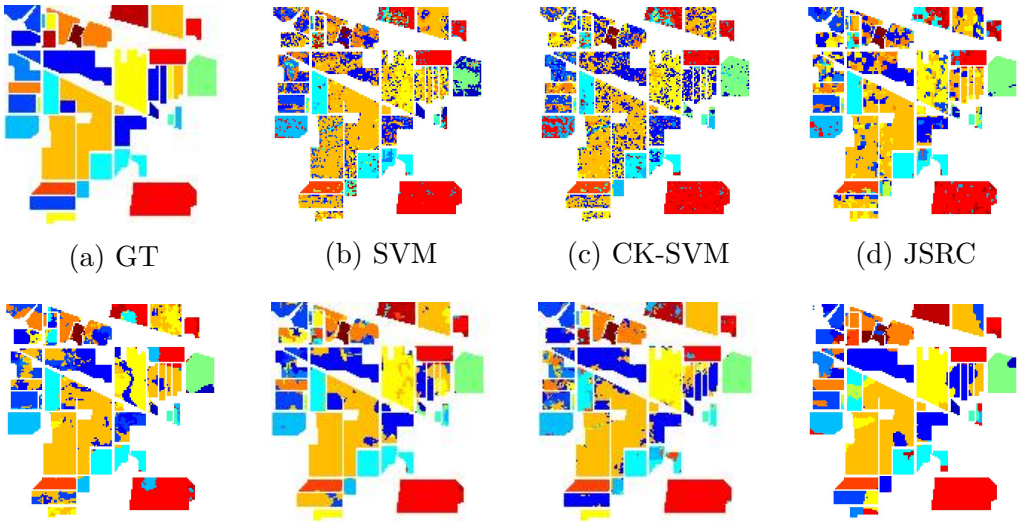


(a) GT (b) SVM (c) CK-SVM (d) JSRC



(e) JKSRC (f) MASR (g) MFASR (h) Ours

Figure 3.13: The classification map of the PaviaU dataset. (a)the Ground Truth (GT), (b) SVM, (c) CK-SVM, (d) JSRC, (e) KSRC, (f) MASR, (g) MFASR, (h) Ours(SMFF-SRC).



(a) GT (b) SVM (c) CK-SVM (d) JSRC

(e) JKSRC (f) MASR (g) MFASR (h) Ours

Figure 3.14: The classification map of the Indian Pine dataset. (a)the Ground Truth (GT), (b) SVM, (c) CK-SVM, (d) JSRC, (e) KSRC, (f) MASR, (g) MFASR, (h) Ours(SMFF-SRC).

1% training samples in each class. The MFASR has the best performance on the AA, but its computational time is about double ours. The MASR has achieved a good classification performance and the best Kappa coefficient, but it has the largest computational burden. The SVM method has the best efficiency but its classification performance is not robust.

In summary, the proposed SMFF-SRC method has the most robust performance on the three public datasets with insufficient training samples. From the experimental results, it can be clearly found out the proposed method can improve the discriminative ability of SRC without resulting in too much computational burden. Although the MASR and MFASR have a relatively good performance, their computational burdens are huge, especially the MASR. Furthermore, the proposed method is more stable than other methods in these three datasets. For example, the performance of CK-SVM on the UH and the PaviaU datasets are

good, but its performance on the Indian Pine dataset is not robust. The reason is that the CK-SVM utilize the cross-validation on the training samples to search the best parameters, the ideal parameters are not easy to find with insufficient training samples. Therefore, the efficiency and robustness of the proposed SMFF-SRC can be demonstrated.

3.4 Summary

In this chapter, two superpixel-based sparse representation classification algorithms are introduced for HSI. A superpixel-based feature specific SRC framework is firstly proposed to fully exploit the spectral-spatial features of the HSI. With the generation of a superpixel map for each HSI, the efficiency of the proposed SPFS-SRC method can be guaranteed. Then, a kernel-based SRC-based classifier is designed to assign each superpixel into certain semantic class. The proposed SPFS-SRC method has proved its superiority from extensive experiments. Different SPFS-SRC, the SMFF-SRC aims to investigate the joint cooperation of all extracted features in remote sensing scenario. The proposed LMCCA helps to fuse the extracted feature with the introduction of local information. The proposed SMFF-SRC method has also been compared with several state-of-the-art methods to show its effectiveness.

Chapter 4

Adaptive Distance based Band Hierarchy (ADBH) for Unsupervised Hyperspectral Band Selection

4.1 Introduction

Based on certain selection strategies, UBS methods aim to select the most representative bands among the HSI data. Recently, many searching strategies have been developed for HSI band selection, which can be separated into two main groups: the ranking-based and the clustering-based methods. Various statistical metrics have been utilized to evaluate each band in the ranking-based methods, including mutual information [92, 93], variance [94] and local density [52], etc. After the band ranking, the desired band subset is determined by selecting bands

with higher ranking values among all bands. Since the ranking process is only implemented once, the computational cost can be rather low. For the clustering-based methods [50,51,53,54], spectrally continuous bands are grouped into desired clusters. Bands in each cluster are contiguous and with similar spectral information, where the most significant band in each cluster based on discriminative ability [51] or some ranking strategies [53] are selected to form the desired band subset. Due to the clustering procedure, this process can be lengthy whilst the selected bands are generally uncorrelated.

Although the aforementioned two groups of UBS methods have achieved certain success for band selection in HSI, both of them still suffer different drawbacks. For ranking-based approaches, the correlation between the selected bands is usually quite high, where the data redundancy can be further reduced. On the contrary, the clustering-based methods usually select one band from each band cluster, thus the data redundancy is low. However, most of the clustering-based methods are very sensitive to the noisy bands because a noisy band can easily form a cluster due to low similarity to other bands thus affect the selection result. Meanwhile, the results of band selection depend on the clustering process, especially on the number of clusters. For example, a certain band can be selected when the number of clusters is three but it can then be deselected when the number of band clusters becomes five, where such inconsistency may lead to low robustness of UBS. Furthermore, the similarity metric between different bands plays a key role in clustering methods, including the efficacy and computational complexity. Some clustering-based methods may have a good performance, but their computational cost can be high due to the complicated metrics.

To tackle the aforementioned drawbacks, a band hierarchy clustering UBS

framework with adaptive distance (ADBH) is proposed. The rest of this chapter is organized as follows. Section 4.2 introduces the proposed methodology. In Section 4.3, experimental results and discussions are given on four HSI datasets. A brief summary will be given in the end of this chapter.

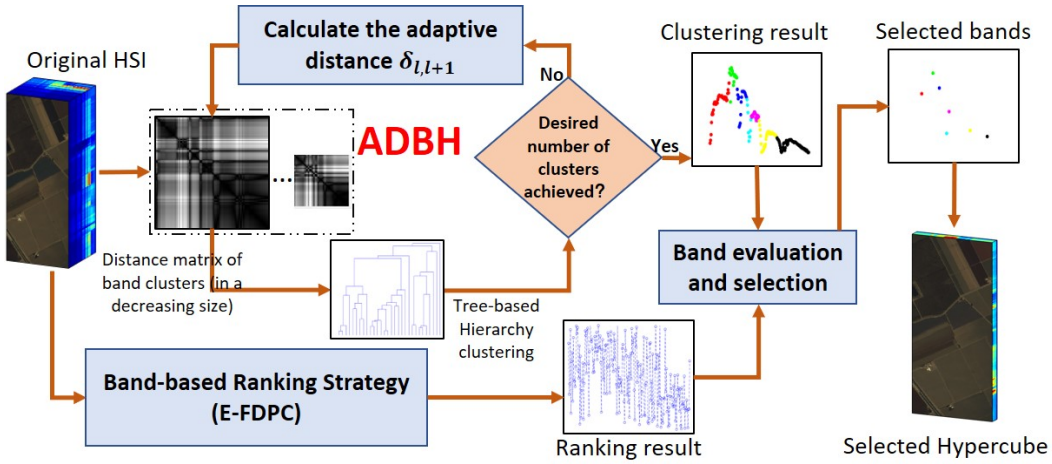


Figure 4.1: The flowchart of the proposed ADBH framework

4.2 Proposed Method

In this section, the proposed ADBH framework for UBS will be presented in detail. First, the tree hierarchy-based clustering strategy is described. Followed by the adaptive distance measurement within the ADBH framework, which is based on the multiplication of the Euclidean distance and cluster density. Afterwards, the band evaluation and selection method is introduced. Finally, the advantages of ADBH are analysed.

Figure 4.1 illustrates the flowchart of proposed ADBH framework. In the proposed framework, the raw HSI dataset is taken as input for both band clustering and band-based ranking. At first, each spectral band is considered as a cluster to

form the initial similarity matrix, from which a tree-based band hierarchy can be constructed. Cluster-based adaptive distance (AD) is then calculated, and mutual neighbouring clusters are merged sequentially according to the determined AD. Afterwards, the similarity matrix will be updated, which actually becomes smaller due to the merged band clusters. The process above forms the proposed ADBH, where the process continues until the number of band clusters reaches the desired number of the selected bands. Relevant bands within the resulted band clusters will be ranked by the band-based ranking strategy (E-FDPC) before band selection. The band with the highest ranked value within each cluster is selected as the most representative band for that cluster, and all the selected bands are then grouped to form a dimension-reduced hypercube for following-on processing and analysis.

4.2.1 Band hierarchy

The clustering-based UBS methods aim to group similar bands into each cluster and select one most significant band from each cluster, which can reduce the data redundancy between the selected bands. Due to the lack of ground truth, the number of band clusters and the exact indexes of bands for each cluster are unknown. As a result, the results of band clustering and the derived band subset become arbitrary, where the consistency of the results can hardly be maintained. To tackle this particular challenge, in this work, a band hierarchy algorithm is proposed. Our method can construct a band hierarchy in a bottom-up manner and generate any number of band clusters (between one and the original amount of bands). As such, a better understanding of the HSI bands can be derived. Moreover, the clustering results can keep consistency despite of various number

of bands are chosen. For instance, with desired k bands, the tree hierarchy can produce k clusters in an iterative way. When a band group with $k-1$ groups is requested, the result will be adjusted in a flexible way by merging two clusters. Similarly, the result can be easily adjusted to $k+1$ groups by cancelling the last merging operation. For iteration-based methods, the computational burden is a common challenge. For efficiency, complicated metrics or complex strategies are avoided in the ADBH framework as explained below.

Let us denote a HSI image as $Y \in R^{M \times N \times L}$, where the spatial size of this cube is $M \times N$ and L is the total number of bands. The lth band can be represented as one vector $Y_l \in R^{1 \times M*N}$ and the spectral signature of one pixel at the spatial location (m, n) can be denoted as $Y_{mn} \in R^L$. To reduce the computational cost, the spectral value of each pixel is normalized to the scale of [0, 1]. Let $G = (V, E)$ denote the HSI data in an undirectional graph, where the node set $V = [1, 2, ..., l, ..., L]$ represents the spectral bands in the HSI dataset. Considering the whole dataset as a forest, each band can be considered as a tree, i.e. each band is an individual cluster initially. E is the utilized similarity metrics to measure the connection of different clusters (bands). Due to the contiguous nature of the spectral bands in HSI, each band is assumed to be more closely linked to its neighbouring bands in the spectral domain. To this end, $E = [e_1, ..., e_l, ..., e_Z]$ represents the linkage between different clusters, where e_l represents the 'edge' between the lth cluster and the $(l+1)$th cluster with $1 \leq Z \leq (L-1)$. Besides, for the first cluster and the last cluster, they only have one edge to connect with their neighbours according to the assumptions above. As a result, it is not necessary to estimate the similarity matrix between bands after each iteration instead of computing similarities between neighbouring band clusters. After that,

the developed tree hierarchy clustering in a bottom-up manner is detailed as follows.

First of all, a 'mutual nearest neighbouring' is defined according to the similarity between each cluster, which is very similar to the mutual nearest neighbours defined in [95]. By examining all connecting edges of each cluster, two clusters can become 'nearest neighbour' when they both have lighter edge with each other. For example, if $e_l < e_{l-1}$, the lth cluster is closer to the $(l+1)$th cluster, but the lth cluster and the $(l+1)$th cluster can be 'nearest neighbours' only if $e_l < e_{l+1}$ is also met. This criterion can identify similar clusters pairs and can be utilized in the following-on merging procedure.

After the 'mutual nearest neighbouring' search, the current clusters can be started to merge. To reflect and be consistent with the data structure of the HSI dataset, the merging is executed in a sequential way. With all the obtained pairs of clusters, the implementation starts from the pair with the shortest edge. Different from some clustering methods which merge the data sample points gradually [96] (i.e. one merging operation in one iteration), each iteration of the algorithm will not be completed until all the mutual neighbouring clusters are merged, i.e. merging all such band pairs simultaneously. For each new cluster, it is depicted by the mean spectral information of its comprised bands and the previous bands are removed while the spectral information is kept for after iterations. This is shown below:

$$\hat{Y_l} = mean(Y_{merged}) \tag{4.1}$$

the representation of the new lth cluster is the mean of all bands it contained. With new clusters, the defined E will also be updated before next iteration. The

number of contained bands in each new cluster is also stored. In the case that no nearest neighbour pairs exist, the ADBH framework will merge clusters gradually with only one merging in one iteration. In this situation, two clusters with the shortest edge will be merged. The whole clustering procedure will continue iteratively until the desired number of clusters has reached. As the purpose of the clustering step is to group similar bands together, the objective function can be transformed to minimize the cost function during clustering:

$$min \sum_{t=1}^{T} e^t \tag{4.2}$$

where the $t = [1, ..., T]$ is the evolution time and the e^t is the sum of merged e during the tth iteration.

In the clustering part, the bottom-up manner considers each band as an initial cluster, where the analogous bands can be determined via the defined 'mutual nearest neighbouring' approach. In each iteration, all the neighbouring pairs of bands can be merged simultaneously, and a stepping method is employed to combine clusters in case of such neighbouring pair of bands remains in certain iteration. This iterative process will only stop after the requested number of clusters have been reached. An example of ADBH clustering process is shown in Figure 4.2.

4.2.2 Adaptive distance

Although the tree hierarchy method can help to understand the data structure of bands within HSI, noisy bands are still a serious problem in all hierarchy-based clustering methods. As the bands are clustered in a bottom-up manner,

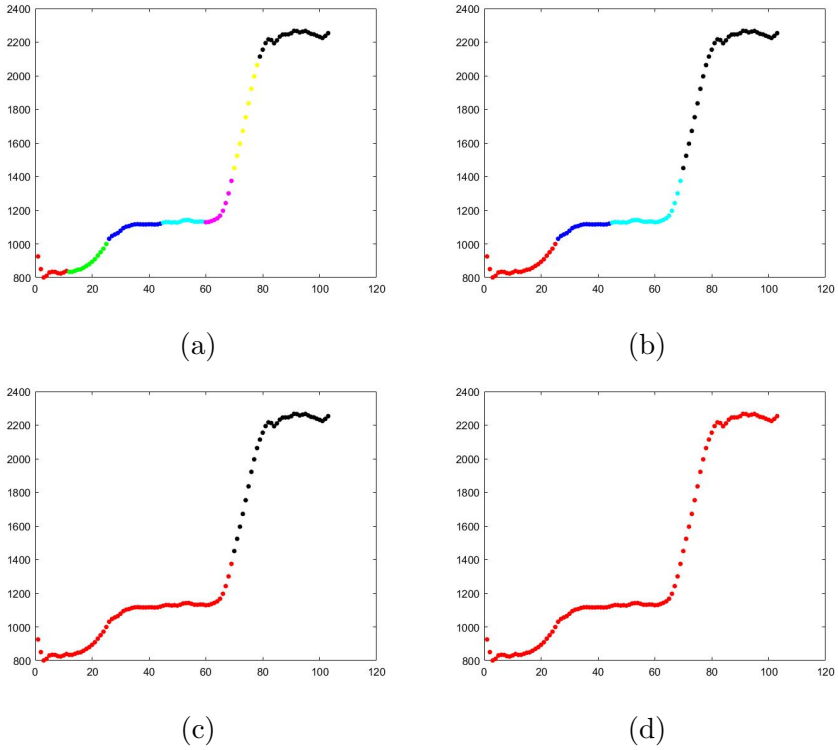


Figure 4.2: The Clustering results with different desired number of clusters on the Pavia University dataset. In each figure, the horizontal axis represents the Band Index, and the vertical represents the mean spectral value. Different color represents different clusters (a) 7 clusters, (b) 4 clusters, (c) 2 clusters, (d) 1 cluster.

potential outlier of bands can be easily identified as a primary cluster in a similar way as other bands. The outlier is prone to forming a cluster even after numerous iterations because it is less correlated or similar to its neighbouring bands in the band hierarchy. Since the final result consists of bands selected from each cluster, it is inevitably that noisy bands may be added into the selected band subset. Besides, the distance measurement for inspecting the similarity between bands is another crucial issue in the ADBH hierarchy. As the distance measurement needs to be updated in each iteration, a complicated one may result in huge computational burden. Thus, an efficient yet robust distance is introduced in the

ADBH band hierarchy as detailed below.

To estimate the differences between two variables, the Euclidean distance is regarded as one fundamental metric. In most of clustering work, the Euclidean distance is widely used to assess the differences between variables [97,98]. In [73] and [52], distances of different bands in HSI are measured using the Euclidean distance to form a distance matrix $S \in R^{L \times L}$ as:

$$S_{ij} = ||Y_i - Y_j||^2 = \sum_{m,n=1}^{M,N} (Y_{mni} - Y_{mnj})^2 \tag{4.3}$$

where the entry S_{ij} represents the difference between the ith band the jth band. According to matrix S, a scaled distance can be obtained as [52,73]:

$$D_{ij} = \sqrt{S_{ij}} * L^{-1} \tag{4.4}$$

In the proposed ADBH framework, the aforementioned distance is applied by setting $e_l = D_{l,l+1}$. However, the obtained result shows that the Euclidean distance is unstable for noisy datasets, for instance, the highly polluted KSC dataset. By only applying the Euclidean distance, it is likely to have the noisy bands as separate clusters because these noisy bands are usually sufficiently dissimilar to other neighbouring bands. To tackle this issue, a novel adaptive distance (AD) is proposed for measuring the distance of bands by considering the number of bands within the associated cluster.

Basically, there are two motivations for designing the AD. The first is to restrict or even avoid a-single-band cluster formed by noisy bands as it will interfere the results of band selection. The second is to improve the computational efficiency especially during the iterative process of band clustering. Inspired by the

above two motivations, a novel metric has been designed to estimate the distance between two adjacent clusters instead of adopting the Euclidean distance. As a regular cluster usually has more than one band, the number of the contained bands is considered as a crucial metric to present the density of each cluster. To effectively represent the characteristic of each cluster, the Euclidean norm of each cluster is also estimated. The Euclidean norm of one cluster $\hat{Y}_l$ in (4.1) corresponds to the average magnitude of this cluster, which can be assumed as a simple data characteristic of $\hat{Y}_l$. Considering the representation of each cluster as a vector, it can be found out that the product of its magnitude and contained bands can reflect its strength. In this way, the cluster density can be determined by both the number of the contained bands and the data characteristics in each cluster. Accordingly, a novel measurement is defined for estimating the cluster density I_l:

$$I_l = norm(\hat{Y}_l) * b_l \tag{4.5}$$

where b_l is the number of contained bands in the lth cluster, which has an initial value of $b_l = 1$. For a cluster with a single band, I_l is the Euclidean norm of that band. Otherwise, I_l is roughly the accumulated Euclidean norm of all the bands within the cluster. With more bands contained in a cluster in the proposed band hierarchy, the cluster density increases in nearly a linear way.

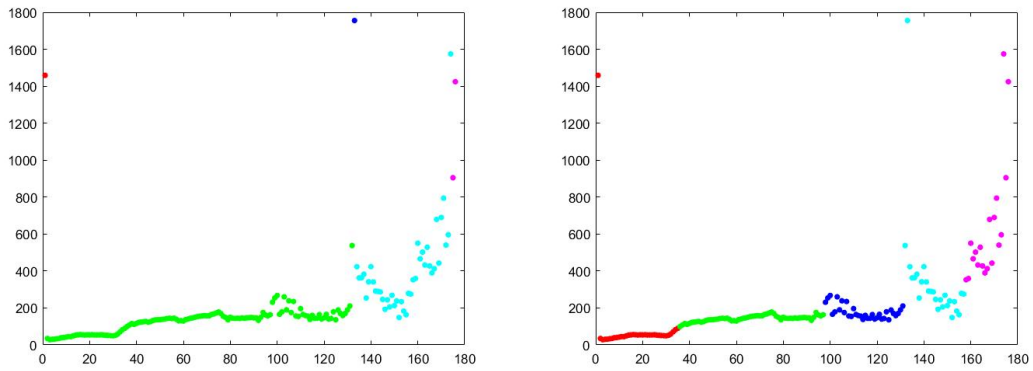
For two neighbouring clusters (the cluster can be a single band before the iterative process) l and $l+1$, their densities are denoted as I_l and I_{l+1} according to (4.5). The defined AD $\delta_{l,l+1}$ is given by combining the Euclidean distance and cluster density as:

$$\delta_{l,l+1} = D_{l,l+1} * I_l * I_{l+1} \tag{4.6}$$

From this proposed distance, a cluster with a lower density will have shorter distance with its adjacent clusters comparing to other clusters with larger densities. As shown in Figure 5.3, the first band of the KSC dataset has a distinct spectrum against its neighboring bands, thus it can be easily regarded as an outlier in the dataset. In Figure 4.3 (a), this band is considered as a single-band cluster when the Euclidean distance is used to measure the distance between band clusters. Accordingly, this band will be selected because it is the only representative band within the cluster. However, in the proposed AD scheme, this band will be suppressed and grouped into other clusters. During the AD based clustering process, the density of a single band cluster will be relatively small due to the fact that it contains only one band. By applying $e_l = \delta_{l,l+1}$ into the band hierarchy, e_l will become quite small thus for the cluster with less bands can easily find its mutual nearest neighbour. As a result, noisy bands will be simply merged in the proposed ADBH hierarchy, which also meets the energy minimization principle according to (4.2). Compared to the commonly used Euclidean distance, the proposed ADBH combines the Euclidean distance with the cluster density, in which the cluster density is estimated by multiplying the Euclidean norm of the mean band and the number of bands contained in the associated cluster. In this way, the computational complexity of the proposed AD is further reduced for efficiency. In addition, for a cluster with noisy band being merged, the representative band can be selected by avoiding these noisy bands with the E-FDPC band ranking scheme, which is further detailed in the next subsection.

Algorithm 4 ADBH

Input: Raw HSI data Y, desired number of bands K.
Initialize: Assume each band as a cluster.
BEGIN
while Number of clusters $> K$ **do**
  Update the AD among clusters by (4.5) and (4.6);
  if Mutual neighbouring clusters exist **then**
    Merging mutual neighbouring clusters pairs sequentially according to their edge;
    Update new cluster;
    if Current Number of clusters $= K$ **then**
      Return clustering result C;
      Break;
    end if
  else
    Merging two clusters with lightest edge;
    Update new cluster;
    if Current Number of clusters $= K$ **then**
      Return clustering result C;
      Break;
    end if
  end if
end while
Choose band subset X among clustering result C by (4.7).
Output: Band subset X.
END



(a) (b)

Figure 4.3: The Clustering results (defined cluster number equals to 5) by Euclidean distance (a) and the proposed AD (b) on the noisy KSC dataset. In each subfigure, the horizontal axis represents the band index and the vertical axis the mean spectral value. Different color represents different clusters.

4.2.3 Band evaluation and selection

In the proposed ADBH, the whole dataset can be grouped into several clusters of bands with similar characteristic. To select the most representative band from each band cluster, the ranking or priority of each band needs to be determined. Recently, many metrics [52, 92–94] have been utilized for this purpose. Among those criteria, E-FDPC is employed as it provides an efficient solution for determining bands with high discriminative ability. Due to the fact that a band which has large local density can be more easily chosen than others [52, 53], E-FDPC is robust to the noisy bands. Although the E-FDPC is still substantially a ranking-based method, the combination of E-FDPC and the clustering process has proved to be effective [53]. Therefore, the E-FDPC algorithm is applied after band clustering work, where the most vital band within each band cluster can be chosen to form the desired band subset. This ranking-based strategy is described as follows:

Denote the clustering result as $C = [c_1, ..., c_k, ..., c_K]$, where c_k is the kth cluster and $k = [1, ..., K]$ is the cluster index with the desired number of bands equalling to K. As the band with the highest ranking value in each cluster is the most vital one, the desired band X_k from the kth cluster can be determined as:

$$X_k = \arg\max_{\psi} \psi_{k_v} \tag{4.7}$$

where ψ is the rank values set for all bands and ψ_{k_v} is the rank value of the vth band in the kth cluster. The band with the highest rank value in the kth cluster is chosen as a band for the desired band subset X. Obviously, the band selection result can be decided with the aid of the proposed ADBH.

4.2.4 Merits of ADBH

With the designed adaptive distance, the ADBH helps to complete the UBS task in a bottom-up tree hierarchy. As the merging process starts from the shortest edge, the sequence can be recorded and the band clustering process can be visualized easily. In Figure 4.2, part of the clustering process from the ADBH of the Pavia University dataset is shown. We have chosen results from certain numbers of clusters to verify the consistency. This advantage may help to further understand the HSI dataset, where any desired number of bands can be easily determined. Secondly, the designed ADBH framework can be regarded as a parameter-free method, which means no other input parameters are needed except only the desired number of bands along with the raw data. Besides, the clustering result will not be affected by varying the requested number of clusters, where the consistency can always be kept. Finally, the clustering results can be

improved with the defined similarity metric, i.e. the AD, which is verified on the KSC dataset in Figure 4.3. It can be seen that the single band cluster is removed after appling the AD into the tree hierarchy, which has successfully suppressed the noisy band being chosen as part of the selected band subset. The proposed UBS framework is summarized in Algorithm 4, and some further experimental results are discussed in the next section to demonstrate the efficacy of the proposed ADBH method for UBS in HSI.

4.3 Experimental Results

Due to the lack of ground truth, the efficacy of band selection is often indirectly evaluated by using the classification accuracy with the selected bands. In experiments, the proposed ADBH framework is benchmarked with several SOTA algorithms based on the classification results from four popular HSI datasets. Relevant details are presented as follows.

4.3.1 Settings

To evaluate the performance of the ADBH framework in HSI classification, the ADBH framework is compared with SOTA algorithms, including OCF (TRC-OC-FDPC) [53], VGBS [55], DSEBS [57], WaLuDi [50], WaLuMi [50], E-FDPC [52] and ASPS [54]. It is worth noting that the ADBH algorithm is parameter-free, only the HSI data and the desired number of bands are needed as input. Similarly, OCF does not have any determined parameters and experiments are implemented on code provided by authors. For other methods including VGBS, DSEBS, WaLuDi, WaLuMi, E-FDPC, and ASPS, experiments are tested on origi-

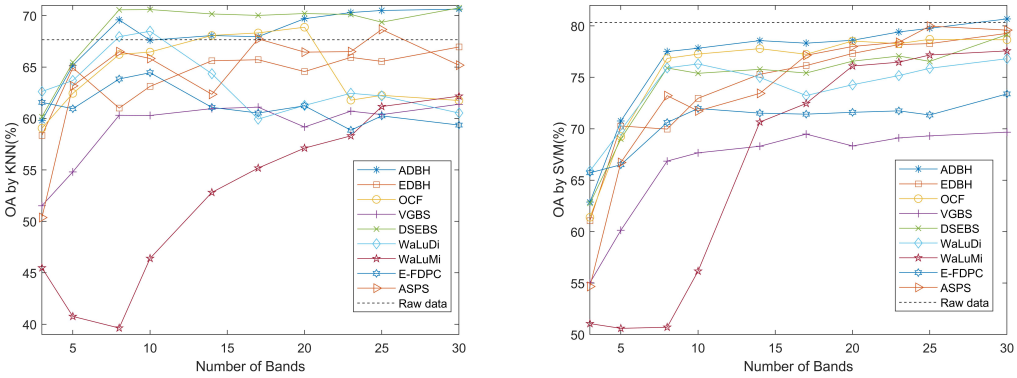
nal codes with default parameters. To better investigate the effect of the proposed AD, the method employing the Euclidean distance instead of the proposed AD is also implemented, which is represented as euclidean distance-based band hierarchy (EDBH). To better verify the effectiveness of the proposed ADBH framework, the classification results using all bands (shown as 'Raw data' in corresponding tables and figures) are also included.

For the classification part, two popular classifiers, K-Nearest Neighbourhood (KNN) [99] and Support Vector Machine (SVM) [100], are employed to validate the classification accuracy of the chosen band subsets on classification of the aforementioned four HSI datasets. In experiments, the parameters in SVM and KNN are optimized through 10-fold cross-validation. In all four HSI datasets, 10% of the samples from each class are randomly selected as the training samples for both classifiers, whilst the rest of samples are used for testing. The experimental results are shown in the next subsection. All the experiments are repeated 10 times, where the average metrics are reported for comparison. For hardware and software settings, all experiments are implemented on the MATLAB 2018b with a 16GB Intel i5-8400 CPU.

4.3.2 Comparison Experiments

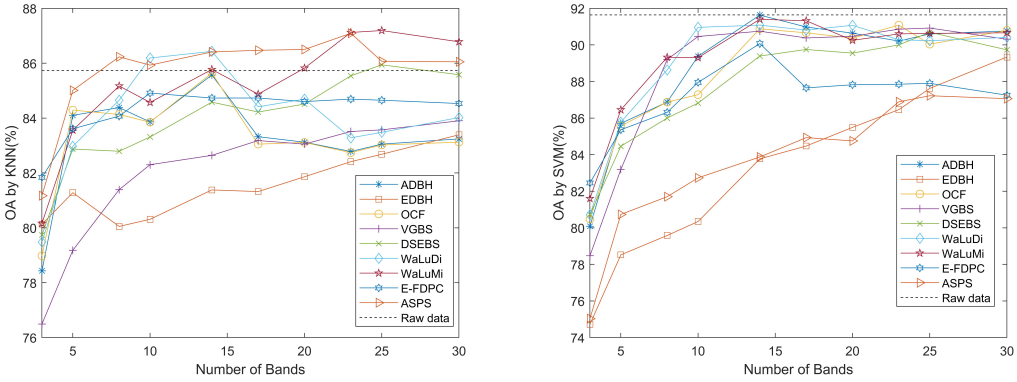
Table 4.1: Classification results from different approaches for the Indian pines dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	ASPS	EDBH	ADBH	Raw data
OA by KNN(%)	68.07±0.01	60.96± 0.01	**70.16±0.01**	64.35±0.01	52.81±0.00	61.08±0.01	62.35±0.01	65.15±0.01	68.06±0.01	67.65±0.01
AA by KNN(%)	58.27±0.02	48.39±0.00	56.05±0.01	51.22±0.00	40.42±0.00	46.68±0.01	49.57±0.01	52.86±0.01	**58.43±0.01**	54.22±0.01
Kappa by KNN	0.63±0.01	0.55±0.01	**0.66±0.01**	0.59±0.01	0.45±0.00	0.55±0.01	0.57±0.01	0.60±0.01	0.63±0.01	0.63±0.01
OA by SVM(%)	77.79±0.01	68.30±0.01	75.78±0.01	74.99±0.01	70.65±0.01	71.52±0.01	73.44±0.01	75.3±0.01	**78.52±0.01**	80.33±0.01
AA by SVM(%)	76.82±0.01	64.53±0.02	74.96±0.01	75.58±0.01	67.35±0.01	70.57±0.02	73.50±0.01	73.30±0.01	**77.75±0.01**	72.09±0.01
Kappa by SVM	0.75±0.01	0.64±0.01	0.72±0.00	0.72±0.01	0.67±0.01	0.67±0.01	0.70±0.01	0.72±0.01	**0.76±0.01**	0.78±0.01



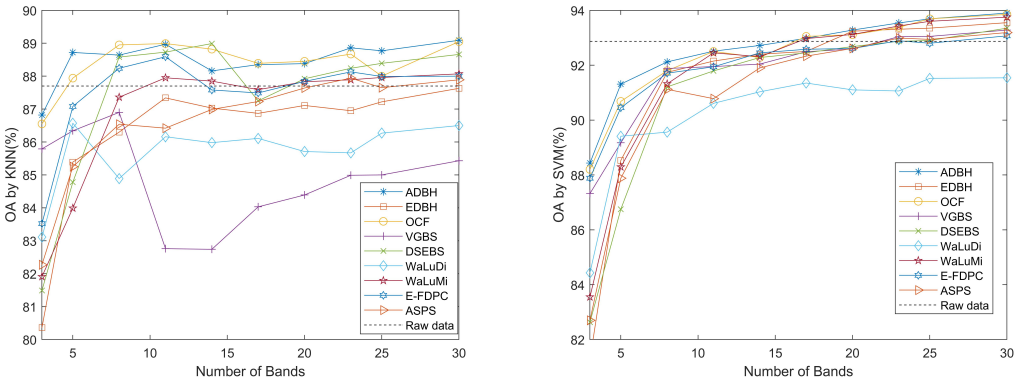
(a) (b)

Figure 4.4: OA curves on the Indian pines dataset with different UBS methods by using KNN (a) and SVM (b).



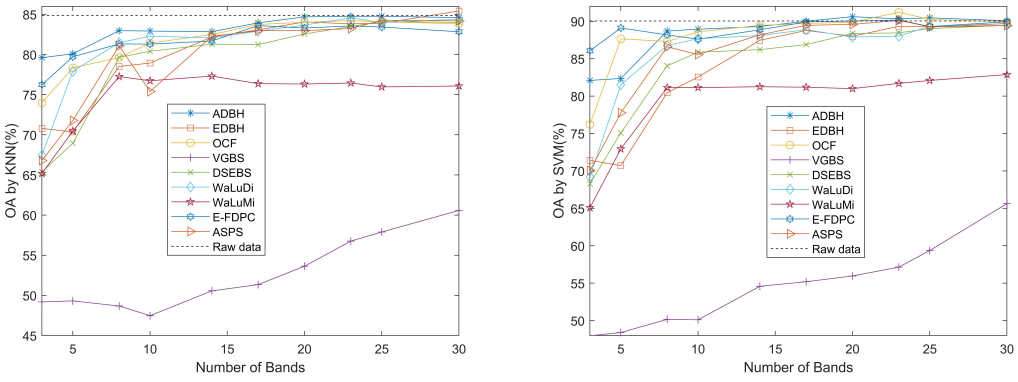
(a) (b)

Figure 4.5: OA curves on the PaviaU dataset with different UBS methods by using KNN (a) and SVM (b).



(a) (b)

Figure 4.6: OA curves on the Salinas dataset with different UBS methods by using KNN (a) and SVM (b).



(a) (b)

Figure 4.7: OA curves on the KSC dataset with different UBS methods by using KNN (a) and SVM (b).

Table 4.2: Classification results from different approaches for the PaviaU dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	ASPS	EDBH	ADBH	Raw data
OA by KNN(%)	85.64±0.01	82.64± 0.01	84.58±0.01	**86.43±0.01**	85.77±0.01	84.73±0.01	86.41±0.01	81.38±0.00	85.57±0.01	85.73±0.01
AA by KNN(%)	82.55±0.00	77.08±0.00	81.32±0.00	82.31±0.00	83.11±0.00	81.24±0.00	**83.47±0.01**	75.61±0.00	81.76±0.00	82.02±0.01
Kappa by KNN	0.80±0.00	0.76±0.00	0.79±0.01	**0.81±0.00**	0.42±0.00	0.79±0.00	0.82±0.00	0.75±0.01	0.80±0.00	0.81±0.01
OA by SVM(%)	90.88±0.00	90.75±0.00	89.39±0.00	91.08±0.00	91.40±0.00	90.07±0.00	83.87±0.01	83.77±0.00	**91.63±0.00**	91.64±0.01
AA by SVM(%)	88.74±0.00	88.25±0.00	87.23±0.00	88.96±0.00	88.82±0.00	85.04±0.0	72.30±0.00	73.31±0.000	**89.30±0.00**	88.12±0.01
Kappa by SVM	0.88±0.00	0.88±0.00	0.86±0.00	0.88±0.00	0.88±0.00	0.84±0.00	0.78±0.01	0.78±0.00	**0.89±0.00**	0.89±0.00

Table 4.3: Classification results from different approaches for the Salinas dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	ASPS	EDBH	ADBH	Raw data
OA by KNN(%)	88.82±0.00	82.74± 0.00	**88.99±0.00**	85.98±0.01	87.85±0.00	87.58±0.00	86.99±0.00	87.03±0.00	88.16±0.00	87.70±0.01
AA by KNN(%)	93.58±0.00	85.6±0.00	**93.73±0.00**	92.53±0.00	92.23±0.00	92.52±0.01	92.06±0.00	91.94±0.00	93.21±0.00	93.27±0.01
Kappa by KNN	**0.88±0.00**	0.81±0.01	**0.88±0.01**	0.86±0.01	0.84±0.00	0.86±0.01	0.86±0.01	0.86±0.00	0.87±0.01	0.86±0.01
OA by SVM(%)	92.28±0.00	92.04±0.00	92.28±0.00	91.03±0.00	92.31±0.00	92.45±0.00	91.90±0.00	92.40±0.01	**92.72±0.00**	92.87±0.00
AA by SVM(%)	95.67±0.00	95.28±0.00	95.93±0.00	95.68±0.00	95.18±0.00	95.91±0.00	95.63±0.00	95.79±0.00	**96.04±0.00**	96.42±0.00
Kappa by SVM	0.91±0.00	0.91±0.00	0.91±0.00	0.91±0.00	0.90±0.00	0.91±0.00	0.91±0.00	**0.92±0.00**	**0.92±0.00**	0.92±0.01

In principle, the HSI classification results can be quantitatively evaluated by three common metrics from the confusion matrix, including the overall accuracy (OA), the average accuracy (AA) and the Kappa coefficient. The OA is the percentage of the corrected classified pixels in total, and the AA reflects the mean classification accuracy over all the classes. The Kappa coefficient is estimated for evaluating the reliability of the classification result. In this section, the compared results will be illustrated in two forms. Firstly, for all four HSI datasets, the OA curves are generated according to OAs against different chosen numbers of bands varying from 3 to 30. Also, the OA, AA and Kappa coefficient of different algorithms have been compared with certain determined numbers of bands. For

Table 4.4: Classification results from different approaches for the KSC dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	ASPS	EDBH	ADBH	Raw data
OA by KNN(%)	82.49±0.01	50.56±0.02	81.28±0.01	82.07±0.01	77.31±0.01	81.68±0.01	82.26±0.01	82.18±0.01	**82.84±0.01**	85.86±0.01
AA by KNN(%)	74.59±0.01	37.08±0.03	71.30±0.01	67.32±0.01	73.08±0.01	73.45±0.01	74.33±0.01	**74.88±0.01**	73.85±0.01	79.15±0.01
Kappa by KNN	0.80±0.01	0.44±0.03	0.79±0.01	0.75±0.01	0.80±0.01	0.80±0.01	0.80±0.01	0.80±0.01	**0.81±0.01**	0.84±0.00
OA by SVM(%)	**89.45±0.01**	54.58±0.03	86.21±0.01	88.09±0.01	81.25±0.01	88.86±0.01	88.19±0.01	87.45±0.00	89.3±0.01	90.04±0.00
AA by SVM(%)	80.47±0.01	44.25±0.04	79.10±0.01	74.67±0.01	80.00±0.01	79.57±0.01	**82.37±0.01**	81.88±0.01	80.87±0.01	85.58±0.00
Kappa by SVM	**0.88±0.01**	0.48±0.01	0.85±0.01	0.79±0.01	0.87±0.01	0.87±0.01	0.86±0.01	0.86±0.00	**0.88±0.01**	0.89±0.01

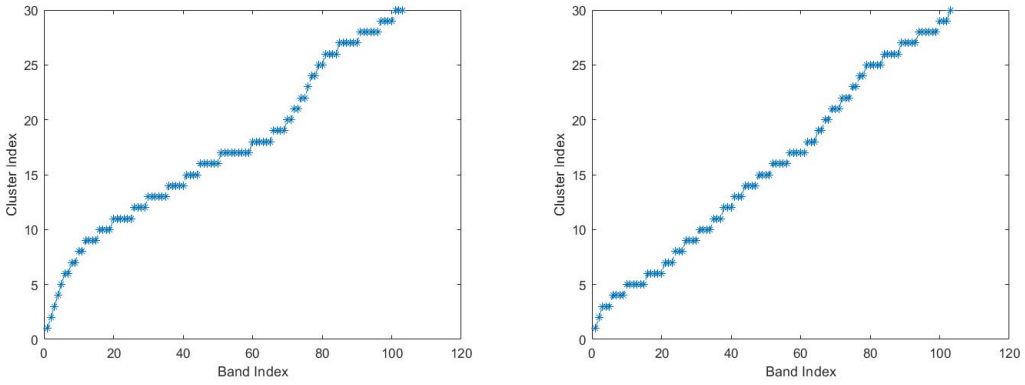
the OA curves in most datasets, the performance of most approaches keep stable after the number of chosen band is around 10 to 15. Even when more bands are chosen, there is no significant improvement for most of them. Therefore, detailed comparison with 14 selected bands on the four datasets, in terms of OA, AA and Kappa, is given in Tables 4.1-4.4. The best performance except the result with raw data are labelled bold.

Figure 4.4 and Table 4.1 show the classification results for the Indian pines dataset. As seen in Figure 4.4, the ADBH has the highest OA on the SVM classifier with 3 to 30 selected bands, which has also produced about the highest OA on the KNN classifier. Although the OA of ADBH on KNN is the second best when the number of the chosen bands is no more than 20, it outperforms DSEBS after more bands are chosen. Despite of the best OA generated on the KNN classifier, DSEBS has quite poor performance on SVM, which shows a certain degree of lack of robustness or stability. The ASPS has poor performance on both KNN and SVM classifiers. Table 4.1 actually shows as an example the classification results of all relevant methods with 14 selected bands. As can be seen, the proposed method has produced the best results in terms of OA, AA and Kappa on the SVM classifier, and the second best on the KNN classifier just after DSEBS. In addition, the ADBH framework can outperform the raw data without band selection when there are more than 10 selected bands on the KNN classifier or more than 30 on SVM, which further validates the superiority of the proposed approach.

Figure 4.5 and Table 4.2 summarize the classification results of all methods on the PaviaU dataset. In Figure 4.5 (a), with the KNN classifier, ASPS, WaLuDi and WaLuMi produce the best results with the number selected bands increasing

from 5 to 30, and the proposed ADBH has not achieved the best result. However, the classification results from KNN only achieves about 87%, which is far less than those from SVM at nearly 92%. For the SVM classifier, the results from ADBH is among the best when 15 or more bands are selected, and other best ones include WaLuMi, WaLuDi and OCF. Surprisingly, ASPS and EDBH produce the worse results on SVM. Although WaLuMi, WaLuDi and OCF seem to produce the best results in this group of experiments, as shown in Figure 4.4 and Table 4.1, they appear to among the worst with the Indian pines dataset on KNN and/or SVM classifiers under a certain range of the selected bands. From Table 4.2, it can be found that WaLuDi, ASPS and WaLuMi have produced the best results with the KNN classifier with 14 selected bands. However, the results from ADBH is the best on the SVM classifier and outperform all these three approaches.

For the Salinas dataset, the related comparison is shown in Figure 4.6 and Table 4.3. In Figure 4.6, ADBH algorithm achieves the most stable result on both classifiers, which is more robust than other methods. For the comparison between the ADBH method and the full dataset, it can be seen that the ADBH has an obvious advantage with the KNN classifier after 5 more bands are chosen. After more than 15 bands are chosen, the ADBH also achieves a better result than the raw dataset. The VGBS method does not perform well with the KNN classifier and the WaLuDi method has not achieved a good performance with the SVM classifier. Although the OCF method has the best performance with the KNN classifier when the number of the chosen bands are around 10, its performance is not as robust as the ADBH from any number of chosen bands. According to Table 4.3, the DSEBS is slightly better than ADBH on KNN while ADBH has the best performance on the SVM.



(a) (b)

Figure 4.8: The Clustering results (defined cluster number equals to 30).(a) ADBH, (b) OCF.

For the KSC dataset, ADBH has the best performance against all others with the KNN classifier. Although the E-FDPC method performs best with the SVM classifier when the number of the selected bands is quite small, ADBH has better result after more bands are chosen. In general, the ADBH method has the best performance on both classifiers from Figure 4.7, whilst the VGBS method performs quite poor. In Table 4.4, the ADBH algorithm has a satisfactory result when 14 bands are chosen with the KNN classifier. Although the OA of the ADBH method is not the best with the SVM, it achieves the second best with a small gap behind the first.

4.3.3 Extended discussions

In this subsection, extended analysis is carried out to compare the performance of different UBS methods over the four tested HSI datasets. Afterwards, the performance on the PaviaU dataset will be highlighted since the ADBH does not have the top robust performance on it. In addition, the computational time of

each method will be compared to evaluate the efficiency of these UBS methods.

Table 4.5: Number of parameters and computational time (s) of different UBS methods with 30 selected bands.

Methods	No.Parm.	Indian pines	Pavia U	Salinas	KSC
ADBH	0	0.35	2.69	2.23	5.91
OCF	0	0.7	0.65	1.13	1.66
VGBS	1	0.54	**0.24**	**0.82**	**0.81**
DSEBS	4	**0.2**	1.02	1.05	3.03
WaLuDi	4	41.95	99.7	198.51	408.68
WaLuMi	4	14.04	13.82	29.68	51.23
E-FDPC	1	0.97	6.85	3.11	27.97

From results of all the compared methods, it can be discovered that some methods have unstable performance on the different datasets and different classifiers. For example, the WaLuMi method achieves better performance on the PaviaU dataset but ranks the last on the Indian pines dataset. Moreover, the VGBS has the worst performance on the KSC dataset, especially the lower OA than those from other datasets. The ASPS has robust performance with the KNN classifier in the PaviaU dataset, but its performance with the SVM is quite poor. From our point of view, this phenomenon may be explained by three reasons. Firstly, the four datasets are from two different HSI sensors, AVIRIS and ROSIS. The AVIRIS sensor seems to be noisy and usually heavily polluted, as seen in the KSC dataset. Therefore, poor results from some methods may indicate their lack of robustness in dealing with noisy datasets. Secondly, most of the compared methods have parameters which are usually set empirically. These fixed parameters may limit the stability of the associated algorithms when different datasets or different classifiers are applied. The inconsistency in performance is prevalent for most unsupervised methods when relying on different parameters. In addition, the UBS is an optimization task as discussed before. If the algorithm focuses too much on local optimal solution, the instability will occur. However, thanks to

the combination of AD and BH, the ADBH method can provide a parameter-free way for solving the inconsistency problem and avoiding the effect of noisy bands. With the decent results of the ADBH method on all four datasets, especially the noisy KSC dataset, the robustness and stability of the proposed ADBH framework have been fully validated. We have noticed that the performance has been hugely improved with the utilization of AD, the poor performance of EDBH illustrates that the euclidean distance is not robust to the noisy band in the designed hierarchy. Besides, the proposed parameter-free framework can prevent from setting empirical parameters, where the number of parameters in each method is compared in Table 4.5.

Although the ADBH method performs consistently well on four datasets, the proposed ADBH have not achieved the best result on the PaviaU dataset, especially with the KNN classifier when more bands are chosen. According to Figure 4.8, it can be noticed that the OCF approach produces similar results to the ADBH, especially after the desired number of band is above 15. Since both ADBH and OCF cluster bands into several groups and select the most significant one from each group, it can be concluded that this kind of strategy is sensitive to noisy bands when the desired number of bands is large. In Figure 4.8, it can be seen that these two methods have clusters with only one band inside, where noisy bands have potential to be chosen. In Figure 4.5, the OA curves of most approaches start to fall when 15 or more bands are chosen, which also infers that the proper number of the selected bands might be around 15. More bands in the chosen subset may have few or even negative effect on the classification accuracy.

The computational complexity is a crucial issue for the efficiency of UBS algorithms. Hence, the computational complexity is compared using the computa-

tional time of every method on various datasets on the same software/hardware platform. Table 4.5 depicts the processing time of different methods when 30 bands are chosen on the four tested datasets. As seen in Table 4.5, the ADBH framework has a fairly good computational time among all compared methods. Although the VGBS is the most efficient one, its performance on the classification is not good. For the WaLuMi and WaLuDi algorithms, their computational burdens are the heaviest, which reflects the drawback of the complicated distance measurement they used. Most of the existing band selection approaches fail to maintain the consistency of the selected bands when the number of the desired bands varies. As such, the aim of ADBH is to provide a band hierarchy to tackle this challenging problem. With the derived band hierarchy, any desired number of bands can be selected without re-running the whole process as most other approaches do, including OCF. Considering the fact that there is no prior information of the optimal number of bands for a given HSI dataset, in practice the process of band selection needs be repeated for quite a few times. As a result, the overall computational costs will be linearly accumulated for most other approaches. However, thanks to the ADBH, the overall computational cost of the designed approach remains almost unchanged as the additional costs in selecting different numbers of bands is minor and can be neglected. To this end, the computational cost of the proposed approach is in fact far more efficient than conventional approaches including OCF. In summary, the proposed ADBH method seems to be a robust, effective and efficient solution for UBS of HSI.

4.4 Summary

In this chapter, an adaptive distance based band hierarchy (ADBH) clustering framework is proposed for UBS in HSI, which can effectively present the hierarchy structure of HSI and restrict the effect of the noisy bands in the band selection process. Any flexible numbers of the band subset can be obtained with the tree-based hierarchy. To reduce the effct of noisy bands, an adaptive distance is proposed by jointly considering the Euclidean distance and cluster intensity. Experiments on four commonly used datasets acquired from two HSI systems have proved the effectiveness of the proposed ADBH framework.

Chapter 5

Concrete Autoencoder for Unsupervised HSI Band Selection

5.1 Introduction

Nowadays, deep-learning based methods have received increasingly wide attention in the computer vision community and beyond [101]. In comparison to the conventional methods, deep-learning based approaches can automatically generate favourable features in the absence of human intervention and subjective parameter settings. Actually, many deep-learning models have already been applied in HSI, such as convolutional neural network (CNN) [76, 102, 103] and autoencoder (AE) [74,75], which are mainly for feature extraction and data classification [103]. Unlike data classification in HSI, the band selection task has no available ground truth to evaluate the chosen band subset in training the deep-learning networks.

To tackle this particular difficulty, some deep-learning based band selection methods combine the band selection network with a pretrained CNN [102]. However, classification based class label information are often adopted in these methods to tune their models, which is not the spotlight of this paper. Another trend is deep-learning based band selection, which is often implemented using the AE in an unsupervised manner. The simplest AE can be composed of an encoder layer and a decoder layer. By applying a reconstruction loss between the input and output layers, the AE can encode the structure of input data and yield a desired representation. The current AE-based methods are mostly ranking-based, where the weight of each node on the encoder layer is utilized to represent the significance of each band. However, there are several drawbacks for this kind of methods. The generated representation from the encoder is more like a combination of the raw data, where the weight values of nodes in the encoder layer can be both positive and negative. Some bands are chosen only because they have large absolute weights, which does not fully represent their significance. Besides, the aforementioned methods heavily rely on the ranking value or the weight to choose the desired band, which inevitably suffer from the disadvantages of the ranking-based UBS methods, i.e. the high correlation between chosen bands.

In this chapter, an improved deep learning-based framework based on the previous work has been proposed. By training an AE with the reconstruction loss, the optimal band subset is found out for reconstructing the original HSI cube. Different from the previous work, a band subset can be obtained directly instead of ranking the significance of each band. The rest of this chapter is organized as follows: Section 5.2 describes the proposed framework in detail. The experimental results and discussions on four HSI datasets are presented in

Section 5.3. In Section 5.4, a brief summary of this chapter is given.

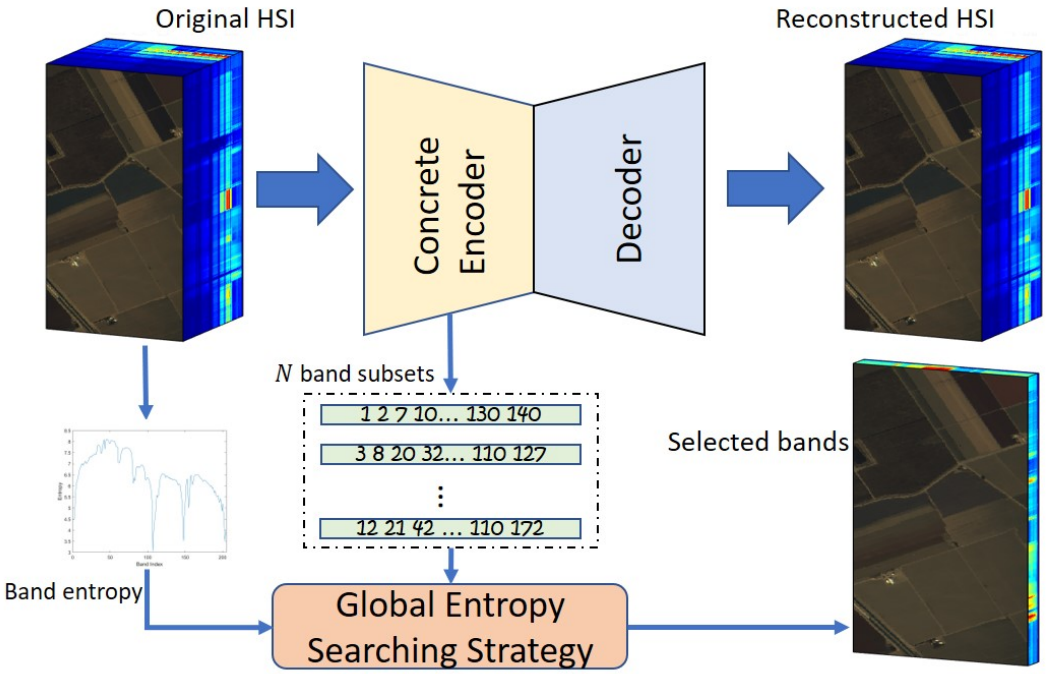


Figure 5.1: The flowchart of the proposed CAE-UBS framework, where the L is the reconstruction loss.

5.2 Proposed Method

In this section, the proposed Concrete Autoencoder framework for unsupervised band selection (CAE-UBS) will be presented in detail, including the concept of CAE based band selection, determining the optimal band subset, and computational complexity analysis. The flowchart of the proposed CAE-UBS framework is depicted in Figure 5.1 and relevant details are discussed as follows.

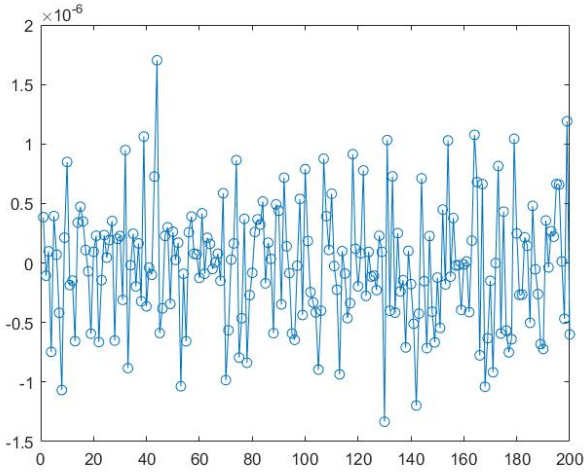


Figure 5.2: Weight values of one column in the learned weight matrix W^1, the horizontal and vertical axes represent the band index and weight values, respectively.

5.2.1 CAE based band selection

In general, a standard AE includes one encoder module and one decoder module. The encoder represents the mapping between input data and the hidden representation while the decoder is to reconstruct the input data from the hidden representation. Let a matrix $X = [X_1, \ldots, X_i, \ldots, X_m] \in R^{m \times D}$ denote the projected data from a hypercube, where m represents the total number of pixels in the HSI image and D is the number of spectral bands. The encoder function can be depicted as $H_i = \sigma_1(X_i W^{en} + b^{en})$ and the decoder function that reconstructs the input data $\hat{X}_i = \sigma_2(H_i W^{de} + b^{de})$, where the H_i is the hidden representation of the input data and the $\hat{X}_i$ is the reconstructed data. σ_{en} and σ_{de} are the activation functions, and W and b are the weighted matrices and bias vector respectively. For the proposed UBS method, each band will be indicated by the columns of the input weight matrix $W^{en} = (w_1, ..., w_D)$. The AE can be trained

by minimizing the reconstruction loss in a supervised manner:

$$L = \frac{1}{2m}||X - \hat{X}||_F \tag{5.1}$$

In the previous work [75] and other similar work [76], the desired band subset can be chosen by ranking the learned weight W^{en} from the encoder part. The basic assumption here is that a highly ranked weight indicates more importance of the corresponding band. However, the weight learned from AE in general cannot represent the significance of the band. For example, Figure 5.2 shows the value of the learned input weight with one column in the learned weight matrix W^{en}. Although positive values reflect the degrees of contribution from the bands, there are also several negative values. Besides, the motivation of AE based band selection is to select the most significant bands for spectrum reconstruction, yet the input weight based band selection strategy seems not linked to this objective. Therefore, it is inappropriate to select the bands according to the associated input weights.

As the purpose of the AE based band selection is to learn an important hidden representation from the input data for the reconstruction of HSI, it would be more reasonable to learn the desired band subset from the encoder part as the key latent features of the raw data. Inspired by this, a sparse input weight matrix is desired, whose values can be only 1 and 0, indicating the corresponding band is selected or not. In this manner, the weight of the bands that do not contribute to the reconstruction will be 0, otherwise it will be 1. Moreover, the extracted band subset will be optimal as the weights of the chosen bands are jointly learned. However, this sparse weight matrix cannot be updated during

the backpropagation in a standard AE as each column of this matrix is a one-hot vector, i.e. a non-differentiable discrete variable. To tackle this problem, a novel concrete AE for the UBS has been introduced, where the sparse matrix can be estimated with the aid of concrete distribution as detailed below [104, 105].

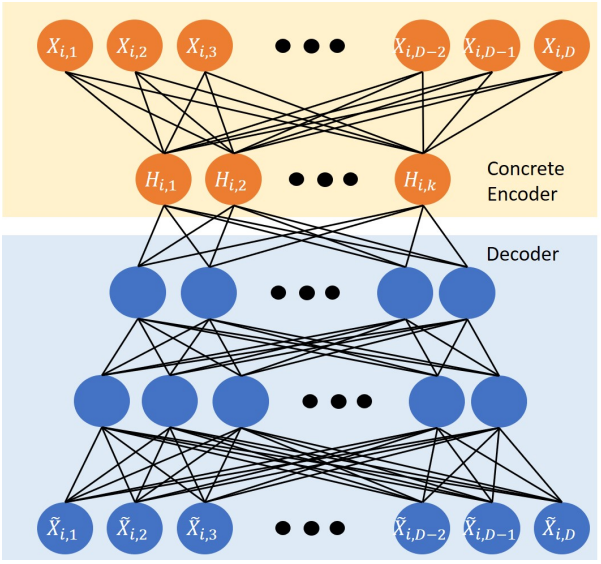


Figure 5.3: The diagram of the designed Concrete autoencoder

The Concrete distribution is defined to produce a continuous distribution over a discrete variable, for example, a one-hot vector. For a categorical variable z with different class probabilities α_k, a one-hot vector can be generated based on the Gumbel-Softmax trick with a Gumbel distribution g_k [104, 105]:

$$z = one\_hot\left\{\arg\max_k [g_k + log(\alpha_k)]\right\} \tag{5.2}$$

As the above operation is non-differentiable, the Concrete distribution is applied to calculate the continuous relaxation of the one-hot vector, where the k^{th} element

of the obtained sample S from the Concrete distribution is defined as:

$$S_k = \frac{exp((g_k + log(\alpha_k))/T)}{\sum_{d=1}^{D} exp((g_k + log(\alpha_k))/T)} \tag{5.3}$$

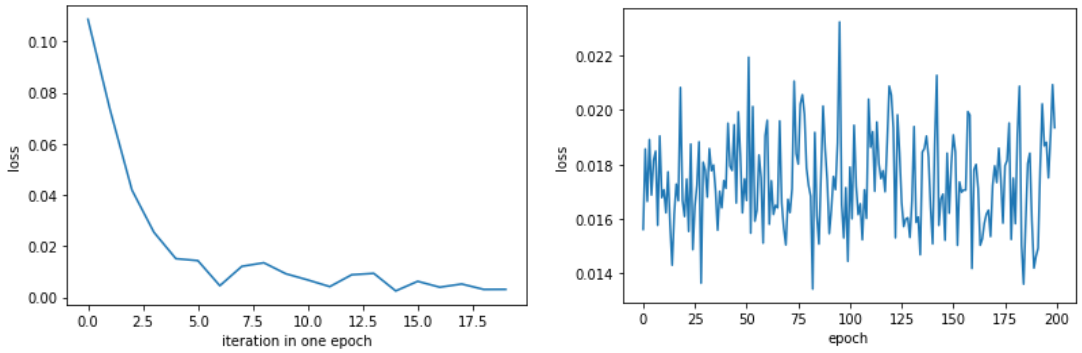
The temperature parameter T controls the relaxation of the one-hot vector, where the S_k will be nearly equal to 1 when T approaches to 0. With the reparameterization trick, S_k is differentiable when estimating the gradient in the backpropagation.

In the proposed CAE-UBS framework, the above Concrete random variables have been employed to select the input bands. Let the desired number of bands in the band subset be k, a new weight matrix S will be built with a size of $D \times k$. For each column of the weight matrix S, a D-dimensional Concrete random variable S_k is sampled following the (5.3). In [104, 105], the α_k is randomly initialized with small positive values for exploring different linear combinations of the input values. However, it is found out that the proposed CAE-UBS framework is not easy to converge with random initialized values. In the designed framework, the α_k is initialized with the weight matrix from a predefined fully connected layer to regularize the learning process, where the utilized weight matrix has the same size of S.

Based on the above strategy, the training process can be faster. In this way, the output of the encoder module is $H_i = X_i S$. As S_k is a one-hot vector, the composed weight matrix S is a desired sparse matrix, in which the selected k bands can be directly chosen. With the aid of the introduced Concrete random variable and trick of reparameterization, the forward propagation can generate a candidate band subset, and the backpropagation will refine the results of selected

band subset iteratively for choosing the best one for optimal reconstruction of HSI. For the decoder module of the proposed CAE-UBS, similar to the traditional stacked AE [74, 75] three stacked fully connected layers are used for effective reconstruction of the original HSI data with the optimal band subset selected from the encoder module. Furthermore, the mean squared error (MSE) is used to measure the reconstruction loss for its simplicity. The diagram of the designed CAE is illustrated in Figure 5.3.

5.2.2 Optimal band subset searching



(a) (b)

Figure 5.4: (a) The training loss from the 100th training epoch, where the number of iterations equals to the number of batches, (b) Training loss of 200 training epochs on Indian Pine dataset.

For searching the desired band subset in an efficient way, all samples from an HSI dataset are divided into different batches in a similar way as other deep learning models [101]. In this way, multiple band subsets can be obtained during each epoch. Let N be the number of band subsets produced in one epoch, it actually equals to the number of iterations, i.e. the number of batches, in each epoch. Although a band subset is selected according to its minimized reconstruction er-

ror, it can be potentially the local optimal solution due to the random selection of the batch, where searching for a global optimal band subset is still needed. To this end, a simple yet robust information entropy based searching strategy [58] is introduced in the proposed CAE-UBS framework as detailed below.

Generally, there are several motivations for considering the global searching strategy. The first is to find an efficient way to obtain the optimal band subset without suffering a huge computational burden, especially for those without deep learning. The other reason is to follow the assumption that the optimal band subset should be the one with the best reconstruction ability. We have further speculated that the desired band groups contain more information, which is beneficial for spectrum reconstruction. Under the consideration of these assumptions, a global searching strategy can be defined based on the information theory [58]. The IE, i.e. the Shannon entropy, is usually utilized to measure the information contained within a random variable. For a given band X_i, the definition of its IE is depicted as:

$$E(X_i) = -\int_{X_i} P(X_i) log(P(X_i)) dx \tag{5.4}$$

where the $P(X_i)$ denotes the probability density function of X_i. After calculating the IE for each band, the band subset with the largest IE is chosen as the desired band subset from N candidates, and the result is considered as the global optimal solution. As this search strategy is quite straightforward and efficient, it has been adopted in the proposed CAE-UBS approach.

For efficiently searching the desired band subset, another key point is the generation of the potential candidates. As one training epoch can produce N candidates, this will end up with a large search space after more training epochs.

Besides, more training epochs also increase the computing time of the whole algorithm. To identify the optimal band subset without too much computational cost, the number of training epochs needs to be reduced. In deep-learning based methods, the importance of more training epochs is to update the weight parameters of the proposed neural network and reach the convergence through multiple backpropagations. With the proposed CAE, the convergence is found to be faster due to the data volume. An example is shown in Figure 5.4, where the training loss, i.e. the reconstruction loss, of 200 training epochs on the Indian Pine dataset is presented. As seen, there are small differences between the training loss of each epoch from Figure 5.4 (a). Besides, the training loss is obviously reduced in each epoch based on Figure 5.4(b). Based on that, it is assumed that the proposed network can converge within only one epoch, thus the optimal band subset can be chosen from the generated N candidates. As a result, the efficiency of the proposed CAE-UBS method can be guaranteed.

5.2.3 Merits of CAE-UBS

With the Concrete random variable-based AE and information-entropy based optimal band subset searching strategy, the CAE-UBS framework can determine an optimal band for the effective reconstruction of the original spectral data. Different from other AE-based band selection frameworks, the band selection task has been formulated as a searching-based process by maximizing the accumulated information entropy of the desired band group instead of ranking the significance of each band. Moreover, the proposed CAE can solve the problem of backpropagation with a discrete variable, which makes the designed network able to be trained with the reconstruction loss. Being trained in a self-learning way without intro-

Algorithm 5 CAE-UBS

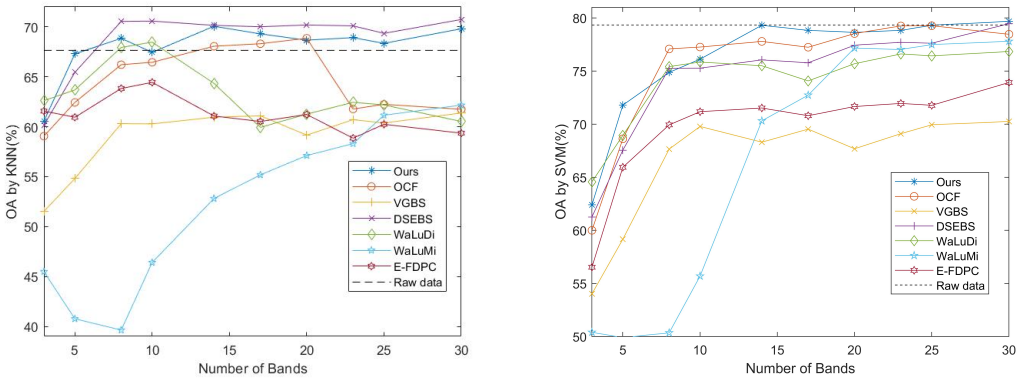
Input: Raw HSI data $X = [X_1, \ldots, X_i, \ldots, X_m] \in R^{m \times D}$, desired number of bands K.
Initialize: Hyperparameters Initialization :Adam optimizer with learning rate lr, Temperature parameter T, Batch size B.
BEGIN
Estimate E of each band in X
while the first epoch **do**
    Encoder module: learn S based on (5.3);
    Save N band subsets
    Decoder module;
    Update reconstruction loss L based on (5.1);
    Backpropagation with optimizer;
end while
Global optimal band subset searching with E of each band and N band subsets;
Output: Band subset n.
END

ducing any class label information, the proposed CAE-UBS has the potential to inspire more related research on deep-learning based band selection in the future. At last, a global search strategy is designed to identify the optimal band subset with the best reconstruction ability, where only one epoch is found to be sufficient in IE based selection of the optimal band subset for efficiency. The whole process of the proposed CAE-UBS is summarized in Algorithm 5, where the performance of the CAE-UBS framework is further discussed in the next section.

5.3 Experimental Results

Due to the lacking of the ground truth in UBS tasks, the performance of the band selection is usually indirectly assessed by evaluating the classification accuracy

with the selected bands. In experiments, the proposed CAE-UBS is compared with several state-of-the-art methods based on the classification performance from four popularly used publicly available HSI remote sensing datasets. Relevant details are presented as follows.



(a) (b)

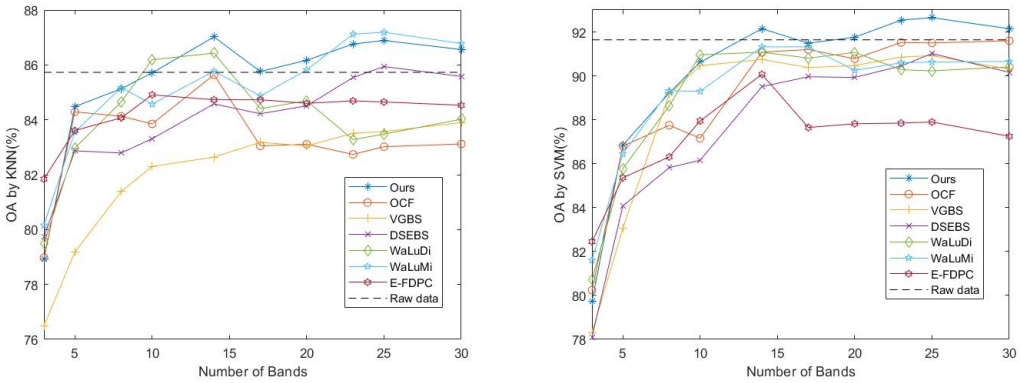
Figure 5.5: OA curves on the Indian Pine dataset with different UBS methods.(a) OA by KNN, (b) OA by SVM

Table 5.1: Classification results from different approaches for the Indian Pine dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	Ours	Raw data
OA by KNN(%)	68.07±0.01	60.96± 0.01	**70.16±0.01**	64.35±0.01	52.81±0.00	61.08±0.01	70.07±0.01	67.65±0.02
AA by KNN(%)	58.27±0.02	48.39±0.00	56.05±0.01	51.22±0.00	40.42±0.00	46.68±0.01	**59.36±0.02**	54.22±0.01
Kappa by KNN	0.63±0.01	0.55±0.01	**0.66±0.01**	0.59±0.01	0.45±0.00	0.55±0.01	**0.66±0.02**	0.62±0.01
OA by SVM(%)	77.79±0.01	68.30±0.01	75.78±0.01	74.99±0.01	70.65±0.01	71.52±0.01	**79.31±0.01**	79.33±0.01
AA by SVM(%)	76.82±0.01	64.53±0.02	74.96±0.01	75.58±0.01	67.35±0.01	70.57±0.02	**77.23±0.01**	71.47±0.01
Kappa by SVM	0.75±0.01	0.64±0.01	0.72±0.00	0.72±0.01	0.67±0.01	0.67±0.01	**0.76±0.01**	0.75±0.01

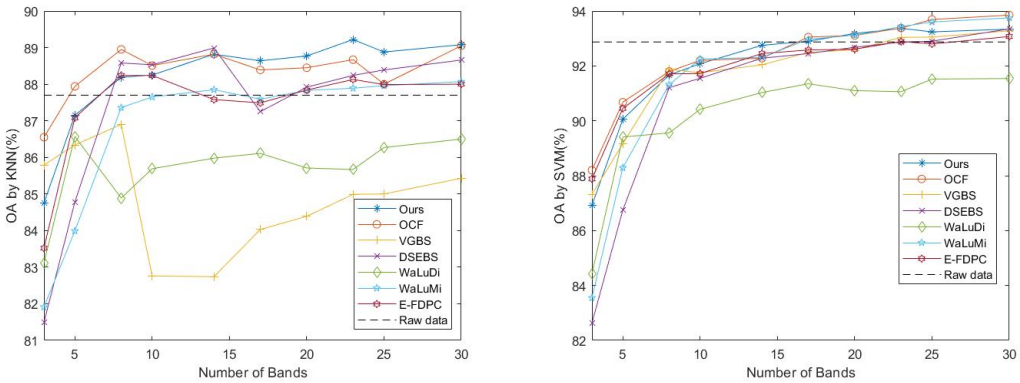
5.3.1 Settings

The result of the HSI classification is quantified by three common metrics generated from the confusion matrix, the overall accuracy (OA), the average accuracy (AA), and the Kappa coefficient. The OA represents the percentage of corrected



(a) (b)

Figure 5.6: OA curves on the PaviaU dataset with different UBS methods.(a) OA by KNN, (b) OA by SVM



(a) (b)

Figure 5.7: OA curves on the Salinas dataset with different UBS methods.(a) OA by KNN, (b) OA by SVM

Table 5.2: Classification results from different approaches for the PaviaU dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	Ours	Raw data
OA by KNN(%)	85.64±0.01	82.64± 0.01	84.58±0.01	86.43±0.01	85.77±0.01	84.73±0.01	**87.02±0.01**	85.73±0.02
AA by KNN(%)	82.55±0.00	77.08±0.00	81.32±0.00	82.31±0.00	83.11±0.00	81.24±0.00	**83.44±0.02**	82.02±0.01
Kappa by KNN	0.80±0.00	0.76±0.00	0.79±0.01	0.81±0.00	0.42±0.00	0.79±0.00	**0.82±0.01**	0.81±0.01
OA by SVM(%)	91.08±0.00	90.75±0.00	89.51±0.00	91.08±0.00	91.32±0.00	90.07±0.00	**92.15±0.01**	91.64±0.01
AA by SVM(%)	88.25±0.00	88.32±0.00	87.33±0.00	88.75±0.00	88.72±0.00	85.04±0.00	**89.76±0.00**	88.12±0.01
Kappa by SVM	0.88±0.00	0.88±0.00	0.86±0.00	0.88±0.00	0.88±0.00	0.84±0.00	**0.89±0.00**	0.89±0.00

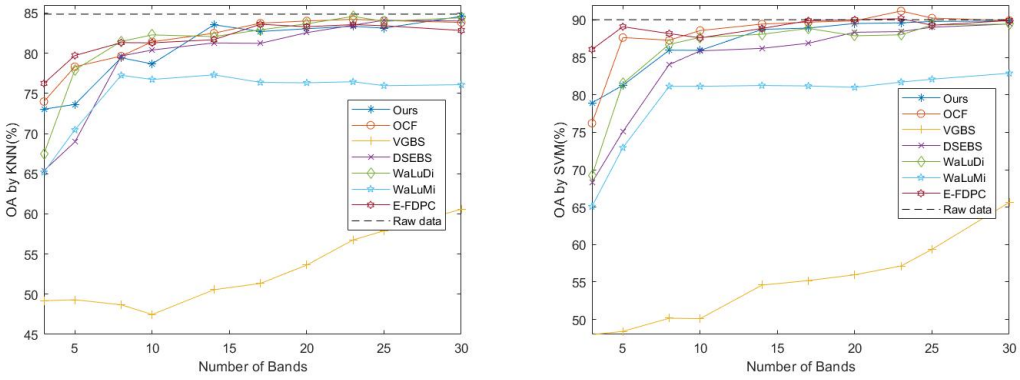
Table 5.3: Classification results from different approaches for the Salinas dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	Ours	Raw data
OA by KNN(%)	88.82±0.00	82.74± 0.00	**88.99±0.00**	85.98±0.01	87.85±0.00	87.58±0.00	88.82±0.00	87.70±0.01
AA by KNN(%)	93.58±0.00	85.6±0.00	**93.73±0.00**	92.53±0.00	92.23±0.00	92.52±0.01	93.62±0.00	93.27±0.01
Kappa by KNN	**0.88±0.00**	0.81±0.01	**0.88±0.01**	0.86±0.01	0.84±0.00	0.86±0.01	0.87±0.01	0.86±0.01
OA by SVM(%)	92.28±0.00	92.04±0.00	92.28±0.00	91.03±0.00	92.31±0.00	92.45±0.00	**92.75±0.01**	92.87±0.00
AA by SVM(%)	95.67±0.00	95.28±0.00	**95.93±0.00**	95.68±0.00	95.18±0.00	95.91±0.00	95.71±0.00	96.42±0.00
Kappa by SVM	0.91±0.00	0.91±0.00	0.91±0.00	0.91±0.00	0.90±0.00	0.91±0.00	**0.92±0.00**	0.92±0.01

classified pixels, and the AA is the mean classification accuracy over all classes. The Kappa coefficient is introduced to estimate the reliability of the obtained result. To verify the effectiveness of the proposed CAE-UBS framework in the HSI classification task, the CAE-UBS method is compared with some state-of-the-art algorithms, including OCF (TRC-OC-EFDPC) [53], DSEBS [57], VGBS [55], WaLuDi/WaLuMi [50], and the E-FDPC [52]. About the compared methods, the original codes are utilized from the authors and their proposed default parameters. Besides, the classification results employing the original data with all

Table 5.4: Classification results from different approaches for the KSC dataset.

Classifier	OCF	VGBS	DSEBS	WaLuDi	WaLuMi	E-FDPC	Ours	Raw data
OA by KNN(%)	82.49±0.01	50.56±0.02	81.28±0.01	82.07±0.01	77.31±0.01	81.68±0.01	**83.54±0.01**	85.86±0.01
AA by KNN(%)	**74.59±0.01**	37.08±0.03	71.30±0.01	67.32±0.01	73.08±0.01	73.45±0.01	74.58±0.01	79.15±0.01
Kappa by KNN	0.80±0.01	0.44±0.03	0.79±0.01	0.75±0.01	0.80±0.01	0.80±0.01	**0.81±0.01**	0.84±0.00
OA by SVM(%)	**89.45±0.01**	54.58±0.03	86.21±0.01	88.09±0.01	81.25±0.01	88.86±0.01	88.74±0.01	90.04±0.00
AA by SVM(%)	**80.47±0.01**	44.25±0.04	79.1±0.01	74.67±0.01	80.00±0.01	79.57±0.01	80.25±0.01	85.58±0.00
Kappa by SVM	**0.88±0.01**	0.48±0.01	0.85±0.01	0.79±0.01	0.87±0.01	0.87±0.01	**0.88±0.01**	0.89±0.01



(a) (b)

Figure 5.8: OA curves on the KSC dataset with different UBS methods.(a) OA by KNN, (b) OA by SVM

bands are also included (shown as 'Raw data' in this paper). For the proposed CAE-UBS method, it has several parameters similar to other deep learning networks. In the training process of CAE-UBS, the Adam optimizer [106] with a 1e-3 learning rate is employed. As stated before, the training epoch is set to be 1 for efficiency, and the batch size of the Indian Pines, PaviaU, Salinas, and KSC are set to be 512, 8192, 8192 and 8192 in the consideration of efficiency. Additionally, the activation function of the designed stacked decoder is ReLU [107]. For the temperature parameter, the CAE-UBS follows the annealing schedule in [105].

For the classification part, two commonly used classifiers, K-Nearest Neighbourhood (KNN) [99] and Support Vector Machine (SVM) [29], are employed to classify the above four datasets with the chosen band subsets from each method. In experiments, the parameters of KNN and the SVM are optimized through cross-validation. The 10% of the labelled samples are randomly chosen as the training samples for these two classifiers, while the rest of the samples are used for testing. For the compared methods, the experiments are repeated 10 times and the average metrics are reported. As the CAE-UBS method is a type of

deep learning method, where the final chosen band subset heavily relies on some stochastic issues like the construction of the batch, etc., the output band subset has a slight difference in each experiment. Nowadays, deep learning-based methods always report the best result with the trained model in other computer vision topics like image segmentation and object detection, etc. However, all the compared methods are not deep learning related, it is not fair if the CAE-UBS only claim the best result. Under this circumstance, the band selection result from five random experiments of the CAE-UBS framework have been chosen. These chosen five band subsets are utilized for the classification task and the experiment of each band subset is also repeated 10 times. Afterwards, the average metrics of these five subsets are claimed as the result of the proposed framework. For the hardware and software settings, the proposed CAE-UBS framework is implemented on the Pytorch 1.1.0 package without CUDA. The other band selection methods and the classification part are implemented on the MATLAB 2019a. All experiments are done with a 16GB Intel i5-8400 CPU. The details of the experimental results are shown in the below subsection.

5.3.2 Results discussion

In this subsection, the performance of the proposed CAE-UBS framework will be demonstrated in two forms. In the first, the OA curves of all the methods on four HSI datasets are generated against different number of chosen bands and shown in Figure 5.5-5.8. As it has been found that most of the methods compete with the performance using raw data when the number of chosen bands is around 30, the selected number of bands for the OA curves varies from 3 to 30. Besides, the OA, AA, and the kappa coefficient has been compared when 14

bands are chosen to better verify the effectiveness of the designed method. For most circumstances, the OA curves of most the methods become stable after 10 to 15 bands are selected. Therefore, detailed comparisons of each method on four datasets are given in Table 5.1-5.4. The best performance is labelled bold except the result with the raw data.

The classification results for the Indian Pine dataset are presented in Figure 5.5 and Table 5.1. In Figure 5.5, it can be seen that the CAE-UBS method has a robust performance on both classifiers. Although the performance is the second best on the KNN classifier, the difference between the CAE-UBS method and the DSEBS is quite small. After more than 20 bands are chosen, only the DSEBS and the CAE-UBS method keep reliable performance. For the SVM classifier, the performance of CAE-UBS is also quite stable, especially when the number of the chosen bands is beyond 20. Although the CAE-UBS method does not lead other methods in all cases, a robust OA curve proves the superiority of CAE-UBS method. Table 5.1 shows the classification results of all the methods with 14 selected bands. From the table 5.1, it can be seen that the proposed method, the OCF, and the DSEBS have better performance than the rest with the KNN classifier, but the performance of DSEBS with the SVM seems not as good as with the KNN classifier. For the proposed method,CAE-UBS have achieved the best with both classifiers, and it has an obvious advantage with the SVM classifier.

For the PaviaU dataset, the relevant comparisons are given in Figure 5.6 and Table 5.2. In Figure 5.6 (a), the proposed method has stable performance. With the KNN classifier, the CAE-UBS has achieved an increasing OA curve, and it outperforms other methods when the number of chosen bands is between 10 to 20. Although the WaLuMi method achieves the best performance with

the KNN classifier when more than 25 bands are selected, it does not perform well when a small number of bands is desired. Besides, the WaLuMi method has a good performance with the KNN classifier, but its performance with the SVM is not robust based on Figure 5.6 (b). With the SVM classifier, the CAE-UBS has achieved a more robust OA curve than the rest of the methods. With the consideration of both classifiers, the generated OA curves of CAE-UBS are steadier, which verifies the robustness of the CAE-UBS method. As can be seen from Table 5.2, the CAE-UBS has achieved the best OA with both classifiers, and advantages are very remarkable.

Figure 5.7 and Table 5.3 give the classification results for the Salinas dataset. In Figure 5.7, the CAE-UBS has achieved good performance with both classifiers. Although the CAE-UBS has not obtained the best results with the KNN classifier when less than 15 bands are chosen, it can be seen that the CAE-UBS method has an obvious advantage when more bands are chosen. Although both OCF and DSEBS have a good performance when less than 15 bands are selected, their OA curves with the KNN classifier are not stable compared to the CAE-UBS. For the VGBS and the WaLuDi methods, both have not obtained a satisfying result with the KNN classifier. With the SVM classifier, most of the methods have achieved robust performance except for the WaLuDi method. Compared to the OCF method, the CAE-UBS have not obtained the best OA curve, but the performance can be considered the second best. From Table 5.3, it can be noticed that the CAE-UBS method ranks the second and first with the KNN and SVM, respectively, where the differences are very small.

For the KSC dataset, the proposed CAE-UBS has not achieved the best performance on both classifiers from Figure 5.8. From Figure 5.8(a), the performance

of CAE-UBS method is in the middle among other compared algorithms when less than 10 bands are chosen. After that, the CAE-UBS has fairly good performance with two best results when the chosen number of bands are 14 and 30, and the difference between the CAE-UBS method and the rest robust methods like OCF, DSEBS, and E-FDPC are quite small. With the SVM classifier, the CAE-UBS has a very steady OA curve and the OCF method has achieved the best performance. The difference between the CAE-UBS method and the OCF is very similar to the circumstance with the KNN classifier. With both classifiers, the results of WaLuMi and the VGBS are quite poor. In Table 5.4, the CAE-UBS method has obtained the best performance on the KNN classifiers and OA of CAE-UBS ranks the third on the SVM classifier.

5.3.3 Extended discussion

Table 5.5: Computational time (s) of different UBS methods on four datasets with 30 selected bands.

Methods	Indian Pine	Pavia U	Salinas	KSC
Ours	0.75	1.9	1.5	1.4
OCF	0.7	0.65	1.13	1.66
VGBS	0.54	**0.24**	**0.82**	**0.81**
DSEBS	**0.2**	1.02	1.05	3.03
WaLuDi	41.95	99.7	198.51	408.68
WaLuMi	14.04	13.82	29.68	51.23
E-FDPC	0.97	6.85	3.11	27.97

To summarize the experimental results from the four datasets, some extended discussions are given below. In particular, three aspects will be discussed, i.e. the relevant poor results from the KSC dataset, the performance of the CAE-UBS method with more selected bands, and analysis of the computational time of each method.

Although the proposed method has obtained quite good results with the two popular classifiers on the four HSI datasets, the OA is not always the best which can be explained as follows. The network architecture and the strategy for searching the optimal band subset used in the proposed method are rather simple. The proposed CAE-UBS framework can be taken as a baseline, where its performance can be further improved by introducing a larger neural network or certain regularization terms such as spatial constraints. Actually, the quite satisfactory results on three of the four datasets, including the Indian Pine, PaviaU, and Salinas from two different sensors, the AVIRIS and the ROSIS have validated the robust performance and high generalized ability of the proposed network. To this end, it can be claimed that the proposed method can generate a global optimal solution in most cases.

As for the relatively less favourable results on the KSC dataset, this is mainly due to lack of sufficiently labelled pixels. Actually, in total there are only 4690 labelled pixels, which accounts for about 1.5% (4690/314368) of all samples in the KSC dataset. In the proposed CAE-UBS framework, the desired band subset is selected according to the reconstruction ability. To be more specific, each potential band group among N candidates is determined by random initialization of batches as stated before. Although the batch size is set to 8192 in the experiment, the labelled pixels do not have much effect in the reconstruction process. In other words, the selected bands might not be crucial for these labelled pixels, which explains the relatively low performance on this particular dataset.

As shown in the previous subsection, the proposed CAE-UBS framework can usually produce better results when more bands are selected. For example, the OA curve of CAE-UBS in Figure 5.8 (a) outperforms all others when more than

20 bands are chosen. As the CAE-UBS method is searching-based, a larger search space with more bands tends to produce better results. Therefore, it is prone to find the optimal band subset from the increased number of band combinations, which validates the searching ability of the developed deep-learning based UBS method.

As a significant issue for verifying the efficiency of UBS methods, the computational time of various methods have been compared. However, there is a dilemma that the CAE-UBS method depends on the Pytorch package and the rest are developed on MATLAB. In this way, the computational time is compared with the same hardware. From Table 5.5, it can discovered that the CAE-UBS method has a comparable computational time with other non-deep learning-based methods. Based on the computational time and the above performance, it can be concluded that the proposed CAE with only one training epoch can provide an efficient solution to the UBS. As declared before that the CAE-UBS method is still a baseline work, there are more potential improvements for deep learning-based methods on the UBS of the HSI.

5.4 Summary

In this chapter, a novel Concrete autoencoder-based framework has been proposed for the UBS in HSI. By introducing the Concrete autoencoder, desired band subsets can be obtained with the supervision of a self-reconstruction loss. After that, the information entropy is utilized to search a global optimal solution from the obtained band subsets. The proposed CAE-UBS has proved its superiority among several state-of-the-art UBS algorithms.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

The objective of this thesis is to design novel algorithms for the feature fusion and selection to improve the performance of HSI classification. These are presented in Chapters 3-5, including two feature fusion algorithms and two UBS methods. The developed techniques can be summarized as follows:

1) In Chapter 3, a superpixel-based feature specific SRC framework has been proposed to fully exploit the spectral-spatial features of the HSI. A superpixel map is generated to acquire better spatial information and save the computational cost. After superpixel generation, a SRC-based classifier is designed to assign each superpixel into one category. With the proposed SRC-based classifier, the developed approach has achieved a better performance than other methods in two HSI datasets, i.e. the Indian Pine dataset and PaviaU datasets. Furthermore, an online Mahalanobis-based metric learning strategy has been designed to exploit the correlation be-

tween different features, where the improved performance has validated the superiority of this designed distance measurement.

2) In Chapter 3, a superpixel-based multiple feature fusion SRC framework is also proposed to address the HSI classification with insufficient training samples. Three kinds of features are extracted from the raw data, including the spectral/spatial features from HSI, and elevation features derived from the corresponding LiDAR data. After that, a superpixel map is generated for each HSI by applying the SLIC algorithm, which can help to combine more contextual information and reduce the computation cost. To generate effective features for HSI classification in SRC, a localized MCCA has been utilized to fuse the extracted features and obtain a low-dimensional representation. After that, a kernelized SRC can assign the semantic label to each superpixel. By comparing other state-of-the-art algorithms on three publicly available datasets, i.e. the UH, Indian Pine, and PaviaU datasets, the efficacy and robustness of the proposed SMFF-SRC approach has been validated.

3) In Chapter 4, a band hierarchy clustering UBS framework for effective band selection has been proposed. A flexible tree hierarchy-based algorithm ADBH is developed to explore the data structure within HSI which can generate any desired number of band subsets. To overcome the effect of noisy bands, a novel adaptive distance (AD) metric has been introduced, which is combined with the ADBH framework. Moreover, the developed approach is parameter-free hence easy for implementation. The satisfactory results from experiments on four publicly available datasets have fully

demonstrated the robustness and efficiency of the proposed ADBH method.

4) In Chapter 5, a novel CAE-UBS framework has been proposed for deep learning-based selection of the optimal band subset in the HSI. To investigate the correlation of bands in the desired band group, a Concrete autoencoder is employed to search potential band subsets with the supervision of a self-reconstruction loss. Afterwards, the optimal band subset can be obtained through a global searching strategy based on the information entropy. The robust performance from experiments on various datasets has fully demonstrated the efficacy and efficiency of the proposed CAE-UBS framework.

6.2 Future Work

Followed by the contributions presented in the last section, the future research directions towards the feature fusion and selection can be highlighted as follows.

1) Nowadays, most of the designed feature fusion algorithms for improving the HSI classification are based on supervised learning. In deep learning era, it is common that many researchers have tried to combine the deep learning models, especially the CNN model, into the HSI classification [108]. However, the insufficient or even no training samples is quite often in practical applications. Therefore, the designed deep learning model is not easy to converge without a large set of training dataset. To this end, the unsupervised or self-supervised learning can be the potential direction in the future.

2) Although data classification is used to evaluate the efficacy of the selected bands, band selection can actually benefit many other applications of HSI, such as spectral unmixing, spectral reconstruction, object detection and data visualisation, etc [109–118]. However, applications in these fields are seldom selected, due mainly to the lack of available ground truth maps for quantitative evaluation. Further verification of the band selection approaches in these applications can be explored in the future. Besides, the CAE-UBS can be assumed to be a new trend for UBS in HSI as it does not require any label information to select the desired band subset. In the future, the CAE-UBS framework can be improved on two aspects. The first one is to make full use of the Concrete AE and speed up the convergence of the network by adding more regularization terms. The other one is to develop a multi-task network for selecting more discriminative bands for classification or other applications in HSI.

3) Nowadays, the object detection [119] has become one of the most popular topics in the remote sensing field. However, most of the object detection applications focus more on the RGB data. In the future, the fusion between the HSI and RGB data is essential. With the numerous spectral information, the discriminative ability between different objects can improve the performance of object detection.

References

[1] J. Tschannerl, J. Ren, F. Jack, J. Krause, H. Zhao, W. Huang and S. Marshall, "Potential of UV and SWIR hyperspectral imaging for determination of levels of phenolic flavour compounds in peated barley malt," *Food. Chem.*, vol. 270, pp. 105-112, Jan. 2019.

[2] J. Tschannerl, J. Ren, H. Zhao, F. Kao, S. Marshall and P. Yuen, "Hyperspectral image reconstruction using Multi-colour and Time-multiplexed LED illumination," *Opt. Laser. Eng.*, vol 121, pp. 352-357, Oct. 2019.

[3] X. Lu, Y. Yuan, and X. Zheng, "Joint dictionary learning for multispectral change detection," *IEEE Trans. Cybern.*, vol.47, no. 4, pp. 884-897, Apr. 2017.

[4] 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, Aug. 2015.

[5] J. Zabalza, C. Qing, P. Yuen, G. Sun, H. Zhao, J. Ren, "Fast implementation of two-dimensional singular spectrum analysis for effective data classification in hyperspectral imaging," *J. Franklin.*, vol. 4, pp. 1733-1751, 2018.

[6] C. Wang, J. Ren, H. Wang, Y. Zhang, J. Wen, "Spectral-spatial classification of hyperspectral data using spectral-domain local binary patterns," *Multimed Tools Appl.* vol. 22, pp. 29889-29903. Apr. 2018.

[7] C. Zhao, X. Li, J. Ren and S. Marshall, "Improved sparse representation using adaptive spatial support for effective target detection in hyperspectral imagery", *Int J Remote Sens.*, vol. 34 no. 24, pp. 8669-8684, Dec. 2013.

[8] D. Ma, Y. Yuan, Q. Wang, "Hyperspectral Anomaly Detection via Discriminative Feature Learning with Multiple-Dictionary Sparse Representation", *Remote Sens.* vol. 5, 745. 2018.

[9] G. Sun, A. Zhang, J. Ren, J. Ma, P. Wang, Y. Zhang, X. Jia, "Gravitation-based edge detection in hyperspectral images", *Remote Sens.* vol. 6, 592. 2017.

[10] R. Dian, S. Li, L. Fang, T. Lu, and J. M. Bioucas-Dias, "Nonlocal sparse tensor factorization for semiblind hyperspectral and multispectral image fusion," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2019.2951572.

[11] Y. Chen, W. He, N. Yokoya, and T-Z. Huang, "Hyperspectral image restoration using weighted group sparsity-regularized low-rank tensor decomposition," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2019.2915094.

[12] M. Chen, Q. Wang, X. Li, Discriminant Analysis with Graph Learning for Hyperspectral Image Classification.*Remote Sens.*, vol 6, 836. 2018.

[13] J. Zabalza, J. Ren, Z. Wang, S. Marshall and J. Wang, "Singular spectrum analysis for effective feature extraction in hyperspectral imaging," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 11, pp. 1886-1890, Nov. 2014.

[14] F. Cao, Z. Yang, J. Ren, W. Ling, H. Zhao, M. Sun, J. A. Benediktsson, "Sparse representation-based augmented multinomial logistic extreme learning machine with weighted composite features for spectral-spatial classification of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.* vol. 56, no. 11, pp. 6263-6279, Nov. 2018.

[15] F. Cao, Z. Yang, J. Ren, W. Ling, H. Zhao, M. Sun, "Extreme sparse multinomial logistic regression: a fast and robust framework for hyperspectral image classification," *Remote Sens.* vol. 9, pp. 1255, 2017.

[16] T. Qiao, J. Ren, Z. Wang, J. Zabalza, M. Sun, H. Zhao, S. Li, J. A. Benediktsson, Q. Dai, S. Marshall. "Effective denoising and classification of hyperspectral images using curvelet transform and singular spectrum analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 119-133, Jan. 2017.

[17] Z. Zhong, J. Li, D. A. Clausi, and A. Wong, "Generative adversarial network and conditional random fields for hyperspectral image classification," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2019.2915094.

[18] H. Li, G. Xiao, T. Xia, Y. Y. Tang, and L. Li, "Hyperspectral image classification using functional data analysis," *IEEE Trans. Cybern.*, vol.44, no. 9, pp. 1544-1555, Sept. 2014.

[19] Y. Zhou, and Y. Wei, "Learning hierarchical spectral–spatial features for hyperspectral image classification," *IEEE Trans. Cybern.*, vol.46, no. 7, pp. 1667-1678, Jul. 2016.

[20] X. Sun, Q. Qu, N. M. Nasrabadi and T. D. Tran, "Structured Priors for Sparse-Representation-Based Hyperspectral Image Classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 7, pp. 1235-1239, Jul. 2014.

[21] W. Liao, A. Pižurica, R. Bellens, S. Gautama and W. Philips, "Generalized graph-based fusion of hyperspectral and LiDAR data using morphological features," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 552-556, Mar. 2015.

[22] Y. Gu, T. Liu, X. Jia, J. A. Benediktsson and J. Chanussot, "onlinear Multiple Kernel Learning With Multiple-Structure-Element Extended Morphological Profiles for Hyperspectral Image Classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3235-3247, Jun. 2016.

[23] J. A. Benediktsson, J. A. Palmason and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol 43, no. 3, pp. 480-491, Mar. 2005.

[24] A. Plaza, P. Martinez, J. Plaza and R. Perez, "Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 466-479, Mar. 2005.

[25] G. Rellier, X. Descombes, F. Falzon and J. Zerubia, "Texture feature analysis using a gauss-Markov model in hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 7, pp. 1543-1551, Jul. 2004.

[26] Y. Zhang and S. Prasad, "Multisource geospatial data fusion via local joint sparse representation," *IEEE Trans. Geosci. Remote Sens.*, vol 54, no. 6, pp. 3265-3276, Jun. 2016.

[27] 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, Jul. 2017.

[28] L. Gan, J. Xia, P. Du and J. Chanussot, "Multiple feature kernel sparse representation classifier for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5343-5356, Sep. 2018.

[29] J. Li, P. R. Marpu, A. Plaza, J. M. Bioucas-Dias and J. A. Benediktsson, "Generalized Composite Kernel Framework for Hyperspectral Image Classification". *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4816-4829, Sep. 2013.

[30] J. Ham Yangchi. Chen, M. M. Crawford, J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492-501, Mar. 2005.

[31] Y. Yuan, J. Lin and Q. Wang, "Hyperspectral Image Classification via Multitask Joint Sparse Representation and Stepwise MRF Optimization," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2966-2977, Dec. 2016.

[32] W. Li, S. Prasad and J. E. Fowler, "Hyperspectral Image Classification Using Gaussian Mixture Models and Markov Random Fields," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 1, pp. 153-157, Jan. 2014.

[33] J. Li, J. M. Bioucas-Dias and A. Plaza, "Spectral–Spatial Classification of Hyperspectral Data Using Loopy Belief Propagation and Active Learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 844-856, Feb. 2013.

[34] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory,* vol. IT-14, no. 1, pp. 55-63, Jan. 2018.

[35] J. Zabalza, J. Ren, M. Yang, Y. Zhang, J. Wang, S. Marshall, and J. Han, "Novel folded-PCA for improved feature extraction and data reduction with hyperspectral imaging and SAR in remote sensing," *ISPRS J. Photogramm. Remote Sens.,* vol. 93, pp. 112-122, Jul. 2014.

[36] X. Kang, X. Xiang, S. Li, and J. A. Benediktsson, "PCA-based edge preserving features for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7140-7151, Dec. 2017.

[37] J. Wang and C.-I Chang, "Independent component analysis-based dimensionality reduction with applications in hyperspectral image analsysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1586-1600, Jun. 2006.

[38] H. Huang, G. Shi, H. He, Y. Duan, and F. Luo, "Dimensionality reduction of hyperspectral imagery based on spatial-spectral manifold learning," *IEEE Trans. Cybern.*, to be published. doi:10.1109/TCYB.2019.2905793.

[39] A. A. Green, M. Berman, P. Switzer and M. D. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Trans. Geosci. Remote Sens.*, vol. 26, no. 1, pp. 65-74, Jan. 1988.

[40] H. Yang, Q. Du, H. Su, and Y. Sheng, "An efficient method for supervised hyperspectral band selection," *IEEE Geosci. Remote Sens. Lett.,* vol. 8, no. 1, pp. 138-142, Jan. 2011.

[41] X. Cao, T. Xiong and L. Jiao,"Supervised band selection using local spatial information for hyperspectral image," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 3, pp. 329-333, Mar. 2016.

[42] S. Patra, P. Modi and L. Bruzzone, "Hyperspectral band selection based on rough set," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 10, pp. 5495-5503, Oct. 2015.

[43] Purdue's University Multispec Site: AVIRIS image Indian Pine Test Site. [Online] Available at: https://engineering.purdue.edu/ biehl/MultiSpec/hyperspectral.html.

[44] ROSIS image Pavia University. [Online]. Available: http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

[45] AVIRIS image Salinas Valley. [Online]. Available:http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

[46] AVIRIS image Kennedy Space Center. [Online]. Available:http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

[47] University of Houston dataset. [Online]. Available:http://hyperspectral.ee.uh.edu/?page_id=459

[48] C. I. Chang, Q. Du, T.-L. Sun, and M. L. G. Althouse, "A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 6, pp. 2631-2641, Nov. 1999.

[49] C. I. Chang, and S. Wang, "Constrained band selection for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 6, pp. 1575-1585, Jun. 2006.

[50] A. Martínez-Usómartinez-Uso, F. Pla, J. M. Sotoca, and P. García-Sevilla, "Clustering-based hyperspectral band selection using information measures," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4158-4171, Dec. 2007.

[51] Y. Yuan, J. Lin, and Q. Wang, "Dual-clustering-based hyperspectral band selection by contextual analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 3, pp. 1431-1445, Mar. 2016.

[52] S. Jia, G. Tang, J. Zhu, and Q. Li, "A novel ranking-based clustering approach for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 88-102, Jan. 2016.

[53] Q. Wang, F. Zhang, and X. Li, "Optimal clustering framework for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 10, pp. 5910-5922, Oct. 2018.

[54] Q. Wang, Q. Li, and X. Li, "Hyperspectral band selection via adaptive subspace partition strategy," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, to be published. doi:10.1109/JSTARS.2019.2941454.

[55] X. Geng, K. Sun, L. Ji, and Y. Zhao, "A fast volume-gradient-based band selection method for hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7111-7119, Nov. 2014.

[56] Y. Yuan, G. Zhu, and Q. Wang, "Hyperspectral band selection by multitask sparsity pursuit," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 631-644, Feb. 2015.

[57] G. Zhu, Y. Huang, J. Lei, Z. Bi, and F. Xu, "Unsupervised hyperspectral band selection by dominant set extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 227-239, Jan. 2016.

[58] J. Tschanneral, J. Ren, P. Yuen, G. Sun, H. Zhao, Z. Yang, Z. Wang and S. Marshall, "MIMR-DGSA: Unsupervised hyperspectral band selection based on information theory and a modified discrete gravitational search algorithm," *Inform Fusion.*, vol. 51, pp. 189-200, Jan. 2019.

[59] Y. Yuan, X. Zhang, and X. Lu, "Discovering diverse subset for unsupervised hyperspectral band selection," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 51-64, Jan. 2017.

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

[61] J. Yang, J. Wright, T. S. Huang and Y. Ma, "Image Super-Resolution Via Sparse Representation," *IEEE Trans. Image Process.* , vol. 19, no. 11, pp. 2861-2873, Nov. 2010.

[62] J. Mairal, M. Elad and G. Sapiro, "Sparse Representation for Color Image Restoration," *IEEE Trans. Image Process.* , vol. 17, no. 1, pp. 53-69, Jan. 2008.

[63] 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, Oct. 2011.

[64] 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, Jan. 2013.

[65] 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. and Remote Sens.*, vol. 7, no. 6, pp. 2056-2065, June 2014.

[66] 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, Dec. 2014.

[67] 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, Sept. 2014.

[68] T. Zhan, L. Sun, Y. Xu, G. Yang, Y. Zhang, Z. Wu, "Hyperspectral image classification via superpixel kernel learning-based low rank representation." *Remote Sens.* vol 10, 1639, 2018.

[69] W. Fu, S. Li, L. Fang, X. Kang, 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, Feb. 2016.

[70] L. Fang, S. Li, W. Duan J. Ren, J. A. Benediktsson, "Classification of hyperspectral images by exploiting spectral-spatial information of superpixel via multiple kernels." *IEEE Trans. Geosci. Remote Sens.* vol. 53, no. 12, pp. 6663-6674, Dec. 2015.

[71] L. Fang, S. Li, X. Kang, 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, Aug. 2015.

[72] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *J. Am. Stat. Assoc.*, vol. 58, no. 301, pp. 236-244, 1963.

[73] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492-1496, Jun. 2014.

[74] K. Han, Y. Wang, C. Zhang, C. Li and C. Xu, "Autoencoder Inspired Unsupervised Feature Selection," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, 2018, pp. 2941-2945.

[75] J. Tschannerl, J. Ren, J. Zabalza and S. Marshall, "Segmented Autoencoders for Unsupervised Embedded Hyperspectral Band Selection," *2018 7th European Workshop on Visual Information Processing (EUVIP)*, Tampere, 2018, pp. 1-6.

[76] Y. Cai, X. Liu and Z. Cai, "BS-Nets: An End-to-End Framework for Band Selection of Hyperspectral Image," *IEEE Trans. Geosci. Remote Sens.*, to be published. doi: 10.1109/TGRS.2019.2951433.

[77] L. Gao, R. Zhang, L. Qi, E. Chen and L. Guan. "The Labeled Multiple Canonical Correlation Analysis for Information Fusion." *IEEE Trans. on Multimedia*, vol. 21, no. 2, pp. 375-387, Feb. 2019.

[78] L. Sun, S. Ji and J. Ye, "Canonical correlation analysis for multilabel classification: A least-squares formulation, extensions, and analysis." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 194-200, Aug. 2010.

[79] W. Liu, X. Yang, D. Tao, J. Cheng and Y. Tang, "Multiview dimension reduction via Hessian multiset canonical correlations." *Inform Fusion.* vol. 41, pp. 119-128, May. 2018.

[80] B. Chandra and R. K. Sharma, "Exploring autoencoders for unsupervised feature selection," *2015 International Joint Conference on Neural Networks (IJCNN)*, Killarney, 2015, pp. 1-6.

[81] C. Hong, J. Yu, J. Wan, D. Tao and M. Wang, "Multimodal Deep Autoencoder for Human Pose Recovery," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5659-5670, Dec. 2015.

[82] X. Hou, L. Shen, K. Sun and G. Qiu, "Deep Feature Consistent Variational Autoencoder," *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Santa Rosa, CA, 2017, pp. 1133-1141.

[83] J. Yu, C. Hong, Y. Rui and D. Tao, "Multitask Autoencoder Model for Recovering Human Poses," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 6, pp. 5060-5068, June 2018.

[84] S. Prasad and L. M. Bruce, "Limitations of principle component analysis for hyperspectral target recognition," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 625-629, Oct. 2008.

[85] J. A. Benediktsson, J. A. Palmason and J. R. Sveinsson, "Classification of hyperspectral data from urban areas based on extended morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 480-491, March 2005.

[86] Z. Li, J. Chen, "Superpixel segmentation using linear spectral clustering," *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1356–1363, 2015.

[87] P. Jain, B. Kulis, I. S. Dhillon, K. Grauman, "Online metric learning and fast similarity search." *Proceedings of the Advances in Neural Information Processing Systems(NIPS)*, pp. 761–768, 2008

[88] X. Lan, S. Zhang, P. C. Yuen and R. Chellappa, "Learning Common and Feature-Specific Patterns: A Novel Multiple-Sparse-Representation-Based Tracker," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 2022-2037, April 2018.

[89] Y. Freund, R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.* vol 1, 119–137, 1997.

[90] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua and S. Süsstrunk, "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274-2282, Nov. 2012.

[91] P. N. Belhumeur, J. P. Hespanha and D. J. Kriegman. "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997

[92] F. Hong, H. Peng, and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, pp. 1226-1338, Aug. 2005.

[93] J. Feng, L. Jiao, F. Liu, T. Sun, and X. Zhang, "Mutual-information-based semi-supervised hyperspectral band selection with high discrimination, high information, and low redundancy," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no 5, pp. 2956-2969, May. 2015.

[94] C.-I. Chang, S. Wang, K.-H. Liu, M.-L. Chang, and C. Lin, "Progressive band dimensionality expansion and reduction via band prioritization for hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 4, no. 3, pp. 591-614, Sep. 2010.

[95] X. Wei, Q. Yang, Y. Gong, N. Ahuja and M. Yang, "Superpixel Hierarchy," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4838-4849, Oct. 2018.

[96] D. B. West *et al.*, *Introduction to Graph Theory*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2001.

[97] Z. Wang, Z. Yu, C. L. P. Chen, J. You, T. Gu, H-S. Wong, and J. Zhang, "Clustering by Local Gravitation," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1383-1396, May. 2018.

[98] Y. Qian, F. Li, J. Liang, B. Liu and C. Dang, "Space structure and clustering of categorical data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 10, pp. 2047-2059, Oct. 2016.

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

[100] 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, Aug. 2004.

[101] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[102] Y. Zhan, D. Hu, H. Xing and X. Yu, "Hyperspectral Band Selection Based on Deep Convolutional Neural Network and Distance Density," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2365-2369, Dec. 2017.

[103] Y. Li, W. Xie and H. Li, "Hyperspectral image reconstruction by deep convolutional neural network for classification," *Pattern Recognit.*, vol 63, pp. 371-383, Mar. 2017

[104] Jang, Eric, Shixiang Gu and Ben Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.

[105] Abid, Abubakar, Muhammad Fatih Balin and James Zou, "Concrete autoencoders for differentiable feature selection and reconstruction," *arXiv preprint arXiv:1901.09346*, 2019.

[106] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[107] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," *Proceedings of the 27th international conference on machine learning*, pp. 807-814, 2010.

[108] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi and J. A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 9, pp. 6690-6709, Sept. 2019.

[109] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.

[110] L. Zhang, L. Zhang, B. Du, J. You, and D. Tao, "Hyperspectral image unsupervised classification by robust manifold matrix factorization," *Inf.Sci.*, vol. 485, pp. 154–169, Jun. 2019.

[111] L. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du, "Compression of hyperspectral remote sensing images by tensor approach," *Neurocomputing*, vol. 147, pp. 358–363, 2015.

[112] B. Arad and O. Ben-Shahar. "Sparse recovery of hyperspectral signal from natural rgb images," *ECCV*, 2016.

[113] B. Arad, O. Ben-Shahar, R. Timofte, L. Van Gool, L. Zhang, M.-H. Yang, et al. "Ntire 2018 challenge on spectral reconstruction from rgb images," *CVPRW*, 2018.

[114] Z. Shi, C. Chen, Z. Xiong, D. Liu and F. Wu,"HSCNN+: Advanced CNN-Based Hyperspectral Recovery from RGB Images," *CVPRW*, 2018.

[115] Z. Shi, C. Chen, Z. Xiong, D. Liu, Z. Zha and F. Wu, "Deep Residual Attention Network for Spectral Image Super-Resolution," ECCVW, 2018

[116] B. Kaya, Y. B. Can and R. Timofte, "Towards Spectral Estimation from a Single RGB Image in the Wild," *ICCVW*, 2019.

[117] L. Wang, T. Zhang, Y. Fu and H. Huang, "HyperReconNet: Joint Coded Aperture Optimization and Image Reconstruction for Compressive Hyperspectral Imaging," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2257-2270, May 2019.

[118] Y. Li, W. Xie and H. Li, "Hyperspectral image reconstruction by deep convolutional neural network for classification," *Pattern Recognit.*, vol. 63, pp. 371–383, Mar. 2017.

[119] G. Xia, et al., "DOTA: A Large-Scale Dataset for Object Detection in Aerial Images," *CVPR*, 2018.

Appendix A

Publications

A.1 Journal Publications

1) **H. Sun**, J. Ren, H. Zhao, Y. Yan, J. Zabalza, and S. Marshall, "Superpixel based Feature Specific Sparse Representation for Spectral-Spatial Classification of Hyperspectral Images," *Remote Sens.*, 11(5), 536, 2019.

2) **H. Sun**, J. Ren, H. Zhao, G. Sun, W. Liao, Z. Fang, and J. Zabalza, "Adaptive Distance based Band Hierarchy (ADBH) for Effective Hyperspectral Band Selection," *IEEE Trans. Cybern.*, doi:10.1109/TCYB.2020.2977750.

A.2 Conference Publications

1) **H. Sun**, Q. Zhu, J. Ren, D. Barclay, and W. Thompson, "Combining Image Analysis and Smart Data Mining for Precision Agriculture in Livestock Farming," *IEEE Smart Data.*, 2017.

2) Z. Fang, **H. Sun**, J. Ren, H. Zhao, S. Zhao, S. Marshall, and T. Durrani,

"3D Sensing Techniques for Multimodal Data Analysis and Integration in Smart and Autonomous Systems," *International Conference in Communications, Signal Processing, and Systems*, 2017

3) **H. Sun**, J. Ren, Y. Yan, J. Zabalza, and S. Marshall, "Joint kernelized sparse representation classification for hyperspectral imagery," *Hyperspectral Imaging Applications (HSI) 2018*, 2018.

4) J. Ren, **H. Sun**, Y. Huang, and H. Gao, "Knowledge-Based Multi-sequence MR Segmentation via Deep Learning with a Hybrid U-Net++ Model," *International Workshop on Statistical Atlases and Computational Models of the Heart*, 2019.

5) Z. Fang, J. Ren, S. Marshall, **H. Sun**, and J. Han, "SAFDet: A Semi-anchor-free Detector for Effective Detection of Oriented Objects in Aerial Images," Under review of *ECCV*

A.3 Journal Publications Under Preparation

1) **H. Sun**, J. Ren, et al. "Superpixel based Feature Specific Sparse Representation for Spectral-Spatial Classification of Hyperspectral Images," *Under preparation.*

2) **H. Sun**, J. Ren, et al. "Concrete Autoencoder (CAE-UBS) for Unsupervised Hyperspectral Band Selection," *Under preparation.*