arXiv:2004.00583v4 [cs.CV] 21 Dec 2020

Improving Deep Hyperspectral Image Classification Performance with Spectral Unmixing

Alan J.X. Guo and Fei Zhu*

Abstract

Recent advances in neural networks have made great progress in the hyperspectral image (HSI) classification. However, the overfitting effect, which is mainly caused by complicated model structure and small training set, remains a major concern. Reducing the complexity of the neural networks could prevent overfitting to some extent, but also declines the networks' ability to express more abstract features. Enlarging the training set is also difficult, for the high expense of acquisition and manual labeling. In this paper, we propose an abundance-based multi-HSI classification method. Firstly, we convert every HSI from the spectral domain to the abundance domain by a dataset-specific autoencoder. Secondly, the abundance representations from multiple HSIs are collected to form an enlarged dataset. Lastly, we train an abundance-based classifier and employ the classifier to predict over all the involved HSI datasets. Different from the spectra that are usually highly mixed, the abundance features are more representative in reduced dimension with less noise. This benefits the proposed method to employ simple classifiers and enlarged training data, and to expect less overfitting issues. The effectiveness of the proposed method is verified by the ablation study and the comparative experiments.

1 Introduction

Hyperspectral image (HSI) is a data cube consisting of reflection or radiance spectra, acquired by the remote sensors when flying over real-world objects or scenes. The height and width of an HSI are decided by the monitored scene at a specific resolution, while the depth records the measurements across a certain wavelength range. Thus, each pixel corresponds to a spectral vector. In the past decades, the HSI analysis has witnessed rapid development with plenty of applications [1, 2]. To effectively explore the rich spectral and spatial information contained in HSIs, different categories of processing techniques have been proposed, *e.g.*, the spectral unmixing [1], the classification [3], the image restoration [4], and the target detection [5]. In this paper, we particularly address the deep learning-based HSI classification problem, where the spectral unmixing technique is smartly integrated to the classification task, such that the classification model is simplified and the training set is amplified. This helps to improve the classification performance.

The HSI classification refers to classifying each of the pixels to a certain class according to the spectral and spatial characteristics. Earlier approaches were developed based on conventional

*A. Guo, and F. Zhu are with the Center for Applied Mathematics, Tianjin University, China. (jiaxiang.guo; fei.zhu@tju.edu.cn)

machine learning algorithms, including the principal component analysis (PCA) [6], the independent component analysis (ICA) [7], the linear discriminant analysis (LDA) [8], the support vector machine (SVM) [9], and the sparse representation [10], to name a few. Despite the great progress recently brought by the neural networks (NN) to HSI classification, the none-NN methods continue to play a role independently or as a part of the NN-based algorithms [11, 12, 13, 14], on account of the drawbacks of the NN, *e.g.*, the overfitting problem on small training sets [15].

Spectral unmixing (SU) is another active area of research in HSI analysis. It is assumed that each spectrum is a mixture of several "pure" material signatures, termed endmembers. The aim of SU is to extract the endmembers and to estimate their respective proportions, namely the abundance fractions, at each pixel [16]. Recently, several works have introduced the SU as a complementary source of information in HSI classification. In [17, 18], the SU was applied to reduce the spectral dimension, in order to avoid the Hughes phenomenon when applying the SVM classifier. In [19], the authors investigated several SU methods and assigned the label to a pixel according to its maximum abundance. In [20, 21], the extracted abundances were used as supplementary information to improve the classification accuracy on the hard samples, namely the highly-mixed spectra. The authors in [22] considered a region-based nonnegative matrix factorization for band group based abundance estimation. The abundance matrices at different ranges of wavelengths were used as input to train a convolutional neural network (CNN) based classifier. Moreover, in the scope of HSI classification by semisupervised learning, the SU was applied in selecting the most informative samples [23, 24, 25, 26].

The NN has gained great popularity and achieved remarkable results in many machine learning fields, *e.g.,* the computer vision, especially after the introduction of CNN and deep learning [27, 28]. Since then, numerous investigations have been made to apply deep learning-based algorithms to the HSI analysis. In the [29, 30], the authors used NN-based model to conduct band selection tasks. The researchers of [31, 32] applied the NN-based autoencoder for SU on HSIs. In the field of HSI classificaiton, many of the early works considered to utilize the stacked autoencoder (SAE) to extract denoised or sparse features from the spectral or spatial-spectral data, and the obtained features were usually classified with a traditional classifier, such as the SVM and the logistic regression [33, 34, 35, 36, 37]. More recently, several end-to-end classification methods based on CNN [38, 39, 40, 41, 42] and the recurrent neural network (RNN) [43, 44] have been proposed and significantly improved the classification results. See [15] for an overview of the deep learning methods for HSI classification.

To train a deep learning-based model typically requires a large amount of labeled data, otherwise, the learned model would be prone to overfitting. However, the availability of training samples is limited in HSIs due to the high expense of acquisition and manual labeling [39]. To tackle this contradiction, different strategies have proved their effectiveness in existing works, *e.g.*, data augmentation and transfer learning. In [39], a virtual sample enhanced method was proposed to improve the performance of the CNN-based model. In [45], the authors designed a pixel-pair-based model, where the training set is composed of pixel-pairs instead of pixels. This ensures the sufficiency of labeled samples for training a deep CNN. Alternatively, transfer learning has been employed to alleviate the overfitting issue, that is, by transferring the knowledge acquired from the source domain to the target domain, the demand of training samples would be reduced. Knowledge was transferred from the ordinary RGB images to HSI classification tasks [40], and transferred from multiple HSIs to the classification tasks on small-scale HSIs [46]. Moreover, the authors in [47] applied the knowledge learned from unsupervised tasks to classification tasks on

the same HSI, by transferring a pre-trained stacked denoising autoencoder and fine-tuning on the labeled samples.

In this paper, we propose an abundance-based multi-HSI classification (ABMHC) method, which alleviates the overfitting issues by taking advantage of the abundance information from multiple HSIs. To be precise, the proposed method benefits from the SU in two perspectives.

- **Simple network structure:** The SU maps the HSI from the spectral domain to the abundance domain. Different from the spectra that are usually highly mixed, the abundance features are more representative in reduced dimension with less noise [1]. By performing classification on the abundance-based features, the original classification tasks are expected to be simplified, which enables the use of simple networks. It is noteworthy that simple networks usually have less overfitting issues [48].

- **Enlarged training set:** Transforming the HSIs into the abundance domain will eliminate the data-specific information in different HSIs, *e.g.,* the type of sensor and the spatial-spectral resolutions. By considering a unified and relatively large number of endmembers in SU of different HSIs, the estimated abundance features of different HSIs are with the same dimension. This ensures the construction of an enlarged training set, that gathers the labeled data from all the HSIs in this study for the subsequent classifier.

The proposed ABMHC is generally composed of two featured procedures, namely 1) SU with deep autoencoder network; 2) CNN-based classification with extracted abundances. Briefly, by the dataset-wise autoencoder-based SU algorithm, the spectra from different HSIs are firstly encoded into abundance vectors, that are in the identical dimension. After that, the abundance vectors from different HSIs are processed to construct an enlarged dataset. Lastly, a CNN-based classifier is trained based on the abundance patches from the enlarged dataset. The flowchart of the proposed ABMHC is given in Fig. 1.

The main contributions of the proposed ABMHC method are summarized as the following aspects.

- We verify that performing classification over abundance representations facilitates the use of simple networks, without deteriorating the performance.

- We verify that the classification performance is improved using a unified dataset constructed from unrelated HSIs, compared with using each single HSI.

- We propose a method termed ABMHC that fulfills the aforementioned motivations of simplifying network structure and enlarging training set. The proposed ABMHC is comparable to the state-of-art methods on several datasets.

The remainder of this paper is organized as follows. Section 2 briefly presents the SU model used in this paper. Section 3 presents the spectral unmixing stage with the autoencoder network, while Section 4 presents the classification stage with CNN. Section 5 reports the evaluation of the proposed method by ablation study and comparative experiments. Conclusions are drawn in Section 6.

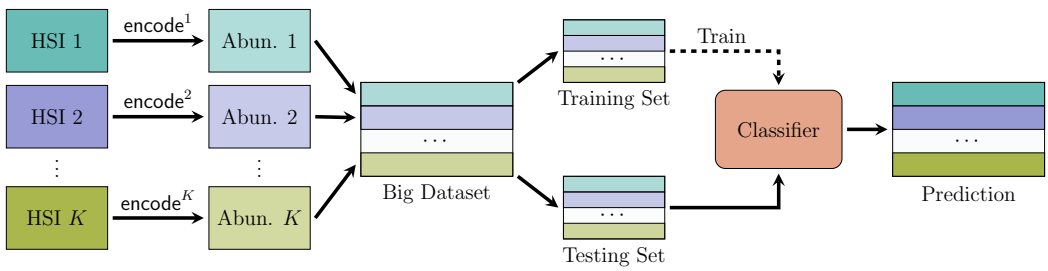

Figure 1: Flowchart of the proposed abundance-based multi-HSI classification method.

2 Notations in Spectral unmixing

The SU consists of decomposing each observed spectrum as a mixture of endmembers with their proportions being abundances. According to different underlying mixing mechanisms, the SU models and associated algorithms are roughly divided into the linear and the nonlinear ones. Extensive SU models and algorithms have been proposed, as reviewed in [1, 49]. Of particular note is the recent applications of deep autoencoders for SU, as investigated in [50, 31, 32, 51, 52, 53]. In this section, we succinctly present the SU model to be considered in this paper, which is proposed in [31, 32].

Given an HSI, let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{B\times N}$ be a matrix composed by N observed spectra over B bands, where $\boldsymbol{x}_i \in \mathbb{R}^B$ is the i-th spectrum vector, for $i = 1, 2, ..., N$. Assume that the HSI is known to be mixed by R endmembers. Let $\boldsymbol{M} = [\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_R] \in \mathbb{R}^{B\times R}$ represent the endmember matrix, with $\boldsymbol{m}_i$ being the spectrum of the i-th endmember. The abundance vector associated with the i-th pixel is denoted as $\boldsymbol{a}_i = [a_{i,1}, a_{i,2}, \ldots, a_{i,R}]'$, its entry $a_{i,j}$ being the fractional abundance w.r.t. the j-th endmember, The linear mixing model (LMM) assumes each observed pixel to be represented as a linear combination of the endmembers, with

$$\boldsymbol{x}_i = \boldsymbol{M}\boldsymbol{a}_i + \boldsymbol{n}, \tag{1}$$

where $\boldsymbol{n} \in \mathbb{R}^B$ is the additive noise vector. Similar to [32], this paper considers a generalized SU model that combines the LMM and an additive nonlinear model, given by

$$\boldsymbol{x}_i = \lambda\boldsymbol{M}\boldsymbol{a}_i + (1-\lambda)\Phi(\boldsymbol{M}, \boldsymbol{a}_i) + \boldsymbol{n}, \tag{2}$$

where Φ is a nonlinear function that characterizes the interactions between the endmembers, parameterized by the abundance vector, and λ is a hyperparameter balancing the weights of the linear and nonlinear parts. To satisfy a physical interpretation, both the abundance nonnegativity constraint (ANC) and abundance sum-to-one constraint (ASC) are enforced to the model, which are

$$\begin{cases} a_{i,j} \geq 0, & j = 1, 2, \ldots, R; \\ \sum_{j=1}^{R} a_{i,j} = 1. \end{cases} \tag{3}$$

3 Spectral unmixing with deep autoencoder network

In this section, we introduce a deep autoencoder, the encoder of which mimics the generalized SU procedure in (2), to estimate the abundance representations from the HSI. The proposed deep

autoencoder follows the same procedure as in [31, 32], but has different network structures and implementation.

Basically, an autoencoder is composed of two parts, namely an encoder and a decoder. The encoder, $\mathsf{encode} : \mathbb{R}^B \mapsto \mathbb{R}^R$, maps a sample from the input space to the feature space by

$$\hat{\boldsymbol{a}} = \mathsf{encode}(\boldsymbol{x}). \tag{4}$$

In most cases, the dimension of the input space is higher than that of the feature space, *i.e.,* $R < B$, which indicates that the encoder compresses the information from input vector $\boldsymbol{x}$ to feature vector $\hat{\boldsymbol{a}}$. The decoder, $\mathsf{decode} : \mathbb{R}^R \mapsto \mathbb{R}^B$, maps the feature vector $\hat{\boldsymbol{a}}$ to an approximation $\hat{\boldsymbol{x}}$ of the original sample $\boldsymbol{x}$, from a low dimensional space to a high dimensional space, by

$$\hat{\boldsymbol{x}} = \mathsf{decode}(\hat{\boldsymbol{a}}), \tag{5}$$

where $\hat{\boldsymbol{x}}$ represents the reconstructed sample. The whole reconstruction procedure from $\boldsymbol{x}$ to $\hat{\boldsymbol{x}}$ by the autoencoder is formulated as

$$\hat{\boldsymbol{x}} = \mathsf{decode}(\mathsf{encode}(\boldsymbol{x})). \tag{6}$$

When the autoencoder is good enough so that the input sample $\boldsymbol{x}$ and the reconstruction $\hat{\boldsymbol{x}}$ are similar under some metric, it is inferred that the feature vector $\hat{\boldsymbol{a}}$ retrieves most of the information from $\boldsymbol{x}$. Specifically, when the feature vector $\hat{\boldsymbol{a}}$, namely the output of the encoder, satisfies both the ANC and the ASC in (3), the encoder itself is interpreted as a blind SU procedure, and $\hat{\boldsymbol{a}}$ is taken as the abundance vector.

In this paper, both the encoder and the decoder are realized by NN. Let θ_e and θ_d be the learnable parameters of the encoder network and the decoder network, respectively. We use the notations $\mathsf{encode}(\cdot;\theta_e)$ and $\mathsf{decode}(\cdot;\theta_d,\boldsymbol{M})$ to represent the encoder and the decoder, where the endmember matrix $\boldsymbol{M}$ is another part of learnable parameters in the decoder.

The structure of the encoder $\mathsf{encode}(\cdot;\theta_e)$ is shown in Fig. 2. It has 4 layers in total, namely two 1D convolutional layers [54], one fully connected layer, and one normalization layer. The 1D convolutional layer operates similarly as the plain 2-D convolution, but the convolutional operation is limited to one dimension. In this paper, the 1D convolutional layers are set with kernel size 3, stride 1, followed by the ReLU activations. The normalization layer is used to impose ANC and ASC to the encoded abundance feature $\hat{\boldsymbol{a}} = [\hat{a}_1, \hat{a}_2, ..., \hat{a}_R]'$ by

$$\hat{a}_i = \frac{|\hat{a}_i|}{\sum_{j=1}^{R} |\hat{a}_j|}, \tag{7}$$

as suggested in [31].

The structure of the decoder $\mathsf{decode}(\cdot;\theta_d,\boldsymbol{M})$ is more complicated. It is designed to consist of two parts, that correspond to the linear and nonlinear mixing models, in accordance with the latent mixing mechanism in (2). As illustrated in Fig. 3, the upper part corresponds to the LMM, expressed by $\boldsymbol{M}\boldsymbol{a}$, while the lower part represents the nonlinear mixing model given by $\Phi(\boldsymbol{M},\boldsymbol{a})$. For the nonlinear part, we first multiply each endmember by its fractional abundance, thus generating a set of weighted endmembers, given by $[\hat{a}_1\boldsymbol{m}_1, \hat{a}_2\boldsymbol{m}_2, \ldots, \hat{a}_R\boldsymbol{m}_R]$. Later, the weighted endmembers flow through five 1D convolutional layers with different numbers of output channels and end up with a vector of length B. All the five 1D convolutional layers are set to

have kernel size 1 and stride 1. This setting ensures that the effect of the 1D convolutional layer could be interpreted by the interaction between the channels of the input data. In view of this, the nonlinear part of the decoder simulates the interactions between the endmember signatures. Each of the 1D convolutional layers is followed by the ReLU activation, except for the last one. In the end, the reconstructed spectrum is estimated by the weighted sum of the linear and nonlinear estimations.

In the proposed method, for each HSI scene, we propose to train a specific pair of encoder and decoder to represent the blind SU procedure and the spectrum reconstruction, respectively. The encoder $\mathsf{encode}(\boldsymbol{x};\theta_e)$ maps the spectrum $\boldsymbol{x}$ to its abundance vector $\hat{\boldsymbol{a}} = \mathsf{encode}(\boldsymbol{x};\theta_e)$, where θ_e represents the network parameters with the implicit endmember matrix $\boldsymbol{M}$. The decoder $\mathsf{decode}(\hat{\boldsymbol{a}};\theta_d,\boldsymbol{M})$ takes the abundance vector $\hat{\boldsymbol{a}}$ as inputs and outputs the reconstructed spectrum $\hat{\boldsymbol{x}} = \mathsf{decode}(\hat{\boldsymbol{a}};\theta_d,\boldsymbol{M})$, and contains two parts of parameters, namely the parameters of the neural network θ_d and the explicit endmember matrix $\boldsymbol{M}$. To learn the parameters θ_e, θ_d and endmember matrix $\boldsymbol{M}$, we jointly optimize the encoder and decoder

$$\hat{\boldsymbol{x}} = \mathsf{decode}(\mathsf{encode}(\boldsymbol{x};\theta_e);\theta_d,\boldsymbol{M}), \tag{8}$$

by minimizing the reconstruction error

$$\mathcal{L}_r = \frac{\sum_{i=1}^{N} ||\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i||_2^2}{N}, \tag{9}$$

where $\boldsymbol{x}_i$ is the spectrum from the HSI scene and $\hat{\boldsymbol{x}}_i$ is the reconstructed spectrum corresponding to $\boldsymbol{x}_i$. To optimize (9), different gradient-based algorithms can be applied [55, 56, 57]. The endmember matrix $\boldsymbol{M}$ is initialized by the vertex component analysis (VCA) [58], and the paramerters θ_e and θ_d are initialized by the uniform distribution, as in [31]. The optimized parameters are denoted as $\hat{\theta}_e, \hat{\theta}_d$ and $\hat{\boldsymbol{M}}$, and hereafter we use $\mathsf{encode}(\cdot)$ and $\mathsf{decode}(\cdot)$ to denote the learned models $\mathsf{encode}(\cdot;\hat{\theta}_e)$ and $\mathsf{decode}(\cdot;\hat{\theta}_d,\hat{\boldsymbol{M}})$, respectively, unless otherwise stated.

Before classification, we utilize every learned encoder to map its corresponding HSI to the abundance representation by

$$\hat{\boldsymbol{A}} = \mathsf{encode}(\boldsymbol{X}). \tag{10}$$

Accounting for the subsequent classification tasks on multiple HSIs, we have the following settings on the autoencoder-based SU procedure. The number of endmembers is set to be $R = 16$ for all the autoencoders, a value larger than the real numbers of endmembers for most HSIs. The redundancy of endmembers ensures that the abundances could have the capacity to retrieve most of the information from spectra. The hyperparameter λ is set to be 0.5, following [31, 32]. The HSIs are normalized to the range $[0, 1]$ before fed to the autoencoder. As the research focus of this work is not the SU model and method, the analysis of the effects of these hyperparameters is omitted. Readers may refer to [31] for a detailed hyperparameter analysis of a similar SU procedure. The autoencoder is optimized by the Adam algorithm [56]. The structures of networks are designed and realized with AutoKeras [59] and Tensorflow [60].

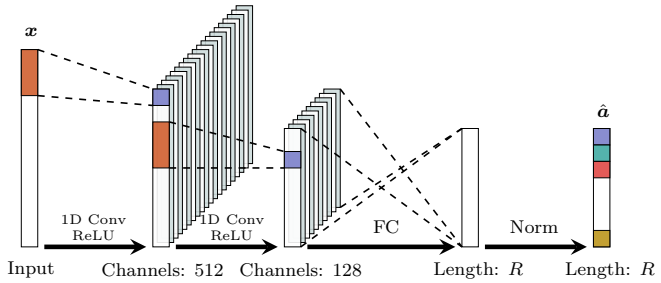

Figure 2: Structure of the encoder.

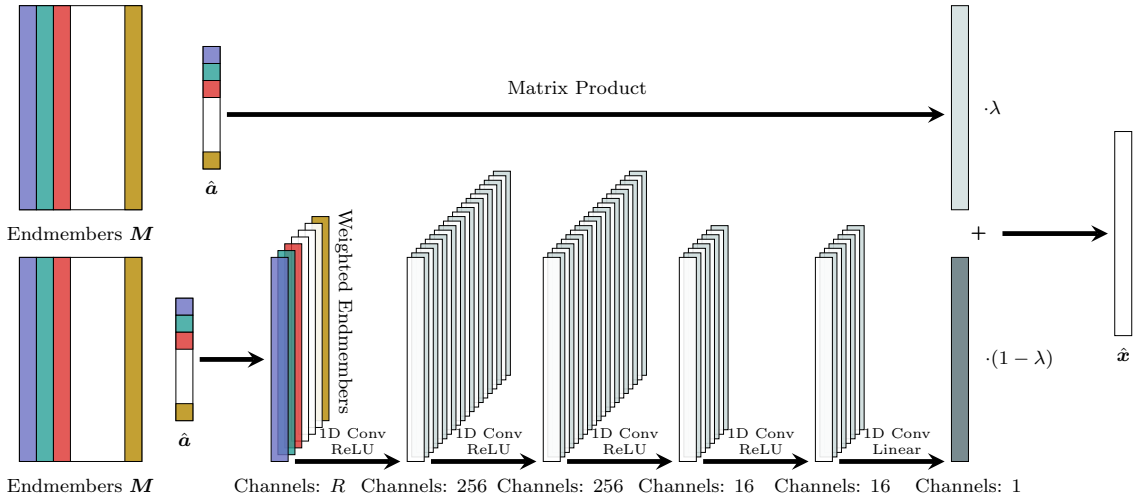

Figure 3: Structure of the decoder.

4 Multi-HSI classification with convolutional neural network

In this section, we use a simple CNN model based on both the spatial information and the abundance representations to jointly classify multiple HSIs. To alleviate the overfitting issue, merging different HSIs into one big dataset is one of the most fundamental motivations of this paper. Different from existing CNN models, which process the raw data from single HSI, the proposed algorithm is capable to process the abundance data from multiple HSIs simultaneously.

4.1 Preparation of training and testing data

To construct a big dataset from different HSIs, we propose the following processings of the autoencoder-extracted abundance representations. Given K HSIs to be classified, namely $\{\boldsymbol{X}^k | \, k = 1, 2, \ldots, K\}$, assume that $\boldsymbol{X}^k$ contains C^k labeled classes. By training an autoencoder $\mathsf{decode}^k(\mathsf{encode}^k(\cdot))$, which is described in Section 3, for every HSI separately, we obtain the following abundance representations

$$\hat{\boldsymbol{A}}^k = \mathsf{encode}^k(\boldsymbol{X}^k), \quad k = 1, 2, \ldots, K. \tag{11}$$

To take advantage of both spatial context and abundance information for improving the classification performance, each abundance representation $\hat{\boldsymbol{A}}^k$ is firstly divided into the labeled abundance patches

$$\{(\hat{\mathfrak{a}}_i^k, y_i^k) |\, i = 1, 2, \ldots, N^k\}, \tag{12}$$

where $\hat{\mathfrak{a}}_i^k$ represents an abundance patch from $\hat{\boldsymbol{A}}^k$, and corresponds to a pixel patch in the original image $\boldsymbol{X}^k$. The label y_i^k is selected as the label of the pixel patch center, and ranges from 0 to $C^k - 1$.

Assume we have following K sets of labeled abundance patches generated from K HSIs,

$$\begin{aligned}
\{(\hat{\mathfrak{a}}_i^1, y_i^1) |\, i = 1, 2, \ldots, N^1\}, &\qquad y_i^1 \in \{0, 1, \ldots, C^1 - 1\}; \\
\{(\hat{\mathfrak{a}}_i^2, y_i^2) |\, i = 1, 2, \ldots, N^2\}, &\qquad y_i^2 \in \{0, 1, \ldots, C^2 - 1\}; \\
&\vdots \\
\{(\hat{\mathfrak{a}}_i^K, y_i^K) |\, i = 1, 2, \ldots, N^K\}, &\qquad y_i^K \in \{0, 1, \ldots, C^K - 1\}.
\end{aligned}$$

A big merged dataset S is constructed by collecting all sets of labeled abundance patches, with labels rearranged to avoid overlap. To be precise, the labeled abundance patch $(\hat{\mathfrak{a}}_i^k, y_i^k)$ is relabeled to

$$\left(\hat{\mathfrak{a}}_i^k, y_i^k + \sum_{j=1}^{k-1} C^j\right) \tag{13}$$

before collected into S. By doing so, in the merged dataset, each HSI occupies a specific interval of integers as class labels, without mutual overlaps of labels with other HSIs. In summary, the big dataset S assembles all the samples from $\hat{\boldsymbol{A}}^k, k = 1, 2, \ldots, K$, with $C = \sum_{k=1}^{K} C^k$ being the total number of classes.

4.2 Classification with CNN

Given a dataset S and a sample $(\mathfrak{a}, y) \in S$ to classify, the classification task is interpreted as finding a function $\mathsf{classify}(\cdot)$ that maps the abundance patch $\mathfrak{a}$ to the correct label y. This task is realized by a simple neural network that consists of most of the well-known CNN layers. As illustrated in Fig. 4, the network takes abundance data, which has R channels, as input. Later, the data flows through three CNN layers, each followed by a ReLU activation. These CNN layers are set with kernel size 3×3 and stride 1. In the last two steps, a fully connected layer maps the data into a vector with length C, namely the total number of classes in S; and a softmax layer finally transforms the vector into the output with one-hot style. The softmax function produces the predicted probability distribution of sample $\mathfrak{a}$ as follows

$$\mathsf{softmax}(\boldsymbol{x})_i = \frac{\exp(x_i)}{\sum_{j=0}^{C-1} \exp(x_j)}, \quad i = 0, 1, \ldots, C-1. \tag{14}$$

Separate the merged dataset S into the training set S_{tr} and the testing set S_{te}. Let θ be the parameters of the proposed CNN, and $\hat{y}$ be the prediction on $\mathfrak{a}$. In the training stage, the parameters θ are optimized by minimizing the following cross-entropy loss

$$\mathcal{L}_s = \sum_{(\mathfrak{a}, y) \in S_{\text{tr}}} \sum_{i=0}^{C-1} -y_i \log \hat{y}_i, \tag{15}$$

over the training set, where $\boldsymbol{y} = [y_0, y_1, \ldots, y_{C-1}]'$ and $\hat{\boldsymbol{y}} = [\hat{y}_0, \hat{y}_1, \ldots, \hat{y}_{C-1}]'$ are the one-hot encodes of y and $\hat{y}$, respectively. After minimizing (15) by gradient-based optimization, the optimized parameters $\hat{\theta}$ are obtained. Hereafter the trained classifier $\mathsf{classify}(\cdot; \hat{\theta})$ is abbreviated by $\mathsf{classify}(\cdot)$.

In the testing stage, when a testing sample $\mathfrak{a}$ is fed into the trained classifier $\mathsf{classify}(\cdot)$, the one-hot style prediction is generated as

$$\begin{aligned} \hat{\boldsymbol{y}} &= \mathsf{classify}(\mathfrak{a}) && (16) \\ &= [\hat{y}_0, \hat{y}_1, \ldots, \hat{y}_{C-1}]'. && (17) \end{aligned}$$

In most of the existing classification models, the final prediction $\hat{y}$ is calculated by

$$\hat{y} = \arg\max_i \{\hat{y}'_i | \, i = 0, 1, \ldots, C-1\}. \tag{18}$$

However, in this paper, as we know in advance from which HSI the sample comes, an elaborated strategy is applied to further improve the accuracy of testing. Assume the testing sample $\mathfrak{a}$ in known to be from the abundance representation $\boldsymbol{A}^k$ of the k-th HSI. The predicted label of $\mathfrak{a}$ is calculated by performing $\arg\max$ function merely on the fragment of $\hat{\boldsymbol{y}}$ corresponding to $\boldsymbol{A}^k$, by

$$\hat{y} = \arg\max_i \left\{ \hat{y}'_i \,\middle|\, i = \sum_{j=1}^{k-1} C^j, \sum_{j=1}^{k-1} C^j + 1, \ldots, \sum_{j=1}^{k} C^j - 1 \right\}. \tag{19}$$

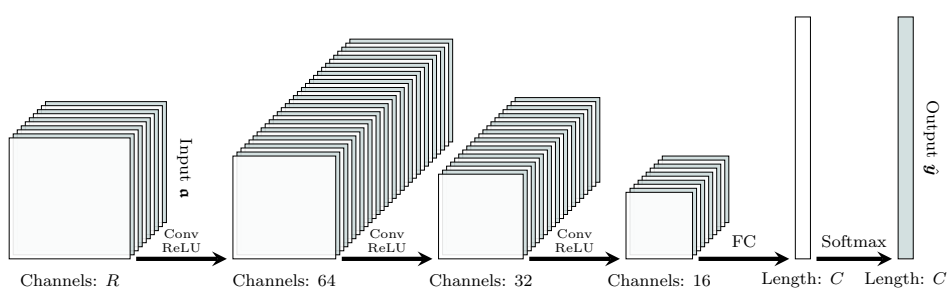

Figure 4: Structure of the classifier.

5 Experiments

In this section, we perform a series of experiments including the ablation study and comparative study with several state-of-the-art methods on four public HSI datasets, to verify the effectiveness of the innovative ideas and to demonstrate the performance of the proposed method.

5.1 Datasets

In this paper, experiments are performed on four public HSI datasets, *i.e.*, the Paiva University scene[1], the Pavia Centre scene[1], the Salinas scene[1], and the Houston2018 scene[2] (grss_dfc_2018)

[1]The datasets are available online: `http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes`

[2]This data is from 2018 IEEE GRSS Data Fusion Contest, where only the training set is used in this paper. The dataset is available online: `http://www.grss-ieee.org/community/technical-committees/data-fusion`

[61].

The Pavia University scene is acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor. Removing the noisy bands and a blank strip, the data size in format height × width × depth is 610 pixel × 340 pixel × 103 band. The spatial resolution is about 1.3 meters. As shown in TABLE 1, the pixels are labeled with 9 classes. In practice, 200 × 9 labeled samples are chosen to form the training set, while the rest of the labeled samples form the testing set. The non-labeled pixels constitute the backgrounds. Fig. 5 depicts the false color composite and the representation of groundtruth.

Table 1: Reference classes and sizes of training and testing sets of Pavia University image

No.	Class	Cardinality	Train	Test
1	Asphalt	6631	200	6431
2	Meadows	18649	200	18449
3	Gravel	2099	200	1899
4	Trees	3064	200	2864
5	Painted metal sheets	1345	200	1145
6	Bare Soil	5029	200	4829
7	Bitumen	1330	200	1130
8	Self-Blocking Bricks	3682	200	3482
9	Shadows	947	200	747
	Total	42776	1800	40976

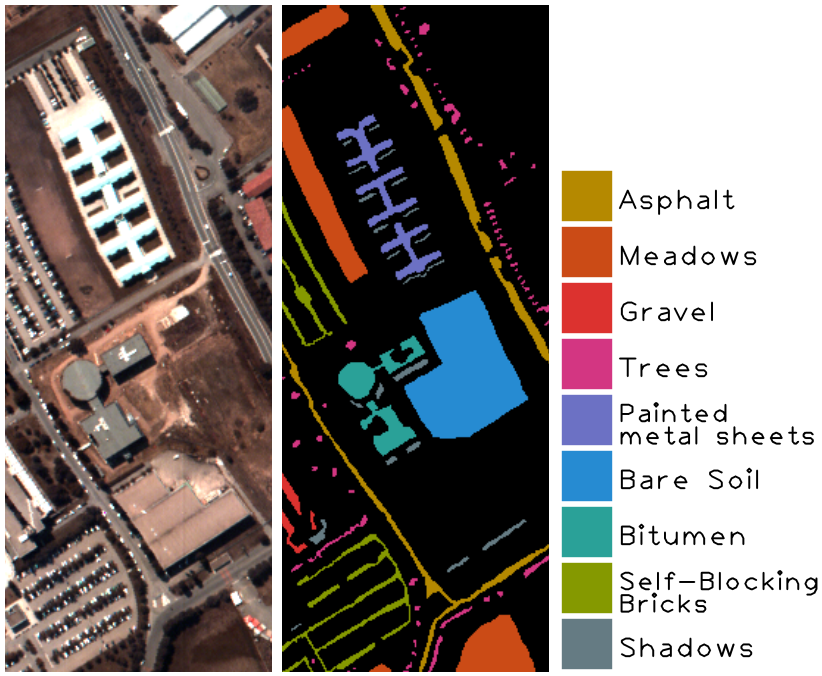

Figure 5: The false color composite (band $10, 20, 40$) and groundtruth representation of Pavia University

The Pavia Centre scene is also acquired by the ROSIS sensor. The data size is 1096 pixel ×

715 pixel × 102 band after removing the noisy bands. The spatial resolution is also 1.3 meters. As illustrated in TABLE 2, the pixels in the Pavia University scene are labeled with 9 classes. The constructions of the training and testing set are in the same way as in the Pavia University scene. Fig. 6 illustrates the false color composite and the representation of groundtruth.

Table 2: Reference classes and sizes of training and testing sets of Pavia Centre image

No.	Class	Cardinality	Train	Test
1	Water	65971	200	65771
2	Trees	7598	200	7398
3	Asphalt	3090	200	2890
4	Self-Blocking Bricks	2685	200	2485
5	Bitumen	6584	200	6384
6	Tiles	9248	200	9048
7	Shadows	7287	200	7087
8	Meadows	42826	200	42626
9	Bare Soil	2863	200	2663
	Total	148152	1800	146352

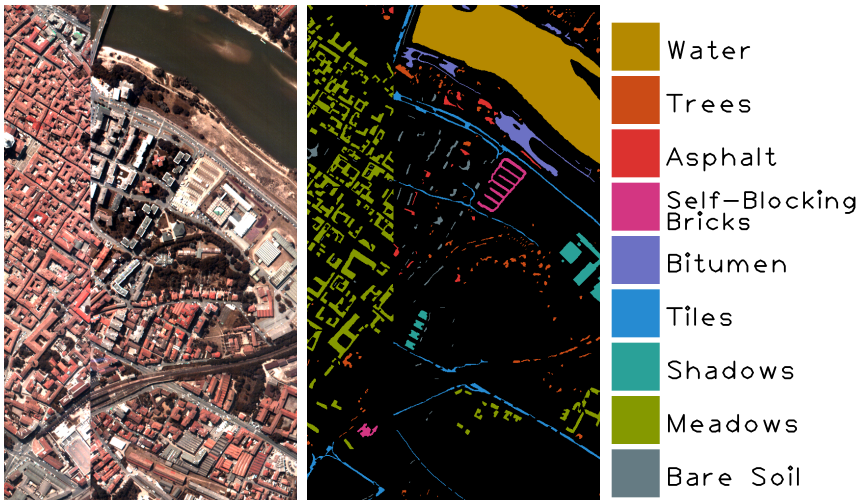

Figure 6: The false color composite (band $10, 20, 40$) and groundtruth map of Pavia Centre

The Salinas scene is collected by the Airborne Visible Infrared Imaging Spectrometer (AVIRIS). After the removal of the water absorption bands, the remained HSI has a size of 512 pixel × 217 pixel × 204 band. The pixels are classified into 16 categories, and 200×16 samples are picked for training, as shown in TABLE 3. The false color composite and the representation of groundtruth is shown in Fig. 7.

The Houston2018 (grss_dfc_2018) scene is acquired by the National Center for Airborne Laser Mapping over the University of Houston campus and its neighborhood. The size of this HSI is 601 pixel × 2384 pixel × 48 band, with a 1-meter ground sample distance. However, the groundtruth matrix has a quadrupled size 1202 pixel × 4768 pixel with a 0.5-meter ground sample distance. In practice, the label for each pixel is determined by the largest vote strategy using the groundtruth

Table 3: Reference classes and sizes of training and testing sets of Salinas image

No.	Class	Cardinality	Train	Test
1	Brocoli green weeds 1	2009	200	1809
2	Brocoli green weeds 2	3726	200	3526
3	Fallow	1976	200	1776
4	Fallow rough plow	1394	200	1194
5	Fallow smooth	2678	200	2478
6	Stubble	3959	200	3759
7	Celery	3579	200	3379
8	Grapes untrained	11271	200	11071
9	Soil vinyard develop	6203	200	6003
10	Corn senesced green weeds	3278	200	3078
11	Lettuce romaine 4wk	1068	200	868
12	Lettuce romaine 5wk	1927	200	1727
13	Lettuce romaine 6wk	916	200	716
14	Lettuce romaine 7wk	1070	200	870
15	Vinyard untrained	7268	200	7068
16	Vinyard vertical trellis	1807	200	1607
	Total	54129	3200	50929

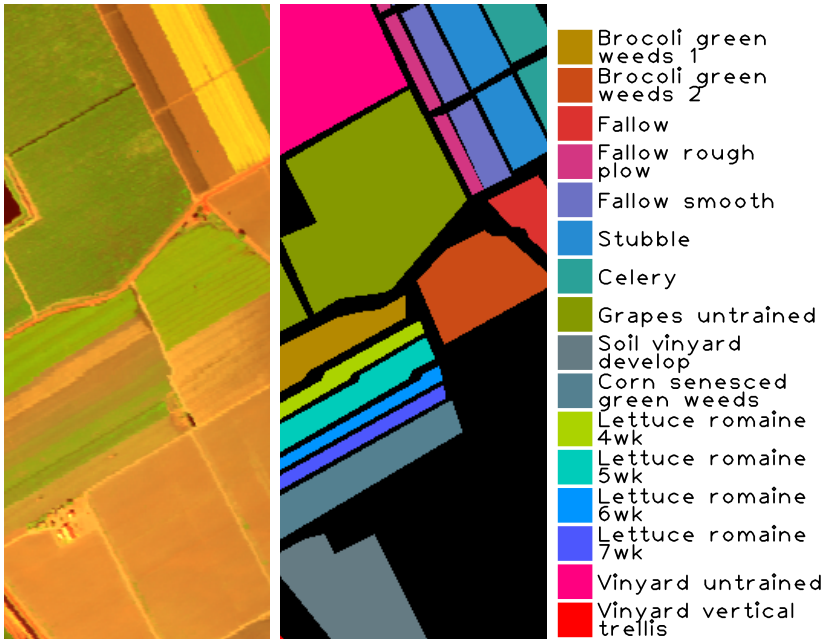

Figure 7: The false color composite (band $180, 100, 10$) and groundtruth representation of Salinas

matrix. As illustrated in TABLE 4, there are 20 classes in the grss_dfc_2018, and a large variance exists among the sample numbers of different classes. For each class, the size of the training set is chosen as 20% of the samples, truncated by 3200.

Finally, for all the datasets, the number of training samples in each class is enlarged to 3200

by data augmentation, using rotation, mirroring and duplicating.

Table 4: Reference classes and sizes of training and testing sets of grss_dfc_2018 image

No.	Class	Cardinality	Train	Test
1	Healthy grass	9799	1959	7840
2	Stressed grass	32502	3200	29302
3	Artificial turf	684	136	548
4	Evergreen trees	13107	2621	10486
5	Deciduous trees	4810	962	3848
6	Bare earth	4516	903	3613
7	Water	266	53	213
8	Residential buildings	38268	3200	35068
9	Non-residential buildings	221145	3200	217945
10	Roads	41178	3200	37978
11	Sidewalks	28609	3200	25409
12	Crosswalks	1399	279	1120
13	Major thoroughfares	44933	3200	41733
14	Highways	9507	1901	7606
15	Railways	6937	1387	5550
16	Paved parking lots	10725	2145	8580
17	Unpaved parking lots	129	25	104
18	Cars	4835	967	3868
19	Trains	4622	924	3698
20	Stadium seats	6824	1364	5460
	Total	484795	34826	449969

5.2 Ablation study and comparative experiments

We design the ablation study to verify that the effectiveness of the proposed ABMHC is mainly attributed to the following two factors: 1) The abundance features extracted by autoencoder-based SU have more discriminative ability than the raw spectra, hence they are better classified by the CNN-based classifier; 2) The combination of abundance representations from different HSIs yields a compatible classifier that is more powerful than the data-specific classifier. As a baseline, we train a CNN-based classifier directly on the raw spectral data for each HSI, and term this method as raw-CNN. Besides, the abundance-based HSI classification is performed on each dataset, and we refer this series of experiments as abun-CNN. The first aforementioned factor can be evaluated by comparing the results of raw-CNN and abun-CNN. Finally, we perform the abundance-based and multi-HSIs-based algorithm on the merged big training set, which is the proposed ABMHC. To conduct the ablation study, the SU procedure is set to be the same for the abun-CNN and the ABMHC, namely for every HSI scene, same endmember matrix and abundance representation are considered in both methods. The only difference between the abun-CNN and

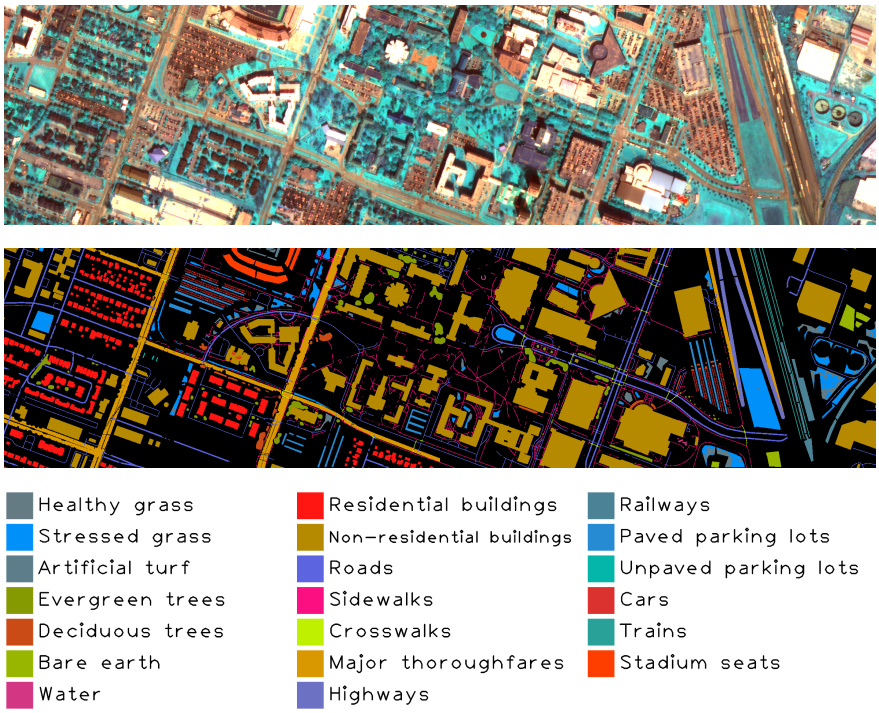

Figure 8: The false color composite (band $48, 28, 8$) and groundtruth representation of grss_dfc_2018

the ABMHC is whether the classifier is involved with multi-HSIs.[3] To keep a fair comparison, the same network structures and hyperparameters are utilized in raw-CNN, abun-CNN and ABMHC, as explicated in Section 3 and Section 4. In the experiments of abun-CNN and ABMHC, the parts of SU are identity. The size of abundance patches and HSI patches adopted in the CNNs of the proposed ABMHC and its ablation study is set to $11\,\text{pixel} \times 11\,\text{pixel}$. The effect of this parameter on classification performance is not analyzed in this paper, as investigations have already been made in several existing works [62, 63].

To further evaluate the performance of proposed ABMHC, we choose three lately-proposed classification algorithms for comparison, namely the method of pixel-pair feature (PPF)[45], the spectral-spatial squeeze-and-excitation residual bag-of-feature learning (S3EResBoF)[64], and the hybrid spectral net (HybridSN)[65]. All these state-of-the-art methods are based on deep learning with CNN and performed on raw spectra of HSIs. They have shown promising classification results on several HSIs datasets. For fairness, all the comparing methods are performed using the training sets with the same size, as described in Section 4.1, except for the PPF algorithm on grss_dfc_2018. In fact, the PPF generates pixel pairs as training samples, so that the size of the training set is squared. This leads to an out-of-memory situation on our server equipped with 256G RAM. In practice, the original training set for PPF on grss_dfc_2018 is constructed by choosing 20% of the labeled samples from each class, and truncating the number by 1600. For this reason, the performance of PPF on this dataset is not satisfactory, as to be given in TABLE 8. To roughly

[3]To train a compatible classifier directly using the collection of raw spectra from different datasets cannot be realized, as the number of bands varies in different HSIs.

explore the effects of introducing abundance representation on complex structured networks, we also perform the HybridSN using the abundance features extracted by autoencoder-based SU on all the four HSI datasets.

5.3 Results analysis

We apply three commonly used metrics to evaluate the performances of all the algorithms, namely OA, AA, and κ. The overall accuracy (OA) represents the ratio of the correctly classified samples number to the total samples number; the average accuracy (AA) is the mean accuracy of different classes; the Cohen's kappa coefficient κ measures the agreement between the predicted labels and the groundtruth labels.

Table 5: classification accuracies (averaged over 5 runs) on Pavia University scene

No.	raw-CNN	abun-CNN	ABMHC	PPF	S3EResBoF	HybridSN(raw/abun)
1	94.59	96.45	97.99	97.25	99.35	**99.74**/99.67
2	95.90	98.12	98.87	95.24	**99.88**	99.43/99.63
3	92.70	96.46	**98.87**	94.17	98.79	97.22/97.22
4	98.65	98.58	**98.86**	97.20	88.04	96.90/96.24
5	**100.00**	99.90	99.98	**100.00**	99.64	99.95/98.84
6	94.54	99.34	**99.40**	99.37	99.02	98.50/98.21
7	97.86	98.69	**99.00**	96.16	98.87	95.66/97.60
8	93.76	95.39	**98.67**	93.83	96.25	97.36/97.84
9	99.68	99.65	**99.90**	99.46	96.22	99.26/99.21
OA(%)	95.63	97.82	98.83	96.25	97.65	98.81/**98.87**
AA(%)	96.41	98.06	**99.06**	96.97	97.34	98.23/98.27
$\kappa(\times 100)$	94.18	97.07	98.43	94.99	96.92	98.39/**98.48**

Table 6: classification accuracies (averaged over 5 runs) on Pavia Centre scene

No.	raw-CNN	abun-CNN	ABMHC	PPF	S3EResBoF	HybridSN(raw/abun)
1	99.89	99.83	99.95	99.15	**100.00**	**100.00**/99.99
2	95.32	97.29	97.61	97.96	99.14	**99.73**/99.49
3	96.73	96.81	**97.73**	97.37	91.69	94.20/95.34
4	98.74	99.78	**99.86**	99.27	99.81	96.48/97.54
5	98.62	98.58	99.17	98.79	98.82	99.24/**99.64**
6	98.36	98.47	98.90	98.95	**99.13**	98.39/98.43
7	96.40	97.44	98.82	94.36	**99.97**	99.23/99.44
8	99.66	99.80	99.84	99.90	99.87	99.97/**99.99**
9	99.87	99.86	**99.96**	**99.96**	91.63	95.92/97.79
OA(%)	99.19	99.38	99.60	99.03	99.45	99.55/**99.65**
AA(%)	98.18	98.65	**99.09**	98.41	97.78	98.13/98.63
$\kappa(\times 100)$	98.85	99.12	99.43	98.62	99.22	99.35/**99.49**

Table 7: classification accuracies (averaged over 5 runs) on Salinas scene

No.	raw-CNN	abun-CNN	ABMHC	PPF	S3EResBoF	HybridSN(raw/abun)
1	99.76	99.96	**100.00**	99.98	99.00	99.98/**100.00**
2	99.17	99.80	99.72	99.58	**100.00**	99.88/99.93
3	98.41	99.64	99.79	99.61	**99.97**	99.76/99.76
4	99.16	99.35	99.69	**99.73**	99.68	98.67/99.44
5	97.75	99.11	99.56	97.43	**99.90**	99.83/99.43
6	99.58	99.70	99.96	99.66	**100.00**	99.92/**100.00**
7	99.63	99.76	99.82	**99.93**	99.87	99.67/**99.93**
8	81.68	88.93	94.21	84.81	**98.66**	96.67/97.15
9	97.15	99.04	99.68	99.15	**100.00**	99.64/99.94
10	93.24	97.93	98.16	96.73	**99.61**	99.00/98.83
11	99.59	99.31	99.60	99.45	**99.70**	98.58/99.43
12	99.81	99.97	**100.00**	**100.00**	99.99	99.88/99.93
13	99.61	**100.00**	99.97	99.50	**100.00**	**100.00**/99.94
14	99.63	99.98	**100.00**	99.47	99.90	99.59/99.84
15	84.26	91.66	**95.40**	81.80	95.05	92.22/90.07
16	99.20	99.18	99.27	98.81	**99.99**	99.07/99.53
OA(%)	92.73	96.03	97.85	93.61	**98.91**	97.91/97.73
AA(%)	96.73	98.33	99.05	97.23	**99.46**	98.90/98.95
$\kappa(\times 100)$	91.89	95.56	97.60	92.85	**98.78**	97.66/97.46

The results obtained by the proposed ABMHC, the ablation study methods, *i.e.*, raw-CNN and abun-CNN and three state-of-the-art methods on the aforementioned HSI datasets are listed in TABLES 5–8[4].

We observe the following facts from the ablation study. Firstly, the abun-CNN method always outperforms the raw-CNN method with a large margin in terms of all the metrics, on all the datasets. This indicates that compared with the raw spectral features, the abundance features extracted by autoencoder demonstrate improved discriminative ability, thus improving the performance of the classifier. Secondly, compared with the abun-CNN method, which employs the abundance representation from one single HSI, the multi-HSI based ABMHC always leads to better classification performances in all the metrics, on all the datasets. This demonstrates that enlarging the training set by merging the abundance information from different HSIs augments the classifier performance. A convincing explanation of this phenomenon is that the merged training set alleviates the overfitting issue on the CNN-based classifier by increasing the number of training samples.

In practice, different HSI scenes may have several classes in common, *e.g.*, Pavia University and Pavia Centre both have 9 classes, 7 of them being the same. However, such coincidence of classes in different datasets will not affect the effectiveness of the proposed method, as to be explained in following aspects. Firstly, even for the HSIs with mutual classes, the chance that the elements in abundance features from different HSIs correspond to the same endmembers, in the same order, is

[4]The results of PPF on grss_dfc_2018 are in italic, as the number of training samples used in this case is not the same as in other methods.

Table 8: classification accuracies (averaged over 5 runs) on grss_dfc_2018 scene

	raw-CNN	abun-CNN	ABMHC	PPF	S3EResBoF	HybridSN(raw/abun)
1	96.63	**96.83**	96.03	*87.18*	81.38	88.56/79.92
2	93.76	94.32	95.54	*94.04*	**96.79**	88.56/96.63
3	99.01	99.67	**99.86**	*98.92*	99.80	88.56/95.41
4	98.47	98.77	**98.78**	*84.94*	92.80	90.17/91.88
5	95.95	**97.16**	96.23	*55.80*	86.70	84.51/80.86
6	99.22	**99.81**	99.66	*93.81*	99.33	96.92/97.19
7	96.06	92.02	98.47	*99.50*	**99.19**	91.16/87.65
8	90.98	93.49	**94.72**	*68.71*	91.96	85.79/88.70
9	86.52	90.53	92.81	*98.26*	**99.72**	99.28/99.03
10	65.32	70.77	75.32	*71.75*	**91.42**	85.09/83.71
11	72.27	78.62	**82.89**	*59.13*	78.54	71.22/70.79
12	60.20	71.93	**77.83**	*39.42*	43.78	37.02/42.72
13	75.94	80.78	87.19	*78.71*	**95.12**	91.21/91.52
14	98.35	**98.78**	98.63	*55.89*	95.26	89.96/87.98
15	99.33	99.78	**99.88**	*93.24*	99.49	97.80/97.88
16	96.49	97.34	97.51	*85.49*	**98.66**	94.12/93.83
17	96.73	98.27	98.08	*98.13*	**100.00**	81.07/89.14
18	94.61	**96.20**	95.91	*37.25*	91.82	67.72/65.30
19	98.29	**99.23**	99.02	*61.27*	98.69	88.18/93.55
20	99.57	99.82	**99.85**	*73.22*	97.95	88.03/91.15
OA(%)	85.23	88.78	91.36	*83.47*	**95.14**	91.86/92.12
AA(%)	90.68	92.71	**94.21**	*76.73*	91.92	85.14/86.24
$\kappa(\times 100)$	80.60	85.10	88.79	*78.51*	**93.47**	89.13/89.45

Table 9: classification accuracies on Pavia University and Pavia Centre, with rearranged classes order

Pavia University				Pavia Centre			
Class	abun-CNN	ABMHC	Progress	Class	abun-CNN	ABMHC	Progress
Asphalt	96.45	97.99	1.54	Asphalt	96.81	97.73	0.92
Meadows	98.12	98.87	0.75	Meadows	99.80	99.84	0.04
Trees	98.58	98.86	0.28	Trees	97.29	97.61	0.32
Bare Soil	99.34	99.40	0.06	Bare Soil	99.86	99.96	0.1
Bitumen	98.69	99.00	0.31	Bitumen	98.58	99.17	0.59
SBB	95.39	98.67	3.28	SBB	99.78	99.86	0.08
Shadows	99.65	99.90	0.25	Shadows	97.44	98.82	1.38
PMS	99.90	99.98	0.08	*Water*	99.83	99.95	0.12
Gravel	96.46	98.87	2.41	*Tiles*	98.47	98.90	0.43

very slim. It is because the permutation of endmembers order in $\hat{\boldsymbol{M}}$ is not taken into account for each data. Secondly, the assemblage of abundance representations from different HSIs is only used to train the classifier, while the testing stage is performed on each specific dataset. As a result,

the testing stage will not be affected by the existence of the shared classes in different HSIs. To show the influence of the shared classes on the training stage, we have rearranged the results of Pavia University and Pavia Centre, two datasets with great coincidence of classes, in TABLE 9. Compared to the abun-CNN, the proposed fusion technique always leads to superior results on every class, whether it appears in both images or not. This demonstrates that the coincidence of classes has little negative effects on the performance.

The proposed ABMHC leads to comparable performance to the state-of-the-art algorithms. It outperforms the PPF method with a large margin on all the datasets. In comparison with the latest S3EResBoF and HybridSN, the proposed ABMHC generally provides comparable results. To be precise, our method slightly outperforms S3EResBoF and HybridSN on the Pavia University scene and the Pavia Centre scene. On the Salinas scene, the results of ABMHC fall between the results of the HybridSN and S3EResBoF. On grss_dfc_2018 dataset, the proposed method surpasses the S3EResBoF and HybridSN by a large margin in AA, while inferior to its counterparts in OA and κ. It is worth noting that the proposed ABMHC utilizes a simple CNN classifier, which is plain and shallow, while the S3EResBoF and the HybridSN employ far more complicated network structures [64, 65].

The effects of introducing abundance representation to the networks with complex structure are shown in the the last columns of TABLES 5–8, where the HybridSN is conducted using both the raw spectral data and the abundance representations. Compared to the raw-data-based counterpart, classification with the abundance representations leads to slight improvements on all the datasets. This demonstrates that even for the complex networks, the abundance strategy still works.

6 Conclusion

In this paper, we proposed an abundance-based multi-HSI classification method, to address the overfitting issue in deep learning-based classification. The original intention of the proposed method is two-fold: 1) The abundance features extracted by SU have more discriminative ability than the raw spectral features, which enables the use of simple networks to alleviate the overfitting issue; 2) Training a classifier with multiple HSIs will lead to superior performance than training with one single HSI, as enlarging the training set usually alleviates the overfitting issue. This idea becomes feasible by transforming multiple HSIs from the spectral domain to the abundance domain by SU.

From these two aspects, we first designed and trained autoencoder-based SU model for each HSI seperately. After that, the HSIs were mapped to the abundance domain by the learned autoencoders. Lastly, a compatible classifier was trained by the abundance features from multiple HSIs, and further applied to predict the labels on the testing sets. The ablation study and comparative experiments were performed on four datasets. The results in the ablation study confirmed the original intention. The comparative experiments showed that our method provided comparable classification performance to the state-of-the-art methods, but using a far more simplified model structure.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant 11801409, 61701337, and the Natural Science Foundation of Tianjin City under Grand 18JC-QNJC01600.

The authors would like to thank the National Center for Airborne Laser Mapping and the Hyperspectral Image Analysis Laboratory at the University of Houston for acquiring and providing the data used in this study, and the IEEE GRSS Image Analysis and Data Fusion Technical Committee.

References

[1] J. M. Bioucas-Dias, A. Plaza, N. Dobigeon, M. Parente, Q. Du, P. Gader, J. Chanussot, Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 5 (2) (2012) 354–379.

[2] C. I. Chang, Hyperspectral Imaging: Techniques for Spectral Detection and Classification, Plenum Publishing Co., 2003.

[3] L. Wang, C. Shi, C. Diao, W. Ji, D. Yin, A survey of methods incorporating spatial information in image classification and spectral unmixing, International Journal of Remote Sensing 37 (16) (2016) 3870–3910. doi:10.1080/01431161.2016.1204032.

[4] H. Fan, Y. Chen, Y. Guo, H. Zhang, G. Kuang, Hyperspectral image restoration using low-rank tensor recovery, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing PP (99) (2017) 1–16.

[5] N. M. Nasrabadi, Hyperspectral target detection : An overview of current and future challenges, IEEE Signal Processing Magazine 31 (1) (2014) 34–44. doi:10.1109/MSP.2013.2278992.

[6] J. Jiang, J. Ma, C. Chen, Z. Wang, Z. Cai, L. Wang, Superpca: A superpixelwise pca approach for unsupervised feature extraction of hyperspectral imagery, IEEE Transactions on Geoscience and Remote Sensing 56 (8) (2018) 4581–4593. doi:10.1109/TGRS.2018.2828029.

[7] A. Villa, J. A. Benediktsson, J. Chanussot, C. Jutten, Hyperspectral image classification with independent component discriminant analysis, IEEE Transactions on Geoscience and Remote Sensing 49 (12) (2011) 4865–4876. doi:10.1109/TGRS.2011.2153861.

[8] W. Li, S. Prasad, J. E. Fowler, L. M. Bruce, Locality-preserving dimensionality reduction and classification for hyperspectral image analysis, IEEE Transactions on Geoscience and Remote Sensing 50 (4) (2012) 1185–1198. doi:10.1109/TGRS.2011.2165957.

[9] F. Melgani, L. Bruzzone, Classification of hyperspectral remote sensing images with support vector machines, IEEE Transactions on Geoscience and Remote Sensing 42 (8) (2004) 1778–1790. doi:10.1109/TGRS.2004.831865.

[10] L. Fang, S. Li, X. Kang, J. A. Benediktsson, Spectral-spatial hyperspectral image classification via multiscale adaptive sparse representation, IEEE Transactions on Geoscience and Remote Sensing 52 (12) (2014) 7738–7749. doi:10.1109/TGRS.2014.2318058.

[11] R. Neware, A. Khan, Survey on classification techniques used in remote sensing for satellite images, in: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 1860–1863.

[12] I. Pachón, S. Ramírez, D. Fonseca, P. Lozano-Rivera, C. Ariza, M. Paula Mancipe, M. Villamizar, H. Castro, E. Cabrera, M. Teresa Becerra, Random forest data cube based algorithm for land cover classification: A colombian case, in: IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium, 2018, pp. 8651–8654.

[13] H. Huang, Y. Duan, H. He, G. Shi, Local linear spatial–spectral probabilistic distribution for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 58 (2) (2020) 1259–1272.

[14] C. Chang, Statistical detection theory approach to hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 57 (4) (2019) 2057–2074.

[15] M. Paoletti, J. Haut, J. Plaza, A. Plaza, Deep learning classifiers for hyperspectral imaging: A review, ISPRS Journal of Photogrammetry and Remote Sensing 158 (2019) 279 – 317. doi:10.1016/j.isprsjprs.2019.09.006.

[16] N. Keshava, J. F. Mustard, Spectral unmixing, IEEE Signal Processing Magazine 19 (1) (2002) 44–57. doi:10.1109/79.974727.

[17] I. Dopido, M. Zortea, A. Villa, A. Plaza, P. Gamba, Unmixing prior to supervised classification of remotely sensed hyperspectral images, IEEE Geoscience and Remote Sensing Letters 8 (4) (2011) 760–764.

[18] I. Dopido, A. Villa, A. Plaza, P. Gamba, A quantitative and comparative assessment of unmixing-based feature extraction techniques for hyperspectral image classification, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 5 (2) (2012) 421–435.

[19] E. Ibarrola-Ulzurrun, L. Drumetz, J. Marcello, C. Gonzalo-Martín, J. Chanussot, Hyperspectral classification through unmixing abundance maps addressing spectral variability, IEEE Transactions on Geoscience and Remote Sensing 57 (7) (2019) 4775–4788.

[20] A. Villa, J. Chanussot, J. A. Benediktsson, C. Jutten, Spectral unmixing for the classification of hyperspectral images at a finer spatial resolution, IEEE Journal of Selected Topics in Signal Processing 5 (3) (2011) 521–533. doi:10.1109/JSTSP.2010.2096798.

[21] B. Fang, Y. Bai, Y. Li, Combining spectral unmixing and 3d/2d dense networks with early-exiting strategy for hyperspectral image classification, Remote Sensing 12 (5) (2020). doi:10.3390/rs12050779.

[22] F. I. Alam, J. Zhou, L. Tong, A. W. Liew, Y. Gao, Combining unmixing and deep feature learning for hyperspectral image classification, in: 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 2017, pp. 1–8.

[23] I. Dópido, J. Li, P. Gamba, A. Plaza, A new hybrid strategy combining semisupervised classification and unmixing of hyperspectral data, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 7 (8) (2014) 3619–3629.

[24] J. Li, I. Dópido, P. Gamba, A. Plaza, Complementarity of discriminative classifiers and spectral unmixing techniques for the interpretation of hyperspectral images, IEEE Transactions on Geoscience and Remote Sensing 53 (5) (2015) 2899–2912.

[25] A. Samat, J. Li, S. Liu, P. Du, Z. Miao, J. Luo, Improved hyperspectral image classification by active learning using pre-designed mixed pixels, Pattern Recognition 51 (2016) 43 – 58.

[26] Y. Sun, X. Zhang, A. Plaza, J. Li, I. Dópido, Y. Liu, A new semi-supervised classification strategy combining active learning and spectral unmixing of hyperspectral data, in: B. Huang, S. López, Z. Wu, J. M. Nascimento, J. Li, V. V. Strotov (Eds.), High-Performance Computing in Geoscience and Remote Sensing VI, Vol. 10007, International Society for Optics and Photonics, SPIE, 2016, pp. 44 – 51.

[27] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (7553) (2015) 436–444.

[28] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.

[29] Y. Cai, X. Liu, Z. Cai, Bs-nets: An end-to-end framework for band selection of hyperspectral image, IEEE Transactions on Geoscience and Remote Sensing 58 (3) (2020) 1969–1984. doi:10.1109/TGRS.2019.2951433.

[30] S. K. Roy, S. Das, T. Song, B. Chanda, Darecnet-bs: Unsupervised dual-attention reconstruction network for hyperspectral band selection, IEEE Geoscience and Remote Sensing Letters (2020) 1–5doi:10.1109/LGRS.2020.3013235.

[31] M. Wang, M. Zhao, J. Chen, S. Rahardja, Nonlinear unmixing of hyperspectral data via deep autoencoder networks, IEEE Geoscience and Remote Sensing Letters 16 (9) (2019) 1467–1471. doi:10.1109/LGRS.2019.2900733.

[32] M. Zhao, M. Wang, J. Chen, S. Rahardja, Hyperspectral unmixing via deep autoencoder networks for a generalized linear-mixture/nonlinear-fluctuation model, arXiv:1904.13017.

[33] Y. Chen, Z. Lin, X. Zhao, G. Wang, Y. Gu, Deep learning-based classification of hyperspectral data, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 7 (6) (2014) 2094–2107. doi:10.1109/JSTARS.2014.2329330.

[34] X. Ma, H. Wang, J. Geng, Spectral–spatial classification of hyperspectral image based on deep auto-encoder, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 9 (9) (2016) 4073–4085. doi:10.1109/JSTARS.2016.2517204.

[35] C. Tao, H. Pan, Y. Li, Z. Zou, Unsupervised spectral–spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification, IEEE Geoscience and Remote Sensing Letters 12 (12) (2015) 2438–2442. doi:10.1109/LGRS.2015.2482520.

[36] P. Zhou, J. Han, G. Cheng, B. Zhang, Learning compact and discriminative stacked autoencoder for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 57 (7) (2019) 4823–4833. doi:10.1109/TGRS.2019.2893180.

[37] X. Zhang, Y. Liang, C. Li, N. Huyan, L. Jiao, H. Zhou, Recursive autoencoders-based unsupervised feature learning for hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 14 (11) (2017) 1928–1932. doi:10.1109/LGRS.2017.2737823.

[38] V. Slavkovikj, S. Verstockt, W. De Neve, S. Van Hoecke, R. Van de Walle, Hyperspectral image classification with convolutional neural networks, in: Proceedings of the ACM international conference on Multimedia, ACM, 2015, pp. 1159–1162.

[39] Y. Chen, H. Jiang, C. Li, X. Jia, P. Ghamisi, Deep feature extraction and classification of hyperspectral images based on convolutional neural networks, IEEE Transactions on Geoscience and Remote Sensing 54 (10) (2016) 6232–6251. doi:10.1109/TGRS.2016.2584107.

[40] L. Jiao, M. Liang, H. Chen, S. Yang, H. Liu, X. Cao, Deep fully convolutional network-based spatial distribution prediction for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 55 (10) (2017) 5585–5599. doi:10.1109/TGRS.2017.2710079.

[41] M. E. Paoletti, J. M. Haut, S. K. Roy, E. M. T. Hendrix, Rotation equivariant convolutional neural networks for hyperspectral image classification, IEEE Access 8 (2020) 179575–179591. doi:10.1109/ACCESS.2020.3027776.

[42] W. Yao, C. Lian, L. Bruzzone, Clustercnn: Clustering-based feature learning for hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters (2020) 1–5doi:10.1109/LGRS.2020.3010837.

[43] L. Mou, P. Ghamisi, X. X. Zhu, Deep recurrent neural networks for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 55 (7) (2017) 3639–3655.

[44] X. Zhang, Y. Sun, K. Jiang, C. Li, L. Jiao, H. Zhou, Spatial sequential recurrent neural network for hyperspectral image classification, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 11 (11) (2018) 4141–4155. doi:10.1109/JSTARS.2018.2844873.

[45] W. Li, G. Wu, F. Zhang, Q. Du, Hyperspectral image classification using deep pixel-pair features, IEEE Transactions on Geoscience and Remote Sensing 55 (2) (2017) 844–853. doi:10.1109/TGRS.2016.2616355.

[46] X. Zhao, Y. Liang, A. J. Guo, F. Zhu, Classification of small-scale hyperspectral images with multi-source deep transfer learning, Remote Sensing Letters 11 (4) (2020) 303–312. doi:10.1080/2150704X.2020.1714772.

[47] C. Xing, L. Ma, X. Yang, Stacked denoise autoencoder based feature extraction and classification for hyperspectral images, Journal of Sensors 2016 (2016).

[48] G. E. Hinton, D. van Camp, Keeping the neural networks simple by minimizing the description length of the weights, in: Proceedings of the Sixth Annual Conference on Computational Learning Theory, COLT '93, Association for Computing Machinery, New York, NY, USA, 1993, p. 5–13. doi:10.1145/168304.168306.

[49] N. Dobigeon, J. Tourneret, C. Richard, J. C. M. Bermudez, S. McLaughlin, A. O. Hero, Nonlinear unmixing of hyperspectral images: Models and algorithms, IEEE Signal Processing Magazine 31 (1) (2014) 82–94. doi:10.1109/MSP.2013.2279274.

[50] R. Guo, W. Wang, H. Qi, Hyperspectral image unmixing using autoencoder cascade, in: 2015 7th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2015, pp. 1–4. doi:10.1109/WHISPERS.2015.8075378.

[51] S. Ozkan, B. Kaya, G. B. Akar, Endnet: Sparse autoencoder network for endmember extraction and hyperspectral unmixing, IEEE Transactions on Geoscience and Remote Sensing 57 (1) (2019) 482–496. doi:10.1109/TGRS.2018.2856929.

[52] Y. Su, J. Li, A. Plaza, A. Marinoni, P. Gamba, S. Chakravortty, Daen: Deep autoencoder networks for hyperspectral unmixing, IEEE Transactions on Geoscience and Remote Sensing 57 (7) (2019) 4309–4321. doi:10.1109/TGRS.2018.2890633.

[53] F. Khajehrayeni, H. Ghassemian, Hyperspectral unmixing using deep convolutional autoencoders in a supervised scenario, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13 (2020) 567–576. doi:10.1109/JSTARS.2020.2966512.

[54] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, D. J. Inman, 1d convolutional neural networks and applications: A survey, arXiv preprint arXiv:1905.03554 (2019).

[55] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Y. Lechevallier, G. Saporta (Eds.), Proceedings of COMPSTAT'2010, Physica-Verlag HD, Heidelberg, 2010, pp. 177–186.

[56] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[57] S. R. Dubey, S. Chakraborty, S. K. Roy, S. Mukherjee, S. K. Singh, B. B. Chaudhuri, diffgrad: An optimization method for convolutional neural networks, IEEE Transactions on Neural Networks and Learning Systems (2019) 1–12doi:10.1109/TNNLS.2019.2955777.

[58] J. M. P. Nascimento, J. M. B. Dias, Vertex component analysis: a fast algorithm to unmix hyperspectral data, IEEE Transactions on Geoscience and Remote Sensing 43 (4) (2005) 898–910. doi:10.1109/TGRS.2005.844293.

[59] H. Jin, Q. Song, X. Hu, Auto-keras: An efficient neural architecture search system, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, 2019, pp. 1946–1956.

[60] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: a system for large-scale machine learning., in: OSDI, Vol. 16, 2016, pp. 265–283.

[61] Y. Xu, B. Du, L. Zhang, D. Cerra, M. Pato, E. Carmona, S. Prasad, N. Yokoya, R. Hänsch, B. Le Saux, Advanced multi-sensor optical remote sensing for urban land use and land cover classification: Outcome of the 2018 ieee grss data fusion contest, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12 (6) (2019) 1709–1724. doi:10.1109/JSTARS.2019.2911113.

[62] A. J. X. Guo, F. Zhu, Spectral-spatial feature extraction and classification by ann supervised with center loss in hyperspectral imagery, IEEE Transactions on Geoscience and Remote Sensing 57 (3) (2019) 1755–1767. doi:10.1109/TGRS.2018.2869004.

[63] X. Cao, X. Wang, D. Wang, J. Zhao, L. Jiao, Spectral–spatial hyperspectral image classification using cascaded markov random fields, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 12 (12) (2019) 4861–4872. doi:10.1109/JSTARS.2019.2938208.

[64] S. K. Roy, S. Chatterjee, S. Bhattacharyya, B. B. Chaudhuri, J. Platoš, Lightweight spectral-spatial squeeze-and-excitation residual bag-of-features learning for hyperspectral classification, IEEE Transactions on Geoscience and Remote Sensing (2020) 1–14doi:10.1109/TGRS.2019.2961681.

[65] S. K. Roy, G. Krishna, S. R. Dubey, B. B. Chaudhuri, Hybridsn: Exploring 3-d–2-d cnn feature hierarchy for hyperspectral image classification, IEEE Geoscience and Remote Sensing Letters 17 (2) (2020) 277–281.