Convolutional Neural Network Based on Bandwise-Independent Convolution and Hard Thresholding for Hyperspectral Band Selection

Jie Feng, *Member, IEEE*, Jiantong Chen, Qigong Sun, *Student Member, IEEE*, Ronghua Shang, *Member, IEEE*, Xianghai Cao, *Member, IEEE*, Xiangrong Zhang, *Senior Member, IEEE*, and Licheng Jiao, *Fellow, IEEE*

***Abstract*—Band selection has been widely utilized in hyperspectral image (HSI) classification to reduce the dimensionality of HSIs. Recently, deep-learning-based band selection has become of great interest. However, existing deep-learning-based methods usually implement band selection and classification in isolation, or evaluate selected spectral bands by training the deep network repeatedly, which may lead to the loss of discriminative bands and increased computational cost. In this article, a novel convolutional neural network (CNN) based on bandwise-independent convolution and hard thresholding (BHCNN) is proposed, which combines band selection, feature extraction, and classification into an end-to-end trainable network. In BHCNN, a band selection layer is constructed by designing bandwise 1×1 convolutions, which perform for each spectral band of input HSIs independently. Then, hard thresholding is utilized to constrain the weights of convolution kernels with unselected spectral bands to zero. In this case, these weights are difficult to update. To optimize these weights, the straight-through estimator (STE) is devised by approximating the gradient. Furthermore, a novel coarse-to-fine loss calculated by full and selected spectral bands is defined to improve the interpretability of STE. In the subsequent layers of BHCNN, multiscale 3-D dilated convolutions are constructed to extract joint spatial–spectral features from HSIs with selected spectral bands. The experimental results on several HSI datasets demonstrate that the proposed method uses selected spectral bands to achieve more encouraging classification performance than current state-of-the-art band selection methods.**

***Index Terms*—3-D dilated convolution, band selection, convolutional neural network (CNN), hard thresholding, hyperspectral image (HSI) classification, straight-through estimator (STE).**

I. Introduction

HYPERSPECTRAL remote sensing has an extremely important position in practical applications. Hundreds of spectral bands with continuous and narrow characteristics are obtained by hyperspectral remote sensing imaging spectrometers [1]. Therefore, compared with other remote sensing images, hyperspectral images (HSIs) have a better distinction for different land-cover classes. HSIs have been successfully applied in different fields, such as military [2], astronomy [3], agriculture [4], and mineralogy [5].

HSI classification has become one of the research hotspots [6]. In the HSI classification, high-resolution spectral bands provide more information for the identification of different land-cover classes. However, high correlation between spectral bands brings a certain degree of redundancy, which increases the computational burden and storage requirements. Furthermore, massive spectral bands may cause the "Hughes phenomenon." It means that the classification performance degrades when the number of spectral bands increases with the limited training samples. Therefore, dimensionality reduction is a crucial step of HSI classification.

The feature extraction and feature selection are two main dimensionality reduction methods [7]. In the feature extraction, new low-dimensional features are obtained by linearly or nonlinearly transforming the original data. Some representative algorithms include local Fisher's discriminant analysis [8], graph embedding [9], principal component analysis [10], and hypergraph discriminant analysis [11]. Feature selection eliminates some noisy, irrelevant, and redundant features through selecting the most discriminative or informative feature subset from original features. Although feature extraction often

Manuscript received October 10, 2019; revised January 31, 2020 and May 7, 2020; accepted June 4, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61871306, Grant 61772400, Grant 61773304, and Grant 61601397, in part by the Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2019JM-194, in part by the Joint Fund of the Equipment Research of Ministry of Education under Grant 6141A020337, in part by the Innovation Fund of Shanghai Aerospace Science and Technology under Grant SAST2019-093, and in part by the Aeronautical Science Fund of China under Grant 2019ZC081002. This article was recommended by Associate Editor P. P. Angelov. *(Corresponding author: Jie Feng.)*

The authors are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University, Xi'an 710071, China (e-mail: jiefeng0109@163.com; jiantongchen1123@163.com; xd_qigongsun@163.com; rhshang@mail.xidian.edu.cn; xianghaicao@hotmail.com; xrzhang@mail.xidian.edu.cn; lchjiao@mail.xidian.edu.cn).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2020.3000725

achieves better classification performance, extracted features are linear or nonlinear combination from original features and hard to interpret. Different from feature extraction, feature or band selection still keeps the physical meaning of original HSI data and has better interpretability for some specific applications. It is favored for HSI processing [12].

In the feature selection methods, filter, wrapper, and embedded are three main types [13]. In filter methods, the quality of selected spectral bands is measured by using certain predefined criteria, which are calculated only by the intrinsic properties of training samples. This type of method can be combined with various classifiers flexibly; thus, these methods have better generalization ability. Peng *et al.* [14] proposed a minimum-redundancy maximum-relevance (mRMR) method, which simultaneously maximizes the dependency of individual feature and minimizes the redundancy between any two features. In mRMR, the redundancy measure is independent of the corresponding classification task. To calculate the redundancy for classification, a new method based on the trivariate mutual information and clonal selection algorithm (TMI-CSA) was proposed for band selection of HSIs [15]. Both mRMR and TMI-CSA use information-based criteria [16], [17]. Furthermore, there are distance-based criteria [18], consistency-based criteria [19], manifold learning-based criteria [20], etc.

Wrapper methods evaluate the quality of the selected band subset through training and testing in the chosen classifier. In these methods, candidate band subsets are obtained first. Then, the classifier is trained iteratively based on each candidate band subset. In [21], nearest neighbor classifier is chosen as the evaluator of candidate band subsets and the genetic algorithm is devised to search in these band subsets. In [22], the fractional-order Darwinian particle swarm optimization (FODPSO) and support vector machine (SVM) are combined for feature selection. The classification result of SVM is considered as the fitness value of FODPSO. Compared with filter methods, wrapper methods often have better classification performance. However, each candidate band subset needs a well-trained classifier. Since the number of candidate band subsets is numerous, this type of wrapper method needs a large number of well-trained classifiers. Thus, these methods are time consuming.

In embedded methods, an optimal feature subset is searched in the process of classifier construction. In [23], an SVM-based feature selection method with recursive feature elimination (RFE-SVM) was proposed. In RFE-SVM, features are eliminated by using the weights calculated in the training of SVM. An improved version of RFE-SVM is modified recursive SVM (MR-SVM) [24]. Compared with RFE-SVM, MR-SVM integrates the mean values of features from different classes to eliminate the features. Later, a novel multiple kernel learning method based on discriminative kernel clustering (DKC-MKL) was proposed for band selection of HSIs [25]. DKC-MKL selects an appropriate band subset and implements classification by optimizing MKL with sparse constraint. In most cases, compared to the wrapper methods, the embedded methods are more efficient. Compared to the filter methods, the embedded methods generally achieve better classification performance.

In the last decade, deep neural-network methods were fascinating due to powerful feature representation and classification ability. Among these methods, convolutional neural network (CNN) with local receptive fields and shared weights has attracted more attention in image processing. Recently, CNN-based methods [26]–[28] have been successfully applied to band selection of HSIs. In [26] and [27], the quality of candidate band subsets is evaluated by CNN. In [26], FODPSO is used to search in candidate band subsets. For each candidate band subset, CNN needs to be trained to evaluate its classification performance. This method is abbreviated as SICNN, which uses the powerful classification ability of CNN effectively. However, SICNN is time consuming because of the repeated training of CNN with numerous parameters. In [27], a new CNN method based on distance density (DDCNN) was proposed for band selection. In DDCNN, candidate band subsets are selected by measuring the distance density among the spectral bands. Then, the classification performance of all the candidate band subsets is evaluated by using 1-D CNN trained beforehand with full-band HSI data. Compared with SICNN, DDCNN saves time by training CNN in advance. But the fixed trainable parameters in DDCNN may not always evaluate the classification performance of candidate band subsets effectively. Recently, an unsupervised band selection method was proposed by considering band selection as a CNN-based spectral reconstruction task, which is abbreviated as BS-Nets [28]. In BS-Nets, the band subset most capable of reconstructing original spectral bands is selected. Then, SVM is used as the final classifier. BS-Nets may lose discriminative bands because the feedback from SVM cannot influence the band selection.

In this article, a novel CNN method based on bandwise-independent convolution and hard thresholding (BHCNN) is proposed, which combines band selection, multiscale spatial–spectral feature extraction, and classification into a unified optimization procedure. In BHCNN, a new bandwise-independent convolution is designed in the band selection layer. Specifically, it performs a single 1×1 convolution for each spectral band of input HSIs independently. Then, hard thresholding is devised to discard some irrelevant, redundant or noisy bands by constraining the small kernel weights of bandwise-independent convolutions as zero. Then, HSI data with selected spectral bands are fed into subsequent layers. To fully utilize spatial and spectral information of HSI data, feature extraction layers with multiscale 3-D dilated convolutions are designed. Finally, multiscale spatial–spectral features are input into the classification layer. At the same time, an auxiliary classifier after the band selection layer is added to improve the discriminative ability of selected spectral bands. In the training of BHCNN, the weights of the band selection layer are difficult to be updated because the gradient of the weights corresponding to unselected bands is zero everywhere. To optimize these weights, straight-through estimator (STE) is designed by using the gradient approximation in BHCNN, which is abbreviated as BHCNN-STE. To improve the interpretability of STE, a new coarse-to-fine loss based on the real gradient is defined by introducing an additional full-band branch in the band selection layer. BHCNN with the coarse-to-fine loss is abbreviated as BHCNN-CFL. In

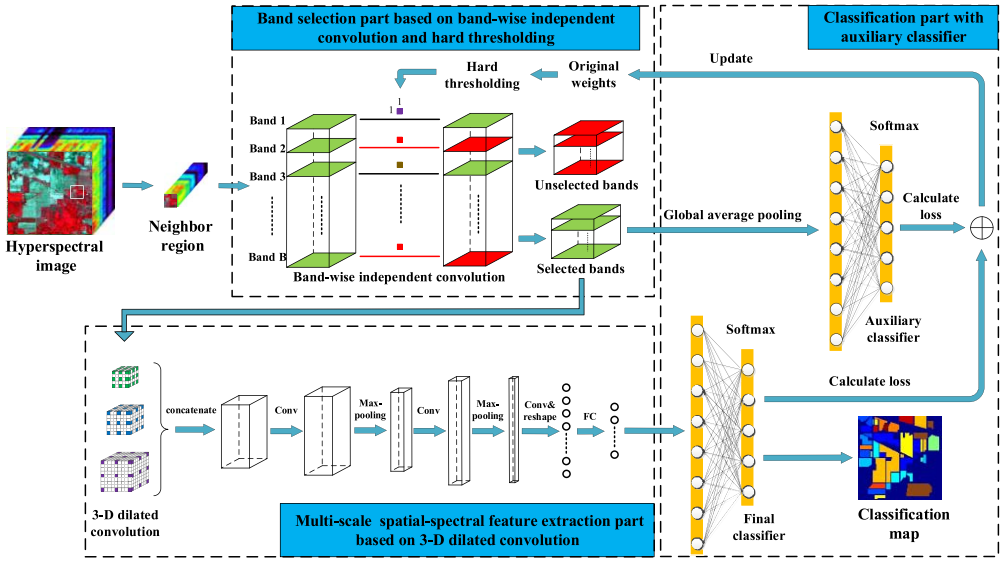


Fig. 1. Architecture of the proposed BHCNN method.

BHCNN-CFL, the coarse loss updates all the weights of the full-band branch and learns the classification model. The fine loss selects the spectral bands based on the weights from the full-band branch and fine-tunes the classification model. During the optimization process of BHCNN-CFL, the loss gradually shifts from the coarse loss to the fine loss by using an adaptive adjustment factor.

The main contributions of this article can be summarized as follows.

1) BHCNN-STE and BHCNN-CFL incorporate the band selection process into the training of CNN. These two methods not only overcome the problem of selecting less discriminative bands caused by the separation of band selection and classification but also alleviate the time-consuming problem caused by training the classification network repeatedly.
2) BHCNN-STE and BHCNN-CFL design the STE and define the coarse-to-fine loss to solve the difficulty that the band selection layer cannot be updated. These two methods can optimize the band selection, spatial–spectral feature extraction, and classification simultaneously. BHCNN-CFL utilizes the real gradient instead of the gradient approximation in BHCNN-STE, which has more physical meaning in the process of optimization and better interpretability.
3) 3-D dilated convolutions with various dilated rates are constructed to increase the receptive fields without introducing extra parameters, and capture multiscale spatial–spectral features simultaneously.
4) To further select more discriminative spectral bands, an auxiliary classification layer is connected after the band selection layer. Both the final and auxiliary classifiers are optimized to improve the performance of band selection and classification.

In the following sections of this article, the architecture of the proposed BHCNN is described in Section II. Section III introduces two optimization methods for BHCNN-based architecture, BHCNN-STE and BHCNN-CFL. In Section IV, the experimental results and analysis based on benchmark HSI datasets are presented. In Section V, some concluding observations and recommendations are provided for future work.

II. Architecture of BHCNN

The main architecture of the proposed BHCNN contains a band selection part based on bandwise-independent convolution and hard thresholding, a multiscale spatial–spectral feature extraction part based on 3-D dilated convolution, and a classification part with auxiliary classifier. It is shown in Fig. 1.

A. Band Selection Part Based on Bandwise-Independent Convolution and Hard Thresholding

HSIs have abundant spectral information with high spectral resolution, which can distinguish among similar spectral signatures. Unfortunately, a large number of spectral bands may contain some irrelevant, redundant, or noisy bands, which result in the deterioration of classification performance. To alleviate this problem, band selection methods only retain the most discriminative or informative bands from original spectral bands. Inspired by the depthwise separable convolution, a novel bandwise-independent convolution is constructed.

Fig. 2 illustrates the standard convolution, depthwise separable convolution, and bandwise-independent convolution. Each input channel is convolved with one specific kernel in standard convolution, and these channels are summed as the output. Depthwise separable convolution was first introduced in [29], which can be seen as a factorized convolution. Specifically, the standard convolution can be factorized into the channel-independent depthwise convolution and the spatial-independent pointwise (1×1) convolution. Due to effective reduction of parameters and computation, depthwise separable convolution has been applied in MobileNet V1 [30], MobileNet V2 [31], and Xception [32].

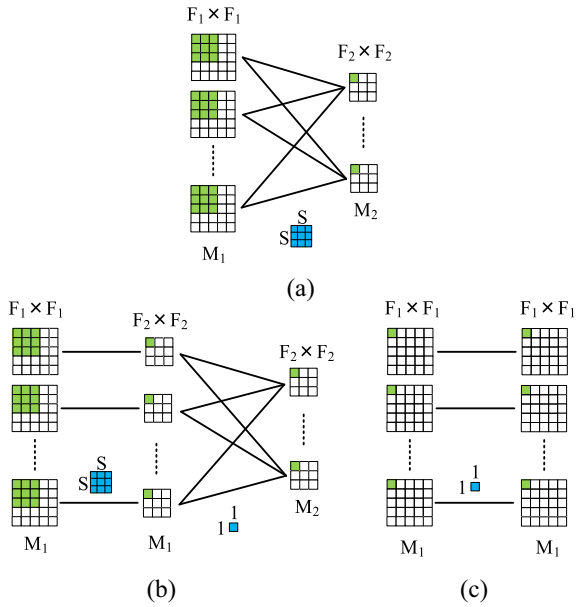


Fig. 2. Comparison of (a) standard convolution, (b) depthwise separable convolution, and (c) bandwise-independent convolution. F_1 and F_2 represent different feature map sizes, and M_1 and M_2 are the number of channels.

The bandwise-independent convolution inherits the characteristics of both depthwise and pointwise convolutions. Compared with the above two methods, bandwise-independent convolution uses a single 1×1 convolution for each input channel independently. For the band selection task, bandwise-independent convolution can retain the original spectral bands by implementing information isolation among the spectral bands. Meanwhile, the kernel weight of bandwise-independent convolution can represent the importance of the corresponding spectral band directly. Specifically, the larger absolute value the kernel weight has, the more important the corresponding spectral band is.

To discard some irrelevant, redundant, or noisy spectral bands, hard thresholding strategy [33] is devised. If the absolute value of the kernel weight is smaller than the threshold, the corresponding value is reset to zero. On the contrary, the corresponding value remains the same. Then, hard thresholding discards some of spectral bands by constraining the weights of the band selection layer. The spectral bands with zero weights are discarded, and only the spectral bands with nonzero weights are selected as the input of the next layer. In hard thresholding, the selection of an appropriate threshold is a critical issue. If the threshold is too large, little discriminative information is retained by selecting few spectral bands, which is not beneficial for classification. Conversely, some irrelevant, redundant, or noisy spectral bands may not be discarded. To find an appropriate threshold, the ranking-based strategy is used in the hard thresholding. In hard thresholding, the weights of all the spectral bands are ranked to determine the threshold. Specifically, the weights of all the bandwise-independent convolution kernels are sorted, and the threshold is determined by using the given number of selected spectral bands.

In HSIs, the training samples are represented by $X_{\text{train}} = \{x_1, x_2, \ldots, x_n\}$, where n is the number of training samples. $Y_{\text{train}} = \{y_1, y_2, \ldots, y_n\}$, where $y_i (1 \le i \le n)$ represents the class label corresponding to the training sample x_i. Each training sample $x_i \in R^{N \times N \times B}$, where $N \times N$ spatial neighborhood region is extracted to represent the current sample. B is the number of spectral bands. The number of layers in BHCNN is denoted as l. Fig. 3 shows the detailed architecture of the band selection layer. In this layer, $l = 1$ and the training sample x_i is used as the input. A single kernel is convolved to each input spectral band $x_{i,k}$, $(1 \le k \le B)$ and independent of other input spectral bands. For all the B input spectral bands, there are B kernels of size 1×1. $w_k^{(1)}$ and $b_k^{(1)}$ are the kth kernel weight and bias of the band selection layer. In this way, each weight of these kernels represents the importance of corresponding spectral band. Then, the hard thresholding $\phi(\cdot)$ is employed to constrain kernel weights and only retain the weights corresponding to selected spectral bands. $w_k^{(1)'} = \phi(w_k^{(1)})$. $f_i^{(1)'}$ represents the output calculated by $w_k^{(1)'}$ and x_i. The band selection layer with bandwise-independent convolution and hard thresholding is formulated as

$$\begin{aligned} f_i^{(1)'} &= \left\{ \phi\left(w_k^{(1)}\right) \odot x_{i,k} + b_k^{(1)} | k = 1, \ldots, B \right\} \\ &= \left\{ w_k^{(1)'} \odot x_{i,k} + b_k^{(1)} | k = 1, \ldots, B \right\} \\ &= \left\{ \left\{ \begin{array}{ll} w_k^{(1)} \odot x_{i,k} + b_k^{(1)}, & \left| w_k^{(1)} \right| > \varepsilon \\ b_k^{(1)}, & \text{else} \end{array} \right| k = 1, \ldots, B \right\} \end{aligned} \tag{1}$$

where $\odot$ represents the bandwise-independent convolution. In the hard thresholding, ε represents the threshold and $\varepsilon = \text{sort}(|w^{(1)}|, u)$. In detail, $\text{sort}(|w^{(1)}|, u)$ means that all the absolute values of kernel weights $\{w_1^{(1)}, w_2^{(1)}, \ldots, w_B^{(1)}\}$ are arranged in the descending order first. Then, the $(u+1)$th top value is used as the threshold. u is the number of selected spectral bands, which is given from the beginning by the users.

B. Spatial–Spectral Feature Extraction Part Based on Multiscale 3-D Dilated Convolutions

HSIs are 3-D data cubes, which have spatial and spectral information simultaneously. In HSIs, different imaging conditions may cause the changes of spectral characteristics with the same class. Therefore, joint spatial–spectral feature extraction is critical for HSI classification. After the band selection layer, feature extraction layers based on 3-D dilated convolutions with various dilated rates are applied, which capture multiscale spatial–spectral features simultaneously. As far as we know, this application has not appeared in HSI processing.

Common 2-D convolution has been frequently used in natural image processing. It is powerful in extracting local spatial information from natural images. Compared with 2-D convolution, 3-D convolution is more suitable for high-dimensional HSI classification. It can extract spectral information as much as possible while extracting spatial information. However, 3-D convolution instead of 2-D convolution in HSI classification results in the dramatic increase in the number of parameters and computational complexity. Dilated convolution [34] was proposed to exponentially enlarge receptive fields while keeping the resolution of the

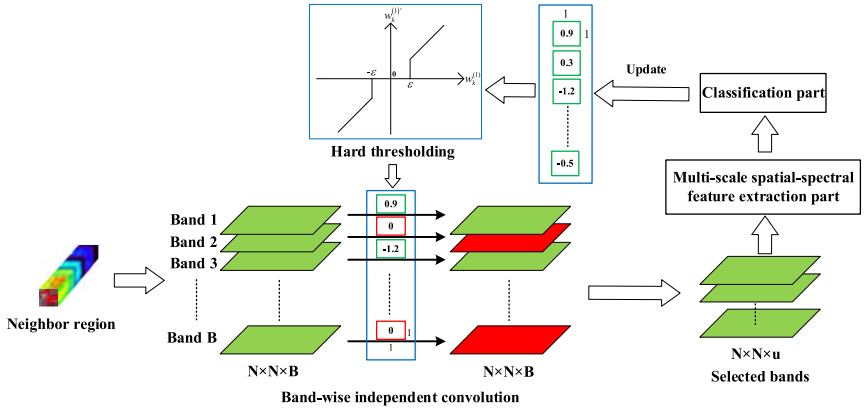


Fig. 3. Architecture of the band selection part.

feature maps. In semantic segmentation, dilated convolution has been successfully used. For BHCNN, dilated convolution can be used to increase receptive fields without increasing the learnable parameters.

By making full advantages of 3-D convolution and dilated convolution, 3-D dilated convolution is constructed to extract joint spatial–spectral features and reduce the computational cost. 3-D dilated convolution inflates 3-D kernel by inserting zeros between the kernel elements. The number of zeros is controlled by the dilated rate. 3-D dilated convolution with dilated rate γ introduces $\gamma - 1$ zeros into adjacent kernel elements, which enlarges the size of an $S \times S \times S$ kernel to $[S+(S-1)(\gamma-1)] \times [S+(S-1)(\gamma-1)] \times [S+(S-1)(\gamma-1)]$ effectively. Compared with 3-D convolution, 3-D dilated convolution reduces the computational complexity when the features with the same scale are extracted. The size of convolution kernels controls how much local contextual information can be utilized to a certain extent. To extract spatial–spectral contextual information with different scales, 3-D dilated convolutions with different dilated rates are designed. The feature maps from multiscale 3-D dilated convolutions are concatenated and passed through the subsequent layers. By adding 3-D dilated convolutions, the classification layer can more effectively evaluate the performance of selected spectral bands.

The architecture of feature extraction layers based on multiscale 3-D dilated convolutions is shown in Fig. 4. Three kinds of 3-D dilated convolutions with dilated rates of 1, 2, and 3 are connected after the band selection layer. Subsequently, three feature maps are concatenated together and further processed by the stacked 2-D convolution and max-pooling layers. Then, a 1-D vector is achieved by flattening the extracted feature maps, and processed by the full connection layer to produce an output. Finally, the output is used as the input of the classification part. To alleviate the overfitting problem, some strategies are used, such as dropout and batch normalization.

C. Classification Part With Auxiliary Classifier

In the classification part, a final softmax-based classification layer is connected at the end of BHCNN. The output of this layer represents the class probability distribution obtained from HSIs with selected bands. Inspired by Inception V1 [35], we

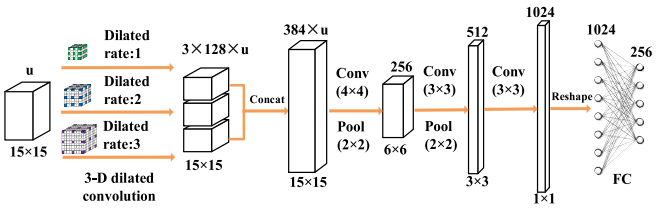


Fig. 4. Architecture of the multiscale spatial–spectral feature extraction part.

add an auxiliary classifier after the band selection layer, which can assist in the selection of more discriminative bands. The output of the band selection layer is flattened by global average pooling and then used as the input of this auxiliary classifier.

In BHCNN, the parameter θ is represented as $\theta = \{\theta^{(l)}\} = \{w^{(l)}, b^{(l)}\}, (1 \le l \le L)$. L indicates the total number of layers in BHCNN. After hard thresholding, the parameter θ' can be represented as $\theta' = \{\theta^{(1)'}, \theta^{(l)}\} = \{w^{(1)'}, b^{(1)}, w^{(l)}, b^{(l)}\}, (2 \le l \le L)$. Then, the loss function of BHCNN is defined by combining the losses of final and auxiliary classifiers $J_{\text{final}}(\theta')$ and $J_{\text{auxiliary}}(\theta')$, which is formulated as follows:

$$\begin{aligned} J(\theta') &= J_{\text{final}}(\theta') + \lambda \cdot J_{\text{auxiliary}}(\theta') \\ &= -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{c}\Big[I(j=y_i)\cdot\log\Big(\hat{y}_j^{\text{final}}(\theta';x_i)\Big) \\ &\qquad + \lambda \cdot I(j=y_i)\cdot\log\Big(\hat{y}_j^{\text{auxiliary}}(\theta';x_i)\Big)\Big] \end{aligned} \tag{2}$$

where $\hat{y}_j^{\text{final}}(\theta';x_i)$ and $\hat{y}_j^{\text{auxiliary}}(\theta';x_i)$ refer to the output of the jth class from final and auxiliary classifiers, respectively. $I(\cdot)$ is the indicator function. c represents the number of classes. λ controls the weight of the auxiliary classifier.

In the test stage, the class labels $Y_{\text{test}} = \{y_1^{\text{test}}, y_2^{\text{test}}, \ldots, y_m^{\text{test}}\}$ of test samples $X_{\text{test}} = \{x_1^{\text{test}}, x_2^{\text{test}}, \ldots, x_m^{\text{test}}\}$ are obtained by the final classifier, where m indicates the number of testing samples. $y_s^{\text{test}} (1 \le s \le m)$ represents the predicted label of the test sample x_s^{test}, which is formulated by the following equation:

$$y_s^{\text{test}} = \arg\max_j \frac{e^{\hat{y}_j^{\text{final}}(\theta';x_s^{\text{test}})}}{\sum_{z=1}^{c} e^{\hat{y}_z^{\text{final}}(\theta';x_s^{\text{test}})}}, j = 1, \ldots, c. \tag{3}$$

The generality of the deep learning network to different datasets is not as good as the unsupervised methods. It is possible to use transfer learning to alleviate this drawback to a certain extent. In the transfer learning process, the domain of large-scale dataset is usually defined as the source domain, and the domain of different small-scale datasets is usually defined as the target domain. The idea of transfer learning is to apply the knowledge learned from the source domain to the target domain, which improves the generality for the target task [36].

III. Optimization of BHCNN

In BHCNN, a unified loss incorporates the band selection process into the training of CNN. However, the weights of the band selection layer are difficult to be updated because some weights corresponding to the unselected bands are forced to be zero by hard thresholding. Specifically, hard thresholding consists of constant and identity functions. Since the gradient is zero everywhere in the constant function, the chain rule in standard backpropagation is inapplicable.

Although BHCNN uses $w^{(1)'}$ to compute the loss in the forward propagation, the original weights $w^{(1)}$ are used to be updated in the backpropagation. Since the changes of original weights $w^{(1)}$ are tiny during the backpropagation, the weights $w^{(1)'}$ after hard thresholding may ignore these changes, which would cause the loss to be difficult to update [37]. In BHCNN, $b^{(1)}$ and $\theta^{(l)}(2 \le l \le L)$ are optimized in the standard backpropagation way. However, $w^{(1)}$ is difficult to be optimized for the band selection layer. In this article, the optimization of $w^{(1)}$ is focused.

In the band selection layer, $\theta^{(1)'} = \{w^{(1)'}, b^{(1)}\} = \{\phi(w^{(1)}), b^{(1)}\}$. In the stage of forward propagation, the parameter $w^{(1)}$ is constrained as $w^{(1)'}$ by using the hard thresholding $\phi(\cdot)$. In the stage of backpropagation, the parameter $w^{(1)}$ is updated as follows:

$$\begin{aligned} w_{k,t}^{(1)} &= w_{k,t-1}^{(1)} + \eta \cdot \frac{\partial J(\theta')}{\partial w_k^{(1)}}\bigg|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \\ &= w_{k,t-1}^{(1)} + \eta \cdot \frac{\partial J(\theta')}{\partial f^{(1)'}} \cdot \frac{\partial f^{(1)'}}{\partial \phi\left(w_k^{(1)}\right)} \cdot \frac{\partial \phi\left(w_k^{(1)}\right)}{\partial w_k^{(1)}}\bigg|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \\ &= w_{k,t-1}^{(1)} - \eta \cdot \frac{1}{n} \\ &\quad \times \sum_{i=1}^{n}\left(\delta_{k,i}^{(1)}(\theta') * \mathrm{rot180}(x_{i,k}) \cdot \frac{\partial \phi\left(w_k^{(1)}\right)}{\partial w_k^{(1)}}\right)\bigg|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \end{aligned} \tag{4}$$

where $w_{k,t}^{(1)}$ represents the weight of the kth kernel of the band selection layer in the t-th iteration. η indicates the learning rate, which obeys the exponential decline during the iterative process. $\delta_{k,i}^{(1)}(\theta')$ is the error term of the kth kernel of the band selection layer. rot180$(\cdot)$ means that the matrix is rotated 180°. $*$ represents the standard convolution.

Using the definition of $\phi(w_k^{(1)})$, (4) can be further transformed into

$$\begin{aligned} w_{k,t}^{(1)} &= \begin{cases} w_{k,t-1}^{(1)} - \eta \frac{1}{n}\sum_{i=1}^{n}\left(\delta_{k,i}^{(1)}(\theta') * \mathrm{rot180}(x_{i,k}) \cdot 1\right)\Big|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \\ \qquad \left|w_k^{(1)}\right| > \varepsilon \\ w_{k,t-1}^{(1)} - \eta \frac{1}{n}\sum_{i=1}^{n}\left(\delta_{k,i}^{(1)}(\theta') * \mathrm{rot180}(x_{i,k}) \cdot 0\right)\Big|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \\ \qquad \text{else} \end{cases} \\ &= \begin{cases} w_{k,t-1}^{(1)} - \eta \frac{1}{n}\sum_{i=1}^{n}\left(\delta_{k,i}^{(1)}(\theta') * \mathrm{rot180}(x_{i,k})\right)\Big|_{w_k^{(1)}=w_{k,t-1}^{(1)}} \\ \qquad \left|w_k^{(1)}\right| > \varepsilon \\ w_{k,t-1}^{(1)}, \qquad \text{else.} \end{cases} \end{aligned} \tag{5}$$

As seen from (5), the weights corresponding to unselected bands are not updated during the training process, which is undesirable for the band selection task of HSIs.

To overcome the difficulty in training the band selection layer by using backpropagation, the STE and coarse-to-fine loss are devised in BHCNN, which are abbreviated as BHCNN-STE and BHCNN-CFL. BHCNN-STE estimates the approximate gradient of the hard thresholding. It directly takes the gradient of the identity function as the gradient of the hard thresholding during backpropagation. BHCNN-CFL defines a novel loss by adding an extra branch based on full spectral bands into the band selection layer. Compared with BHCNN-STE, BHCNN-CFL uses the real gradient instead of the gradient approximation and has better interpretability.

A. Optimization of BHCNN-STE

The idea of STE originates from Hinton in his lecture [38]. STE is used to optimize the neural network with binary activation. Later, Courbariaux *et al.* [39] proposed a novel BinaryConnect method. The derivative of identity function is treated as the proxy of the original derivative of the binary function. Binarized neural networks [40] and XNOR-Net [37] were proposed to deal with the binarization of weights and corresponding activations. Saturated STE was used to train the binarized neural networks and XNOR-Net, which substitutes the derivative of the sign function with $1_{\{|x|\le 1\}}$ in the backpropagation.

In BHCNN-STE, STE uses the gradient of the identity function instead of the gradient of hard thresholding for unselected spectral bands. It is formulated as follows:

$$\begin{aligned} \frac{\partial \phi\left(w_k^{(1)}\right)}{\partial w_k^{(1)}} &= \begin{cases} 1, & \left|w_k^{(1)}\right| > \varepsilon \\ 0, & \text{else} \end{cases} \\ &\overset{\text{STE}}{\approx} \begin{cases} 1, & \left|w_k^{(1)}\right| > \varepsilon \\ 1, & \text{else} \end{cases} \equiv 1. \end{aligned} \tag{6}$$

Then, substituting (6) into (4), the update of the weight $w_k^{(1)}$ under STE is rewritten as

$$w_{k,t}^{(1)} = w_{k,t-1}^{(1)} - \eta \frac{1}{n}\sum_{i=1}^{n}\left(\delta_{k,i}^{(1)}(\theta') * \mathrm{rot180}(x_{i,k})\right)\Big|_{w_k^{(1)}=w_{k,t-1}^{(1)}}. \tag{7}$$

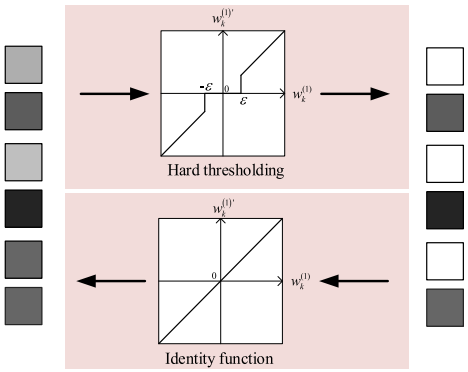


Fig. 5. Training of the band selection layer in BHCNN-STE.

The training of the band selection layer in BHCNN-STE is illustrated in Fig. 5. In the forward propagation of BHCNN-STE, the original weights are constrained by the hard thresholding. The derivative of the identity function is used instead of the derivative of the hard threshold function in the backpropagation.

Although STE can provide a way to update the weights corresponding to unselected bands, the update via STE is not well founded [41]. This is due to the discrepancy between the hard thresholding in the forward propagation and the identity function in the backpropagation. The update does not correspond to the gradient of the forward propagation. The gradient approximation in STE lacks interpretability.

B. Optimization of BHCNN-CFL

To improve the interpretability of STE, a new branch based on bandwise-independent convolution is added to the band selection layer of BHCNN. In the additional branch, the training samples with all the spectral bands are also used as the input. The original and additional branches differ in whether the weights corresponding to all the spectral bands are constrained by hard thresholding. In the additional branch, unconstrained weights are used. Then, these weights constrained by hard thresholding are used as the weights of the original branch. After additional and original branches of the band selection layer, full and selected spectral bands are used for subsequent layers, respectively.

According to these two branches, a novel coarse-to-fine loss is defined by combining the losses from additional and original branches. The coarse loss updates all the weights of additional branch and optimizes the classification model under full spectral bands. The fine loss is calculated by constraining the weights from the additional branch, which selects spectral bands and further fine-tunes the classification model. An adaptive adjustment factor is designed to balance the coarse and fine losses. During the optimization process of BHCNN-CFL, the loss gradually adjusts the focus from the coarse loss to the fine loss. This novel coarse-to-fine loss $J_{\mathrm{CFL}}(\theta)$ is defined as follows:

$$J_{\mathrm{CFL}}(\theta) = \sigma \cdot J(\theta) + (1-\sigma) \cdot J\left(\theta'\right) \tag{8}$$

where $J(\theta)$ is the loss related with the additional branch. The adjustment factor σ is calculated by

$$\sigma = 1 - \frac{t}{T} \tag{9}$$

where T indicates the total number of iterations.

In the beginning of iterations, the coarse-to-fine loss focuses on the coarse loss. In this case, BHCNN-CFL is inclined to update the weights corresponding to full spectral bands and learn a basic classification model. As the number of iterations t increases, σ reduces from 1 to 0. The focus of the coarse-to-fine loss shifts from the coarse loss to the fine loss. In this case, BHCNN-CFL is inclined to select the spectral bands based on the weights of additional branch and learn a more accurate classification model. According to (8), the derivative of $J_{\mathrm{CFL}}(\theta)$ to $w_k^{(1)}$ is calculated as follows:

$$\begin{aligned}
\frac{\partial J_{\mathrm{CFL}}(\theta)}{\partial w_k^{(1)}} &= \frac{\sigma \cdot \partial J(\theta) + (1-\sigma) \cdot \partial J\left(\theta'\right)}{\partial w_k^{(1)}} \\
&= \sigma \cdot \frac{\partial J(\theta)}{\partial f^{(1)}} \cdot \frac{\partial f^{(1)}}{\partial w_k^{(1)}} + (1-\sigma) \cdot \frac{\partial J\left(\theta'\right)}{\partial f^{(1)'}} \cdot \frac{\partial f^{(1)'}}{\partial \phi\left(w_k^{(1)}\right)} \\
&\quad \times \frac{d\phi\left(w_k^{(1)}\right)}{dw_k^{(1)}} \\
&= -\sigma \cdot \frac{1}{n} \sum_{i=1}^{n} \left(\delta_{k,i}^{(1)}(\theta) * \mathrm{rot180}\left(x_{i,k}\right)\right) \\
&\quad - (1-\sigma) \cdot \frac{1}{n} \sum_{i=1}^{n} \\
&\quad \times \left(\delta_{k,i}^{(1)}\left(\theta'\right) * \mathrm{rot180}\left(x_{i,k}\right) \cdot \frac{d\phi\left(w_k^{(1)}\right)}{dw_k^{(1)}}\right) \\
&= \begin{cases}
-\sigma \cdot \frac{1}{n} \sum_{i=1}^{n} \left(\delta_{k,i}^{(1)}(\theta) * \mathrm{rot180}\left(x_{i,k}\right)\right) \\
-(1-\sigma) \cdot \frac{1}{n} \sum_{i=1}^{n} \left(\delta_{k,i}^{(1)}\left(\theta'\right) * \mathrm{rot180}\left(x_{i,k}\right)\right) \\
\qquad \left|w_k^{(1)}\right| > \varepsilon \\
-\sigma \cdot \frac{1}{n} \sum_{i=1}^{n} \left(\delta_{k,i}^{(1)}(\theta) * \mathrm{rot180}\left(x_{i,k}\right)\right), \quad \text{else.}
\end{cases}
\end{aligned} \tag{10}$$

Substituting (10) into (4), the update of $w_k^{(1)}$ based on the coarse-to-fine loss is formulated as follows:

$$\begin{aligned}
w_{k,t}^{(1)} &= w_{k,t-1}^{(1)} + \eta \frac{\partial J_{\mathrm{CFL}}(\theta)}{\partial w_k^{(1)}}\bigg|_{w_k^{(1)} = w_{k,t-1}^{(1)}} \\
&= \begin{cases}
w_{k,t-1}^{(1)} - \eta \cdot \Big[\sigma \cdot \frac{1}{n} \sum_{i=1}^{n} \left(\delta_{k,i}^{(1)}(\theta) * \mathrm{rot180}\left(x_{i,k}\right)\right) \\
\qquad + (1-\sigma) \cdot \frac{1}{n} \sum_{i=1}^{n} \\
\qquad \left(\delta_{k,i}^{(1)}\left(\theta'\right) * \mathrm{rot180}\left(x_{i,k}\right)\right)\Big]\Big|_{w_k^{(1)} = w_{k,t-1}^{(1)}} \\
\qquad \left|w_k^{(1)}\right| > \varepsilon \\
w_{k,t-1}^{(1)} - \eta \cdot \sigma \cdot \frac{1}{n} \sum_{i=1}^{n} \\
\qquad \left(\delta_{k,i}^{(1)}(\theta) * \mathrm{rot180}\left(x_{i,k}\right)\right)\Big|_{w_k^{(1)} = w_{k,t-1}^{(1)}}, \quad \text{else.}
\end{cases}
\end{aligned} \tag{11}$$

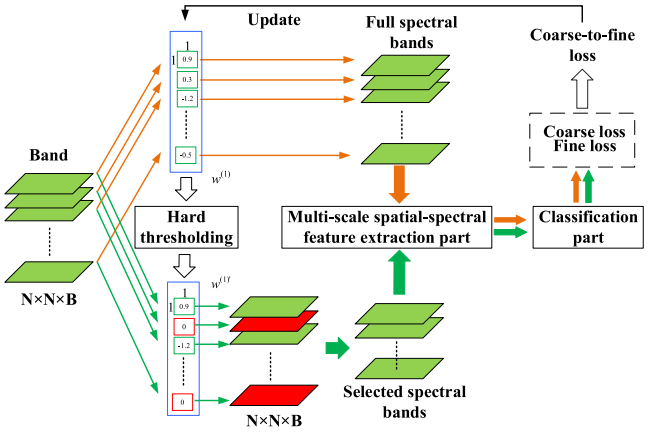


Fig. 6. Forward and backpropagations of BHCNN-CFL.

Fig. 6 shows the forward and backpropagations of BHCNN-CFL. In the backpropagation of BHCNN-CFL, the coarse and fine losses are used together to update the unconstrained weights.

Compared with BHCNN-STE, BHCNN-CFL updates the weights corresponding to unselected spectral bands by adding additional branch of full spectral bands. Therefore, the real gradient is used in the optimization in BHCNN-CFL instead of the gradient approximation in BHCNN-STE, which makes optimization physically meaningful and has better interpretability. The coarse-to-fine loss combines the losses calculated by full spectral bands and selected spectral bands. It enables BHCNN-CFL dynamically shift from the update of all the weights and the training of basic classification model to the selection of spectral bands and the fine-tuning of classification model. Compared with BHCNN-STE, BHCNN-CFL has better band selection capabilities and more powerful classification performance. The detailed procedure of BHCNN-CFL is summarized in Table I.

The computational complexity of BHCNN-CFL is measured by the number of learnable parameters and multiply-accumulate (MAC) operations [42]. The corresponding results of different types of learnable layers are recorded in Table II. In Table II, $S^{\bullet}$ represents the size of the convolution kernel. $F^{\bullet}_{\text{out}} \times F^{\bullet}_{\text{out}}$ indicates the spatial size of the output feature map. The number of input and output channels is represented as $M^{\bullet}_{\text{in}}$ and $M^{\bullet}_{\text{out}}$, respectively. In the 3-D dilated convolution layer, γ represents the dilated rate. T is the size of the output feature map along the spectral dimension in each channel. In Table II, the last line lists the results of BHCNN-CFL. α is the scale number of multiscale 3-D dilated convolution. l_c and l_f represent the number of convolution layers and full connection layers, respectively.

IV. Experimental Results and Analysis

In this section, three benchmark HSI datasets are used to validate the proposed BHCNN-STE and BHCNN-CFL methods. The performance of BHCNN-STE and BHCNN-CFL is investigated from the following aspects: classification results of HSI datasets, sensitivity to the number of training samples, running time, sensitivity to the number of selected spectral bands, and so on.

TABLE I
Procedure of BHCNN-CFL

INPUT: HSI dataset $\{X_{train}, Y_{train}\}$ and $\{X_{test}\}$, the number of iterations T, the number of selected bands u, the size of the neighbor region $N \times N$, the weight of the auxiliary classifier λ

Begin

Initialization

1. Normalize HSI data by MinMaxScaler method.
2. Parameter initialization: θ~N$(0, 0.1)$, N means Gaussian distribution.

Training stage

3. **for** every iteration t:
4. Update learning rate η
5. Update coefficient σ: $\sigma \leftarrow 1 - \frac{t}{T}$
6. Constraint the original weight of band-wise independent convolution:

$$w_k^{(1)'} \leftarrow \begin{cases} w_k^{(1)}, & \left|w_k^{(1)}\right| > \varepsilon \\ 0, & else \end{cases}$$

7. Calculate the coarse-to-fine loss $J_{CFL}(\theta)$ by (8).
8. Update parameter: $b^{(1)}, w^{(l)}, b^{(l)} \leftarrow J_{CFL}(\theta)$, $2 \le l \le L$
9. Update the weight $w^{(1)}$ of band-wise independent convolution by (11)
10. **end for**

Testing stage

11. Predict the label Y_{test} by (3).

END

OUTPUT: the selected band subset and the labels of the test samples Y_{test}

A. Data Description

In this article, we adopt three HSI datasets for the experiments: 1) the Indian Pines dataset; 2) the Pavia University dataset; and 3) the University of Houston dataset. Fig. S-1 in the supplementary material shows the false-color composite image.

1) The Indian Pines dataset was collected over the Indian Pines test site by the airborne visible/infrared imaging spectrometer sensor (AVIRIS) in June 1992. It consists of 145 × 145 pixels and 220 spectral bands. There are some absorption bands [100-104], [150-163], and 220, which are removed in the experiment. Sixteen vegetation classes are included in this dataset. The bands 50, 27, and 17 are used to show the false-color composite image in Fig. S-1(a) in the supplementary material.
2) The Pavia University dataset was collected by the reflection optical system imaging spectrometer (ROSIS) over Pavia, northern Italy. It has 610 × 340 pixels and 115 spectral bands. In the experiments, 103 bands are retained by removing 12 noisy bands. The bands 53, 31, and 8 are used to show the false-color composite image in Fig. S-1(b) in the supplementary material.
3) The University of Houston dataset was collected by the NSF-funded Center for Airborne Laser Mapping using compact airborne spectrographic imager (CASI-1500) on June 23, 2012. It covers the University of

TABLE II
NUMBERS OF LEARNABLE PARAMETERS AND MAC OPERATIONS OF DIFFERENT TYPES OF LEARNABLE LAYERS IN BHCNN-CFL

	Learnable parameters	**MAC operations**
Band-wise independent convolution	$\left(S^{BIC}\times S^{BIC}+1\right)\times M_{in}^{BIC}$	$S^{BIC}\times S^{BIC}\times M_{in}^{BIC}\times F_{out}^{BIC}\times F_{out}^{BIC}$
3D dilated convolution	$\left[M_{in}^{DC}\times\left(\frac{S^{DC}-1}{\gamma}+1\right)^{3}+1\right]\times M_{out}^{DC}$	$M_{in}^{DC}\times\left(\frac{S^{DC}-1}{\gamma}+1\right)^{3}\times M_{out}^{DC}\times F_{out}^{DC}\times F_{out}^{DC}\times T$
2D Convolution	$\left[M_{in}^{conv}\times\left(S^{conv}\times S^{conv}\right)+1\right]\times M_{out}^{conv}$	$M_{in}^{conv}\times M_{out}^{conv}\times S^{conv}\times S^{conv}\times F_{out}^{conv}\times F_{out}^{conv}$
Full connection	$\left(M_{in}^{FC}+1\right)\times M_{out}^{FC}$	$M_{in}^{FC}\times M_{out}^{FC}$
BHCNN-CFL	$\left(S^{BIC}\times S^{BIC}+1\right)\times M_{in}^{BIC}$ $+\sum_{i=1}^{\alpha}\left\{\left[M_{in}^{DC}\times\left(\frac{S_i^{DC}-1}{\gamma_i}+1\right)^{3}+1\right]\times M_{out}^{DC}\right\}$ $+\sum_{i=1}^{l_c}\left\{\left[M_{in_i}^{conv}\times\left(S_i^{conv}\times S_i^{conv}\right)+1\right]\times M_{out_i}^{conv}\right\}$ $+\sum_{i=1}^{l_f}\left[\left(M_{in_i}^{FC}+1\right)\times M_{out_i}^{FC}\right]$	$S^{BIC}\times S^{BIC}\times M_{in}^{BIC}\times F_{out}^{BIC}\times F_{out}^{BIC}$ $+\sum_{i=1}^{\alpha}\left[M_{in}^{DC}\times\left(\frac{S_i^{DC}-1}{\gamma_i}+1\right)^{3}\times M_{out}^{DC}\times F_{out}^{DC}\times F_{out}^{DC}\times T_i\right]$ $+\sum_{i=1}^{l_c}\left[M_{in_i}^{conv}\times M_{out_i}^{conv}\times S_i^{conv}\times S_i^{conv}\times F_{out_i}^{conv}\times F_{out_i}^{conv}\right]$ $+\sum_{i=1}^{l_f}\left(M_{in_i}^{FC}\times M_{out_i}^{FC}\right)$

Houston campus and its adjacent downtown area. The dataset comprises 1905×349 pixels, 144 spectral bands, and 15 classes. The bands 28, 45, and 65 are used to show the false-color composite image in Fig. S-1(c) in the supplementary material. The dataset is obtained from the IEEE GRSS Fusion Contest [43].

B. Experimental Setting

In this article, five representative band selection methods, mRMR [14], TMI-CSA [15], MR-SVM [24], SICNN [26], and DDCNN [27] are used as the comparison methods to verify the effectiveness of the proposed BHCNN-STE and BHCNN-CFL methods. Furthermore, SVM with the radial basis function (RBF-SVM) [44] is also used for comparison. Three popular indices are used to measure the classification performance: 1) overall accuracy (OA); 2) average accuracy (AA); and 3) the Kappa coefficient. The Kappa coefficient is a statistical measure of the degree of agreement, which is calculated by considering all the elements in the confusion matrix. It is defined by Kappa $= [m\sum_{i=1}^{c}p_{ii}-\sum_{i=1}^{c}(\sum_{j=1}^{c}p_{ij}\cdot\sum_{i=1}^{c}p_{ij})]/(m^2-\sum_{j=1}^{c}p_{ij}\cdot\sum_{i=1}^{c}p_{ij})$, where p_{ij} represents the number of samples of the jth class classified as the ith class. The experiments are run 30 times independently for all the methods. In the experiment, Python language and tensorflow library are used. GPU computation is implemented by an NVIDIA 1080Ti graphics card.

In RBF-SVM, multiclass classification is achieved by using the one-against-all strategy, and five-fold cross-validation is used to determine the penalty and gamma parameters. For mRMR, the Parzen window is used to evaluate the mutual information, and SVM is chosen as the classifier. For TMI-CSA [15], the clone scale factor and the mutation probability factor are determined by searching in the range of [500, 5000] and [1000, 10 000]. The maximum number of generations is set as 100, and the population size is set as 50. For MR-SVM, the grid search is used to determine the parameters of the classifier. For SICNN, the parameters in FODPSO are suggested in [26]. Moreover, the input HSI dataset is normalized in $[-0.5, 0.5]$, and the size of the input spatial window is 27×27. For DDCNN, the accuracy threshold of selected bands is set as 0.8. The number of training epochs and learning rate are set to the default values in [27]. For BHCNN-STE and BHCNN-CFL, the structure and parameters of the main

TABLE III
MAIN STRUCTURE AND PARAMETERS OF BHCNN-STE AND BHCNN-CFL

Operation	**Kernel**	**Padding**	**Stride**	**Activation**
Band-wise independent convolution	1×1×B	-	1	-
3D dilated convolution	3×3×3×128 (dilated rate:1)	-	1	ReLU
	3×3×3×128 (dilated rate:2)	-	1	ReLU
	3×3×3×128 (dilated rate:3)	-	1	ReLU
2D Convolution	4×4×256	-	1	ReLU
Max-pooling	2×2	-	2	-
2D Convolution	3×3×512	1	1	ReLU
Max-pooling	2×2	-	2	-
2D Convolution	3×3×1024	1	1	ReLU
Full connection	1024×512	-	-	ReLU
Global average pooling	15×15	-	1	-
Full connection	μ ×c	-	-	Softmax
Full connection	512×c	-	-	Softmax

TABLE IV
CLASSIFICATION RESULTS OF RBF-SVM, MRMR, TMI-CSA, MR-SVM, SICNN, DDCNN, BHCNN-STE, AND BHCNN-CFL ON THE INDIAN PINES DATASET

Class	RBF-SVM	mRMR	TMI-CSA	MR-SVM	SICNN	DDCNN	BHCNN-STE	BHCNN-CFL
1	55.8±14.1	29.5±5.8	41.7±7.8	67.4±6.3	83.4±7.0	75.0±6.9	59.1±14.8	68.2±13.2
2	74.6±1.7	69.0±2.8	69.5±3.7	70.5±4.3	81.9±4.9	84.8±5.3	92.6±3.2	98.1±1.5
3	63.4±4.4	60.8±4.5	50.9±5.5	54.0±2.1	82.8±3.2	80.7±2.4	99.8±0.1	99.9±0.1
4	42.1±5.1	34.7±2.8	31.8±3.7	23.8±0.8	85.4±2.3	85.3±1.1	96.0±0.9	98.2±1.8
5	88.7±2.7	84.6±4.8	84.4±3.9	88.8±4.6	79.9±8.4	85.8±1.4	95.0±1.1	95.2±0.9
6	95.6±2.2	94.3±3.6	96.1±1.3	93.7±0.8	96.3±2.0	92.6±2.6	99.1±0.2	98.7±1.0
7	70.8±11.1	40.7±2.8	73.1±4.6	65.4±7.9	82.4±3.9	40.7±7.3	70.4±7.8	74.1±12.0
8	97.4±1.0	93.9±3.6	98.9±0.1	96.4±0.4	86.0±9.8	94.0±3.9	100.0±0.0	100.0±0.0
9	24.2±13.7	21.1±6.4	15.8±1.9	10.5±1.7	42.2±9.8	42.1±9.3	100.0±0.0	99.7±0.3
10	72.1±3.3	69.0±4.0	62.1±3.4	64.9±2.7	78.4±3.7	49.5±3.1	95.7±2.1	98.0±1.3
11	79.1±2.3	72.7±2.9	84.3±3.1	84.5±2.6	94.1±3.4	84.7±1.6	97.2±0.7	98.1±0.9
12	60.9±3.8	52.7±7.1	41.7±2.0	59.8±3.1	78.3±6.8	70.8±3.2	92.5±2.1	87.6±4.1
13	97.2±1.5	95.3±2.6	93.8±0.2	95.3±0.1	82.3±8.0	98.9±7.6	98.5±0.3	99.6±0.4
14	91.4±2.7	93.1±1.8	95.0±0.7	95.5±0.4	97.7±1.4	98.0±1.3	99.2±0.2	99.7±0.2
15	43.3±6.3	43.0±3.5	54.3±1.0	48.8±8.3	83.6±3.3	76.8±4.6	86.9±2.0	92.1±0.4
16	89.2±5.0	85.2±6.4	87.4±2.9	89.7±9.1	88.3±6.3	90.9±4.5	93.2±1.9	97.7±0.2
OA (%)	77.5±0.4	73.5±0.7	75.3±1.9	76.8±1.0	87.6±1.0	82.7±1.0	96.1±0.7	97.4±0.9
AA (%)	71.6±1.6	65.0±2.1	67.6±3.7	69.3±2.4	82.7±2.6	78.2±2.1	92.2±1.3	94.1±1.6
Kappa(%)	74.3±0.5	69.8±0.8	71.5±2.4	73.3±1.7	85.9±1.1	80.4±1.5	95.6±1.0	97.0±1.4

network are shown in Table III. Batch-size and learning rate are selected by the suggestion in [45] and [46]. The other parameters are determined by a trial-and-error procedure. 20% of the training samples are used as the validation set for parameter tuning. The spatial window size of the input is 15×15. In the training process of BHCNN-STE and BHCNN-CFL, the learning rate adopts an exponential decline method to further reduce the loss. The initial learning rate is 0.8, and the learning rate per 100 iterations is multiplied by 0.05. The number of iterations is 1200, and the batch size is 128. The weight of the auxiliary classifier λ is set to 0.3.

C. Classification Results of HSI Datasets

1) The Indian Pines dataset is divided into 5% training set and 95% test set randomly. In all the band selection methods, 100 spectral bands are selected. The sensitivity to different numbers of selected spectral bands will be investigated in Section IV-E. Table S-I in the supplementary material shows the number of training and test samples for each class. Table IV records the average class-specific accuracy, OA, AA, and Kappa of all the methods over 30 runs. The best classification results are emphasized in gray regions. In Table IV, TMI-CSA is superior to mRMR due to the usage of an effective redundancy measure for classification. In mRMR and TMI-CSA, the criteria of band selection are independent of the chosen SVM classifier. Compared with them, MR-SVM, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL improve the classification performance. MR-SVM and DDCNN are inferior to SICNN. In MR-SVM, the selected bands are discarded in the current iteration, which cannot be re-evaluated and selected in the subsequent iterations. In DDCNN, 1DCNN is hard to extract abundant spatial information in the Indian Pines dataset. Compared with RBF-SVM, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL achieve better classification performance due to the effective band selection and hierarchical feature extraction. Compared with SICNN and DDCNN, BHCNN-STE improves at least 8.6% in terms of OA index by embedding the band selection into the training of CNN and extracting multiscale spatial–spectral features effectively. Compared with BHCNN-STE, BHCNN-CFL further improves the classification performance, which shows the superiority of the coarse-to-fine loss. Among the eight methods, BHCNN-CFL obtains the best classification performance.
The ground truth and visual classification maps of all the algorithms are shown in Fig. S-2 in the supplementary material. As shown in Fig. S-2(b)–(e) and S-2(g) in the supplementary material, there are massive noisy scattered points in RBF-SVM, mRMR, TMI-CSA, MR-SVM, and DDCNN, especially the soybean-clean, soybean-notill, and corn-notill classes. Compared with these methods, SICNN, BHCNN-STE, and BHCNN-CFL have better regional uniformity. Compared with SICNN, BHCNN-STE, and BHCNN-CFL better maintain the boundaries of different classes, such as the soybean-mintill, soybean-notill, and grass-trees classes.
2) The Pavia University dataset is divided into 3% training set and 97% test set randomly. In all the band selection methods, 60 spectral bands are selected. Table S-II in the supplementary material shows the number of training and test samples for each class. Table V records the average class-specific accuracy, OA, AA, and Kappa of all the methods.
As shown in Table V, compared with other methods, BHCNN-STE and BHCNN-CFL gain a certain degree of improvement in all the classes, especially in the bare soil and bitumen classes. For the bitumen class, BHCNN-STE and BHCNN-CFL have an increase of at least 14.7% and 15.4%. Compared with other algorithms, BHCNN-CFL improves by 11.7%, 13.6%, 12.6%, 12%, 4.4%, 9.4%, and 0.7% in terms of OA index. The ground truth and visual classification

TABLE V
CLASSIFICATION RESULTS OF RBF-SVM, MRMR, TMI-CSA, MR-SVM, SICNN, DDCNN, BHCNN-STE, AND BHCNN-CFL ON THE PAVIA UNIVERSITY DATASET

Class	RBF-SVM	mRMR	TMI-CSA	MR-SVM	SICNN	DDCNN	BHCNN-STE	BHCNN-CFL
1	88.5±1.0	94.7±2.3	91.8±2.2	83.2±2.3	93.0±1.9	99.2±0.3	98.4±0.7	99.6±0.2
2	97.5±0.2	97.9±0.7	98.7±0.1	94.4±0.1	99.0±0.4	97.6±0.9	99.7±0.1	99.8±0.2
3	65.5±3.1	34.3±1.8	61.0±0.9	89.1±2.0	96.1±2.5	74.2±2.5	94.7±1.1	98.6±0.9
4	90.0±1.6	94.5±0.7	88.2±0.3	94.2±0.9	94.8±2.6	85.5±2.7	97.8±0.9	98.9±0.4
5	99.3±0.1	99.5±0.1	99.8±0.1	96.3±0.1	99.2±0.2	82.9±0.4	100.0±0.0	100.0±0.0
6	61.1±1.1	41.9±0.7	51.4±1.0	78.0±5.5	87.2±1.8	66.5±0.6	99.4±0.2	100.0±0.0
7	76.6±4.5	84.4±8.3	60.0±6.9	30.4±9.3	78.1±1.7	69.8±4.3	99.1±0.5	99.8±0.1
8	85.5±2.6	89.0±2.3	85.0±3.1	83.9±2.4	96.6±2.0	93.6±0.1	98.3±1.0	98.8±1.0
9	95.9±2.1	87.4±0.2	97.8±0.2	98.5±0.4	93.6±4.1	86.2±1.8	99.6±0.2	99.6±0.3
OA (%)	88.0±0.5	86.1±0.4	87.1±0.2	87.7±0.6	95.3±0.9	90.3±0.7	99.0±0.2	99.7±0.2
AA (%)	84.4±0.6	80.4±1.2	81.5±0.6	83.1±1.5	93.1±0.7	83.9±0.7	98.6±0.3	99.4±0.2
Kappa(%)	83.9±0.6	81.0±0.8	82.4±0.5	83.8±1.0	93.7±0.7	87.1±0.8	98.7±0.2	99.4±0.3

TABLE VI
CLASSIFICATION RESULTS OF RBF-SVM, MRMR, TMI-CSA, MR-SVM, SICNN, DDCNN, BHCNN-STE, AND BHCNN-CFL ON THE UNIVERSITY OF HOUSTON DATASET

Class	RBF-SVM	mRMR	TMI-CSA	MR-SVM	SICNN	DDCNN	BHCNN-STE	BHCNN-CFL
1	93.1±1.7	92.8±0.3	95.5±0.2	91.8±0.1	97.6±0.7	92.9±1.1	94.1±1.5	97.8±0.5
2	96.7±0.4	95.5±0.4	98.9±0.3	96.3±0.3	98.5±0.5	96.1±1.5	98.8±0.5	99.0±0.3
3	97.4±0.3	96.7±0.1	97.9±0.1	97.3±0.1	98.5±0.4	99.8±0.1	98.5±0.9	97.7±2.1
4	92.6±2.1	88.7±0.3	79.4±1.1	94.1±0.3	98.7±0.1	89.1±2.0	99.2±0.2	97.7±2.0
5	97.2±0.2	99.7±0.2	99.7±0.1	98.1±0.7	96.5±0.5	98.0±0.8	100.0±0.0	100.0±0.0
6	94.2±0.4	95.1±0.4	94.8±0.2	93.9±0.9	93.6±1.4	89.3±0.6	92.5±2.7	94.6±3.9
7	94.6±0.6	92.7±1.4	96.3±0.7	86.1±1.1	91.4±0.9	94.8±0.6	99.0±0.3	98.8±1.2
8	76.2±3.8	61.5±2.1	71.4±2.4	77.8±3.7	83.8±2.9	80.0±3.4	92.7±2.9	97.1±2.9
9	89.5±1.7	85.8±1.4	84.9±1.1	94.5±1.3	92.4±1.7	88.6±0.7	88.4±4.3	90.5±2.2
10	95.4±2.8	97.2±0.9	96.3±1.6	92.7±3.3	94.3±2.3	92.9±0.4	98.3±0.7	97.7±1.9
11	96.8±0.6	96.6±1.3	97.4±1.2	96.1±2.4	86.7±1.3	98.7±0.4	98.8±0.4	98.8±0.6
12	86.4±3.6	83.3±5.4	88.2±3.9	86.7±3.1	89.9±4.4	90.3±1.5	94.8±2.1	95.7±2.7
13	73.1±8.3	72.6±8.8	92.9±1.1	72.9±7.3	97.6±1.2	82.5±2.1	92.3±3.7	94.6±1.9
14	99.0±0.1	99.5±0.1	88.0±0.3	98.5±0.2	97.1±0.4	97.3±1.9	100.0±0.0	100.0±0.0
15	99.5±0.3	96.0±0.5	97.0±0.2	99.5±0.3	97.8±0.1	98.2±0.2	98.6±0.8	99. 5±0.2
OA (%)	92.1±0.7	89.9±0.6	90.2±1.1	91.7±0.8	93.8±0.7	92.5±0.9	96.5±0.5	97.4±0.4
AA (%)	92.1±1.2	90.2±1.4	89.2±1.5	91.7±1.3	92.7±1.7	92.6±1.1	96.4±0.6	97.2±0.5
Kappa(%)	91.5±0.8	89.1±1.1	89.4±1.3	91.0±0.9	93.4±0.9	91.9±1.1	96.2±0.5	97.1±0.3

maps of the Pavia University dataset are shown in Fig. S-3 in the supplementary material. As shown in Fig. S-3(b)–(e) and S-3(g) in the supplementary material, many samples belonging to the bare soil class are misclassified as the meadows class. SICNN, BHCNN-STE, and BHCNN-CFL provide a better distinction for these two classes by the effective usage of spatial and spectral information. Compared with other methods, BHCNN-CFL can classify samples in the near-edge regions more accurately and provide more similar results to the ground truth.

3) In the University of Houston dataset, 5% training set and 95% test set are divided randomly. In all the band selection methods, 80 spectral bands are selected. Table S-III in the supplementary material shows the number of training and test samples for each class. Table VI records the average class-specific accuracy, OA, AA, and Kappa of all the methods. It can be seen that compared with other methods, BHCNN-STE and BHCNN-CFL obviously improve the classification results in the commercial and parking_lot1 classes. For the commercial class, BHCNN-CFL improves by 20.9%, 35.6%, 25.7%, 19.3%, 13.3%, 17.1%, and 4.4%. For the soil and tennis_court classes, BHCNN-STE and BHCNN-CFL achieve completely correct classification results. Among all the eight methods, BHCNN-CFL achieves the best classification performance. The ground truth and visual classification maps of the University of Houston dataset are shown in Fig. S-4 in the supplementary material. RBF-SVM, mRMR, TMI-CSA, MR-SVM, and DDCNN exhibit noisy estimations in the classification maps, which are shown in Fig. S-4(b)–(e) and S-4(g) in the supplementary material. Compared with SICNN and BHCNN-STE, BHCNN-CFL keeps better boundary characteristics in the running_track class while performing better regional uniformity of the soil class.

D. Sensitivity to the Number of Training Samples

Fig. 7 records the classification results of all the methods with different percentages of training samples. Specifically, 1%, 3%, 5%, 7%, and 9% samples from each class on the Indian Pines dataset, 1%–5% on the Pavia University dataset, and 1%, 3%, 5%, 7%, and 9% on the University of Houston dataset are randomly selected as the training samples. CNN-based methods are usually heavily parameterized and sufficient training samples are required to guarantee the performance. When the training samples are sufficient, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL have obvious improvement over traditional methods. As the percentage of training samples decreases, the classification performance of all the methods degrades. In this case, SICNN and DDCNN have no

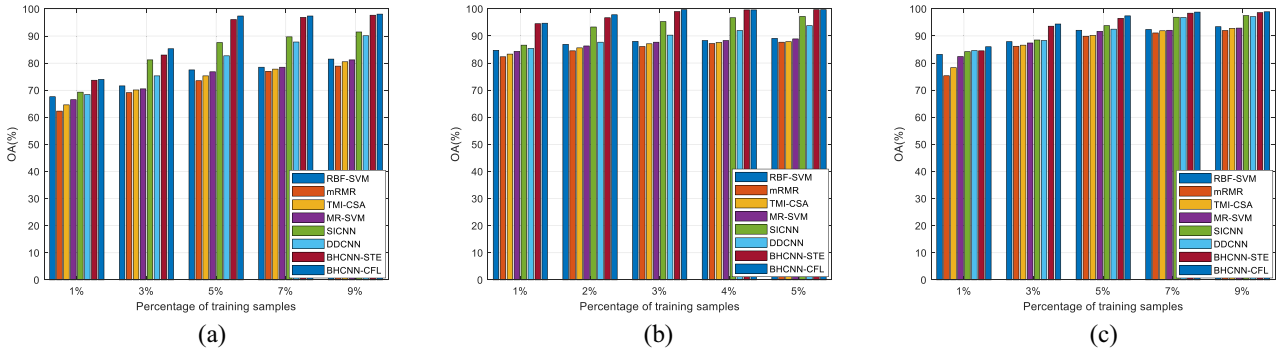


Fig. 7. OA results of RBF-SVM, mRMR, TMI-CSA, MR-SVM, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL with different percentages of training samples on the (a) Indian Pines, (b) Pavia University, and (c) University of Houston datasets.

obvious advantage over RBF-SVM. Compared with RBF-SVM, mRMR, TMI-CSA, MR-SVM, SICNN, and DDCNN, BHCNN-STE and BHCNN-CFL consistently achieve superior classification performance with different percentages of training samples. When the percentage of training samples exceeds 5% in the Indian Pines dataset, 1% in the Pavia University dataset, and 3% in the University of Houston dataset, BHCNN-STE and BHCNN-CFL achieve more than 90% in terms of OA index.

E. Classification Results With Fixed Number of Training Samples

The classification performance is analyzed with the fixed number of training samples per class for all the methods, as shown in Tables S-IV–S-VI in the supplementary material. In the Indian Pines dataset, 18 samples from each class are randomly selected as the training set. In the Pavia University and University of Houston datasets, 50 samples from each class are randomly selected as the training set. The remaining samples are used for testing.

As shown in Tables S-IV–S-VI in the supplementary material, the proposed methods achieve better classification results on the classes with fewer samples, such as the 1st, 7th, 13th, and 16th classes of the Indian Pines dataset, the 9th class of the Pavia University dataset, and the 6th class of the University of Houston dataset. At the same time, the proposed methods maintain a relatively good classification performance on most classes with a large number of samples. In addition, the proposed methods obtain the best classification performance in terms of OA, AA, and Kappa indices.

When the training and testing samples are divided proportionally, the classes with a large number of samples account for a large proportion of the overall loss. By optimizing the overall loss, the proposed methods prefer to improve the classification performance of these classes. When the fixed number of training samples is selected, this problem is alleviated.

F. Investigation on Running Time

The training and test time of different algorithms for three datasets are shown in Tables S-VII–S-IX in the supplementary material. As shown in Tables S-VII–S-IX in the supplementary material, mRMR costs less training time than TMI-CSA due to the fast incremental search strategy. Compared with mRMR and TMI-CSA, MR-SVM takes longer training time to optimize the classifier based on the current remaining band subset. Compared with RBF-SVM, mRMR, TMI-CSA, and MR-SVM cost less test time by reducing the dimensionality of original spectral bands. Compared with RBF-SVM, mRMR, and TMI-CSA, deep-learning-based methods, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL take longer training time due to heavily parameterized CNN models. SICNN is very time consuming since CNN needs to be trained repeatedly for each candidate band subset. Compared with SICNN, DDCNN takes less training time because it only trains the CNN model once. Compared with SICNN and MR-SVM, BHCNN-STE and BHCNN-CFL save the training time significantly because the band selection is embedded in the CNN training process. During the test phase, BHCNN-STE and BHCNN-CFL take a similar time as SICNN.

G. Sensitivity to the Number of Selected Spectral Bands

In this section, the classification performance of the eight algorithms with different numbers of selected bands is investigated, as shown in Fig. S-5 in the supplementary material. RBF-SVM uses all the original spectral bands and keeps stable with different numbers of selected spectral bands.

As shown in Fig. S-5 in the supplementary material, as the number of selected bands changes from 2 to 20 on three datasets, the OA curves of mRMR, TMI-CSA, MR-SVM, SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL rise sharply. This is because the increasing number of spectral bands provides complementary discriminative information for classification. As the number of selected bands changes from 20 to 100, 20 to 60, and 20 to 80 on three datasets, the OA curves of mRMR, TMI-CSA, MR-SVM, SICNN, and DDCNN still rise sharply. However, the rise of BHCNN-STE and BHCNN-CFL becomes slow. Compared with these methods, BHCNN-STE and BHCNN-CFL have potential to select more compact subset from original spectral bands. When the number of selected bands exceeds 100, 60, and 80 on three datasets, the OA curves of SICNN, DDCNN, BHCNN-STE, and BHCNN-CFL fluctuate or even drop. In this case, these methods may select some redundant or lowly discriminative bands, which hardly bring more discriminative

TABLE VII
OA RESULTS OF COMBINING ALL BAND SELECTION METHODS WITH KNN, SVM, ANN, AND THE PROPOSED NETWORK ON THE INDIAN PINES DATASET

Methods	KNN	SVM	ANN	Proposed Network
mRMR	60.1±0.5	70.7±0.7	76.0±1.1	89.6±0.9
TMI-CSA	60.9±0.9	71.2±1.9	76.3±1.1	89.9±0.7
MR-SVM	61.5±0.3	71.7±1.0	77.1±0.9	90.2±1.2
SICNN	64.8±0.8	73.1±0.7	78.4±1.1	92.7±0.5
DDCNN	62.1±1.0	72.0±1.2	77.5±0.7	90.9±1.4
BHCNN-STE	70.1±0.5	77.5±0.9	83.6±1.2	96.1±0.7
BHCNN-CFL	70.9±0.5	77.8±0.4	83.9±1.0	97.4±0.9

information for classification and even deteriorate the classification performance. When different numbers of spectral bands are selected, BHCNN-STE and BHCNN-CFL always obtain better classification results than other methods.

H. Analysis of the Selected Spectral Bands

Figs. S-6–S-8 in the supplementary material show the spectral bands selected by all the algorithms on the three datasets. Each class is represented by the mean of samples in this class. The selected bands are represented by the dotted lines. In Figs. S-6(a), S-7(a), and S-8(a) in the supplementary material, RBF-SVM uses all the bands for classification. For clarity, only 30 spectral bands per dataset are selected and displayed for all the band selection algorithms. As shown in Figs. S-6–S-8 in the supplementary material, the spectral bands overlapping in most classes are difficult to provide discriminative information for classification, such as bands [103–106] of the Indian Pines dataset and bands [118–122] of the University of Houston dataset. These lowly discriminative spectral bands are selected in mRMR and TMI-CSA due to the separation of band selection and classification. In MR-SVM and DDCNN, some adjacent bands are selected, such as [80-85] in the Indian Pines dataset, [35-50] in the Pavia University dataset, and [75-85] in the University of Houston dataset. Since the spectral bands are approximately continuous in HSIs, adjacent bands often have high correlation. Compared with other algorithms, the proposed BHCNN-STE and BHCNN-CFL methods select more dispersed bands and cover most spectrums with large intervals among different classes. Generally, the dispersed bands have more diversity and the large interval in different classes has high discrimination.

I. Effectiveness Analysis to Each Part in BHCNN-STE and BHCNN-CFL

In this section, some ablation experiments are added to verify the effectiveness of the proposed methods in the Indian Pines dataset. To verify the effectiveness of band selection, all the band selection methods are combined with conventional classifiers (KNN, SVM, and ANN) and the proposed feature extraction and classification network, as shown in Table VII. To verify the effectiveness of multiscale 3-D dilated convolutions and auxiliary classifier, the proposed methods without 3-D dilated convolutions (BHCNN-STE-W3DC and BHCNN-CFL-W3DC) and without auxiliary classifier (BHCNN-STE-WAC and BHCNN-CFL-WAC) are used for comparison, as shown in Table VIII.

TABLE VIII
CLASSIFICATION RESULTS OF BHCNN-STE, BHCNN-STE-WAC, BHCNN-STE-W3DC, BHCNN-CFL, BHCNN-CFL-WAC, AND BHCNN-CFL-W3DC ON THE INDIAN PINES DATASET

Methods	OA (%)	AA (%)	Kappa (%)
BHCNN-STE	96.1±0.7	92.2±1.3	95.6±1.0
BHCNN-STE-WAC	94.6±0.7	90.4±1.3	93.8±0.8
BHCNN-STE-W3DC	95.1±0.5	91.0±2.2	94.4±0.5
BHCNN-CFL	97.4±0.9	94.1±1.6	97.0±1.4
BHCNN-CFL-WAC	95.8±0.6	94.4±1.5	95.3±0.7
BHCNN-CFL-W3DC	96.5±0.7	93.2±2.2	96.0±0.8

As shown in Table VII, all the methods with SVM perform better than these methods with KNN. These methods with ANN further improve the classification performance. The proposed methods perform better than the comparison methods under the same classifier. It is shown that the proposed methods can select more discriminative spectral bands. At the same time, compared with original methods, mRMR, TMI-CSA, MR-SVM, SICNN, and DDCNN improve the classification performance by combining the proposed feature extraction and classification network. This indicates that the proposed feature extraction and classification network is beneficial for classification. Among all the methods, the proposed methods still achieve the best classification performance.

In Table VIII, compared with BHCNN-STE-WAC and BHCNN-CFL-WAC, BHCNN-STE and BHCNN-CFL improve by 1.5 % and 1.6% in terms of OA index, respectively. It is shown that adding an auxiliary classifier after the band selection layer is beneficial for classification, which better optimizes the band selection layer. Compared with BHCNN-STE-W3DC and BHCNN-CFL-W3DC, BHCNN-STE and BHCNN- CFL achieve better classification performance. This indicates that 3-D dilated convolutions with different dilated rates improve the classification performance by extracting multiscale spatial–spectral features.

J. Analysis of Convergence Processes in BHCNN-STE and BHCNN-CFL

In this section, the convergence processes of BHCNN-STE and BHCNN-CFL are shown in Fig. S-9 in the supplementary material. The horizontal axis represents the number of iterations. The vertical axis represents the training loss. As shown in Fig. S-9 in the supplementary material, BHCNN-CFL converges faster than BHCNN-STE during the first 300 iterations. This means that the optimization of band selection with the coarse-to-fine loss is better than that with STE at early iterations. After 500 iterations, BHCNN-STE and BHCNN-CFL gradually converge. As the number of iterations increases, the selected spectral bands tend to be fixed. BHCNN-CFL maintains a lower training loss than BHCNN-STE during the iterations. The reason is that BHCNN-CFL selects more discriminative spectral bands.

V. CONCLUSION

In this article, novel BHCNN-STE and BHCNN-CFL methods have been proposed for band selection of HSIs. To select

the discriminative band subset from original spectral bands, BHCNN-STE and BHCNN-CFL design the band selection layer based on the bandwise-independent convolution and hard thresholding. Furthermore, in order to extract multiscale spatial–spectral features for HSIs with selected spectral bands, 3-D dilated convolutions with various dilated rates are constructed. To update the weights of the band selection layer, two kinds of optimization methods based on STE and CFL are devised. Compared with existing band selection methods, BHCNN-STE and BHCNN-CFL embed the band selection into the training process of CNN and jointly optimize the band selection, multiscale spatial–spectral feature extraction, and classification. The experimental results on several real HSI datasets demonstrated that the proposed methods are superior to current state-of-the-art band selection methods in terms of quantitative metrics and visual qualities of classification maps. In the future, the fusion of band selection and more deep architectures will be our focus to further improve the band selection and classification performance of HSIs. In addition, the proposed band selection part will be plugged into other deep-learning-based tasks, such as object detection and optimized sampling and coding.

REFERENCES

[1] C. I. Chang, *Hyperspectral Data Exploitation: Theory and Applications*. Hoboken, NJ, USA: Wiley, 2007.

[2] I. Makki, R. Younes, C. Francis, T. Bianchi, and M. Zucchetti, "A survey of landmine detection using hyperspectral imaging," *ISPRS J. Photogramm. Remote Sens.*, vol. 124, pp. 40–53, Feb. 2017.

[3] A. J. Brown, M. R. Walter, and T. J. Cudahy, "Hyperspectral imaging spectroscopy of a Mars analogue environment at the North Pole Dome, Pilbara Craton, Western Australia," *Aust. J. Earth Sci.*, vol. 52, no. 3, pp. 353–364, Jun. 2005.

[4] C. McCann, K. S. Repasky, R. L. Lawrence, and S. Powell, "Multi-temporal mesoscale hyperspectral data of mixed agricultural and grassland regions for anomaly detection," *ISPRS J. Photogramm. Remote Sens.*, vol. 131, pp. 121–133, Sep. 2017.

[5] S. Schneider, R. J. Murphy, and A. Melkumyan, "Evaluating the performance of a new classifier—The GP-OAD: A comparison with existing methods for classifying rock type and mineralogy from hyperspectral imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 145–156, Dec. 2014.

[6] S. Jia, L. Shen, J. Zhu, and Q. Li, "A 3-D Gabor phase-based coding and matching framework for hyperspectral imagery classification," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1176–1188, Apr. 2018.

[7] X. Jia, B. Kuo, and M. M. Crawford, "Feature mining for hyperspectral image classification," *Proc. IEEE*, vol. 101, no. 3, pp. 676–697, Mar. 2013.

[8] W. Li, S. Prasad, J. E. Fowler, and L. M. Bruce, "Locality-preserving dimensionality reduction and classification for hyperspectral image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 4, pp. 1185–1198, Apr. 2012.

[9] 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.*, vol. 50, no. 6, pp. 2604–2616, Jun. 2020, doi: 10.1109/TCYB.2019.2905793.

[10] W. Di, L. Zhang, D. Zhang, and Q. Pan, "Studies on hyperspectral face recognition in visible spectrum with feature band selection," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 40, no. 6, pp. 1354–1361, Nov. 2010.

[11] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.

[12] W. Sun and Q. Du, "Hyperspectral band selection: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 118–139, Jun. 2019.

[13] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, "Feature selection via dependence maximization," *J. Mach. Learn. Res.*, vol. 13, pp. 1393–1434, May 2012.

[14] H. Peng, F. Long, 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, no. 8, pp. 1226–1238, Aug. 2005.

[15] J. Feng, L. C. Jiao, X. Zhang, and T. Sun, "Hyperspectral band selection based on trivariate mutual information and clonal selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 4092–4105, Jul. 2014.

[16] 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.

[17] L. Jiao, J. Feng, F. Liu, T. Sun, and X. Zhang, "Semisupervised affinity propagation based on normalized trivariable mutual information for hyperspectral band selection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2760–2773, Jun. 2015.

[18] L. Zhang, Y. Zhong, B. Huang, J. Gong, and P. Li, "Dimensionality reduction based on clonal selection for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4172–4186, Dec. 2007.

[19] G. V. Lashkia and L. Anthony, "Relevant, irredundant feature selection and noisy example elimination," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 888–897, Apr. 2004.

[20] L. Zhang, Q. Zhang, B. Du, X. Huang, Y. Y. Tang, and D. Tao, "Simultaneous spectral-spatial feature selection and extraction for hyperspectral images," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 16–28, Jan. 2018.

[21] C. Vaiphasa, A. K. Skidmore, W. F. de Boer, and T. Vaiphasa, "A hyperspectral band selector for plant species discrimination," *ISPRS J. Photogramm. Remote Sens.*, vol. 62, no. 3, pp. 225–235, Aug. 2007.

[22] P. Ghamisi, M. S. Couceiro, and J. A. Benediktsson, "A novel feature selection approach based on FODPSO and SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2935–2947, May 2015.

[23] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[24] R. Zhang and J. Ma, "Feature selection for hyperspectral data based on recursive support vector machines," *Int. J. Remote Sens.*, vol. 30, no. 14, pp. 3669–3677, 2009.

[25] J. Feng, L. Jiao, T. Sun, H. Liu, and X. Zhang, "Multiple kernel learning based on discriminative kernel clustering for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 11, pp. 6516–6530, Nov. 2016.

[26] P. Ghamisi, Y. Chen, and X. X. Zhu, "A self-improving convolution neural network for the classification of hyperspectral data," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 10, pp. 1537–1541, Oct. 2016.

[27] 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.

[28] Y. Cai, X. Liu, and Z. Cai, "BS-Nets: An end-to-end framework for band selection of hyperspectral image," Apr. 2019. [Online]. Available: arXiv:1904.08269.

[29] L. Sifre and S. Mallat, "Rigid-motion scattering for image classification," Ph.D. dissertation, École Polytechnique, 2014.

[30] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017. [Online]. Available: arXiv:1704.04861.

[31] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4510–4520.

[32] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1800–1807.

[33] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, "Wavelet shrinkage: Asymptopia?" *J. Roy. Stat. Soc. B*, vol. 57, no. 2, pp. 301–337, 1995.

[34] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–13.

[35] C. Szegedy *et al.*, "Going deeper with convolutions," 2014. [Online]. Available: arXiv:1409.4842.

[36] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.

[37] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet classification using binary convolutional neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1603.05279.

[38] G. Hinton, *Neural Networks for Machine Learning*, Univ. Toronto, Toronto, ON, Canada, 2012.

[39] M. Courbariaux, Y. Bengio, and J. P. David, "BinaryConnect: Training deep neural networks with binary weights during propagation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, Dec. 2015, pp. 3123–3131.
[40] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training neural networks with weights and activations constrained to $+1$ or -1," Feb. 2016. [Online]. Available: arXiv:1602.02830.
[41] M. Alizadeh, J. Fernández-Marqués, N. D. Lane, and Y. Gal, "An empirical study of binary neural networks' optimisation," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 3123–3131.
[42] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proc. IEEE*, vol. 105, no. 12, pp. 2295–2329, Dec. 2017.
[43] C. Debes *et al.*, "Hyperspectral and LiDAR data fusion: Outcome of the 2013 GRSS data fusion contest," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2405–2418, Jun. 2014.
[44] 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.
[45] H. Wu and S. Prasad, "Semi-supervised deep learning using pseudo labels for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1259–1270, Mar. 2018.
[46] S. Hao, W. Wang, Y. Ye, T. Nie, and L. Bruzzone, "Two-stream deep architecture for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 4, pp. 2349–2361, Apr. 2018.

Jie Feng (Member, IEEE) received the B.S. degree from Chang'an University, Xi'an, China, in 2008, and the Ph.D. degree from Xidian University, Xi'an, in 2014.

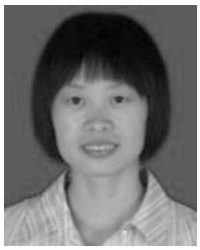
She is currently an Associate Professor with the Laboratory of Intelligent Perception and Image Understanding, Xidian University. Her current interests include remote sensing image processing, deep learning, and machine learning.

Jiantong Chen received the B.S. degree from the Qingdao University of Science and Technology, Qingdao, China, in 2017. He is currently pursuing the M.S. degree with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, School of Electronic Engineering, Xidian University, Xi'an, China.

His current interests include machine learning, remote sensing image processing, and pattern recognition.

Qigong Sun (Student Member, IEEE) received the B.S. degrees in intelligence science and technology from Xidian University, Xi'an, China, in 2015, where he is currently pursuing the Ph.D. degree in circuits and systems.

He is currently a member with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, and International Research Center for Intelligent Perception and Computation, Xidian University. His research interests include deep learning and image processing.



Ronghua Shang (Member, IEEE) received the B.S. degree in information and computation science and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2003 and 2008, respectively.

She is currently a Professor with Xidian University. Her current research interests include machine learning, pattern recognition evolutionary computation, image processing, and data mining.

Xianghai Cao (Member, IEEE) received the B.E. and Ph.D. degrees from the School of Electronic Engineering, Xidian University, Xi'an, China, in 1999 and 2008, respectively.

Since 2008, he has been with the School of Artificial Intelligence, Xidian University, where he is an Associate Professor. His research interests include remote sensing image processing, pattern recognition, and deep learning.

Xiangrong Zhang (Senior Member, IEEE) received the B.S. and M.S. degrees in computer science and technology and the Ph.D. degree in pattern recognition and intelligent system from Xidian University, Xi'an, China, in 1999, 2003, and 2006, respectively.

She is currently a Professor with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, School of Electronic Engineering, Xidian University. Her current research interests include visual information analysis and understanding, pattern recognition, and machine learning.

Licheng Jiao (Fellow, IEEE) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

He has authored or coauthored more than 150 scientific papers. His research interests include image processing, natural computation, machine learning, and intelligent information processing. He has charged of about 40 important scientific research projects, and published more than 20 monographs and a 100 papers in international journals and conferences.