Hierarchical CNN Classification of Hyperspectral Images Based on 3-D Attention Soft Augmentation

Xinyuan Miao, *Student Member, IEEE*, Ye Zhang, *Member, IEEE*, Junping Zhang, *Senior Member, IEEE*, and Xuejian Liang, *Graduate Student Member, IEEE*

***Abstract*—The distinction of similar classes has always been the core issue in image classification. In this article, a new hierarchical classification process based on three-dimensional attention soft augmentation (HC-3DAA) is proposed to improve the accuracy of classifiers, especially for the accuracy between similar classes. In HC-3DAA processing, the merging matrix is first constructed based on the validation confusion matrix to measure the similarity among different classes. Specifically, the 3-D attention soft augmentation module combined with CutMix is designed for guiding the network model to focus on the key discriminative features. Then, the extracted 3-D feature differences between similar classes are inserted into the attention module for the reclassification to get higher classification accuracy. To evaluate the performance of HC-3DAA, CutMix models with different feature dimensions and the HC module are separately discussed on two widely used hyperspectral datasets. Two different classifiers 3-D convolutional neural network and ResNet are included in the comparative analysis. Besides, experimental results also demonstrate that the proposed HC-3DAA outperforms several state-of-the-art attention-based methods.**

***Index Terms*—Data augmentation, hierarchical classification (HC), hyperspectral image, similar classes reclassification, three-dimensional (3-D) attention neural network.**

I. Introduction

HYPERSPECTRAL imagery (HSI) has hundreds of contiguous bands, which provide rich spectral information for material identification. Based on the spectral and spatial characteristics, the HSI classification assigns each pixel of an HSI to a certain label and has been one of the most pervasive tasks among the applications of HSI [1]–[4].

Over the past years, numerous HSI classification methods have been proposed aiming at effective feature extraction and accurate classification results, such as the maximum-likelihood classification (MLC) method [5], random forest [6]–[8], and support vector machine (SVM) [9]–[12]. These methods have been exploited for solving varied and numerous classification problems. However, the MLC, random forest, and SVM are characterized as "shallow" models [13] as compared to deep networks which are able to extract hierarchical, deep feature representations. Recently, deep learning, which is mainly characterized by deep networks, has been proposed and has been quite successful in solving a wide range of problems, such as natural language processing [14], [15], computer vision [16]–[19], etc.

In hyperspectral image classification, some deep-learning-based methods, such as deep belief network [20], stacked autoencoder (SA) [21], recurrent neural network (RNN) [22]–[24], and deep convolutional neural networks (CNNs) [25]–[27], have achieved encouraging performance because of its excellent performance in deep feature extraction. DBA, SA, and RNN are usually exploited to extract only spectral features from spectral signatures. However, it is not enough for satisfactory classification accuracy. The CNN can extract not only the spectral features but also the spatial features from the hyperspectral image. It has better performance in the HSI classification task and has become one of the most popular deep frameworks in deep-learning-based methods. However, there are two issues in the CNN network design for hyperspectral image classification: 1) how to efficiently get the discriminative features among different classes; 2) how to avoid the overfitting problem to get a robust model. To tackle these problems, many promising strategies have been proposed, in which the attention mechanism and data augmentation are two grateful methods and have proved to be effective for encouraging classification accuracy.

The attention mechanism is inspired by the human visual system to capture key features from images for classification [28]–[30]. Zhang *et al.* [31] and Diao *et al.* [32] captured salient areas of an image using saliency detection, which can be regarded as the early attention mechanism. Hu *et al.* [33] used a squeeze-and-excitation (SE) module to guide CNNs to automatically learn the different importance of different channel features. However, SENet only focuses on which layers of the channel level will have stronger feedback ability, it does not get the attention in the spatial dimension. Then, a convolutional block attention module [34] was proposed and applied attention to both spectral and spatial dimensions. Li *et al.* [35] proposed a dynamic selection mechanism in CNNs named selective kernel unit, in which multiple branches with different kernel sizes were fused using softmax attention. Y. Cao *et al.* [36] constructed a global context network with lightweight property, which generally

Manuscript received March 2, 2022; revised April 4, 2022 and April 24, 2022; accepted May 8, 2022. Date of publication May 11, 2022; date of current version June 2, 2022. This work was supported by the National Natural Science Foundation of China under Grant 61871150. *(Corresponding author: Ye Zhang.)*

The authors are with the Department of Information Engineering, School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China (e-mail: hit_mxy@hit.edu.cn; zhye@hit.edu.cn; zhangjp@hit.edu.cn; liangxuejian@hit.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3174301

outperforms SENet on major benchmarks for various recognition tasks. Recently, Mou and Zhou [37] designed a network unit, which was termed the spectral attention module, that made use of a gating mechanism to adaptively recalibrate spectral bands by selectively emphasizing informative bands and suppressing less useful ones. However, similar to the SENnet, this method only used the attention module in the spectral dimension. Sun *et al.* [38] proposed a spectral-spatial attention network (SSAN) to capture discriminative spectral-spatial features from attention areas of HSI cubes and the attention module was embedded after every spectral and spatial kernel. It has been proved that SSAN outperformed several state-of-the-art methods.

The attention-mechanism-based methods mentioned above usually obtain the discriminative features by designing the structure of the CNN network. The network structure can be particularly complex. Besides, the features selected by the attention module are poor in interpretability and intuitiveness. So, data augmentation can be a more direct and effective way for HSI classification.

Recently, several promising strategies of data augmentation based on regional dropout [39]–[44] have been proposed to enhance the performance of CNN classifiers. The regional dropout methods erase random regions on the input to improve generalization and localization. They have proved to be effective for guiding the model to attend to not only the most discriminative parts of objects but rather the entire object region. However, the informative pixels' removal can cause information loss in the image. To tackle this problem, Zhang *et al.* [45] mixed training samples by interpolating both the image and labels, which certainly improved the classification performance. A recently developed strategy called CutMix [46] overcome the problem of Mixup [45] that samples tend to be unnatural after the mix of training samples. Instead of simply mixing pixels, it replaced the partial region of one image with a patch region in another image. CutMix shared similarities with Mixup which mixed two samples by interpolating both the image and labels. But the enhanced sample was more natural. It was shown that CutMix had a better performance than Cutout and Mixup. However, the original ∗∗∗∗2-D CutMix only focuses on the enhancement of spatial information so the spectral information in HSI is omitted.

Both attention mechanism and data augmentation show their benefits in HSI classification. No matter what type of method, it can usually achieve satisfactory results in classes with obvious differences. Therefore, the final classification performance of a classifier is mainly limited by the accuracy among similar classes. These similar classes can often belong to one same big category (tree, grass, etc.), or be made of similar materials (roof and road that are made of concrete, etc.), or be spatial adjacency (bare land, grassland, etc.), and so on. Then the discriminative features to distinguish these similar classes tend to focus on some specific spatial and spectral characteristics and the others can be redundancy and interference, which will affect the discrimination of the classifier. The unique assets of hyperspectral images are their rich spectral and spatial content, so it is more necessary for HSI to extract discriminative and efficient features for the classification task.

In this article, the following questions are addressed based on the research status mentioned above.

1) Since different spectral bands and spatial pixels contribute not equally to a CNN for classification tasks, how to task-drivenly find informative features and suppress redundant ones?
2) Since both the attention mechanism and data augmentation have good use in feature extraction, how to combine these two methods to get the informative 3-D features more effectively?
3) Since the key issue of classification tasks is the differentiation of similar classes, how to design a reasonable classification structure for these classes so that it can obtain more accurate classification accuracy?

Motivated by the above problems, we design a three-dimensional attention soft augmentation (3DAA) module for analyzing the significance of different spectral bands and spatial regions of the HSI image. Besides, we design a new classification process named hierarchical classification (HC) to help the classifier better focus on the distinction between similar classes for more precise classification accuracy. The main contributions are listed as follows.

1) The 3DAA module: A spectral-spatial (3-D) feature extraction module that combines the attention mechanism and data augmentation. The 3-D attention mechanism is used to guide the CNN to attend to the most discriminative features of HSI. Besides, the 2-D CutMix is expanded to 3-D CutMix and adopted into the attention part for better performance in feature extraction.
2) The HC process: To further improve the classification accuracy of similar classes, the HC process is designed as a coarse-fine two steps method. HC first measures the similarity between classes based on the initial classification results of validation samples and then the similar classes are precisely reclassified on the premise of avoiding interference from the other classes.

The rest of this article is organized as follows. First, Section II introduces the proposed method named the HC based on 3-D attention soft augmentation (HC-3DAA). Second, Section III verifies the proposed approach and presents the corresponding analysis and discussion. Finally, Section IV concludes the article.

II. Proposed Method

A. Framework of the Proposed Method

The framework of our proposed HC-3DAA is shown in Fig. 1. The whole framework consists of two main parts: the 3-D attention soft data augmentation module combined with CutMix, the HC process for reclassification of similar classes.

First, sliding windows with fixed sizes are used to divide the HSI into patches. Then all patches are then divided into several batches and input into the 3DAA module. In the 3-D CutMix part, the dropped region of every patch in one training batch is replaced by the same region of one randomly selected sample. The trainable 3-D attention module is then applied to the mixed patch to extract discriminative features of HSI. Second,

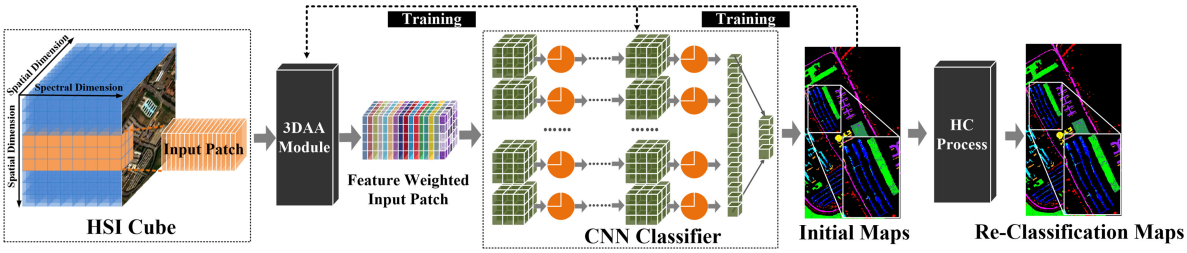

Fig. 1. Framework of HC-3DAA for HSI classification. The whole framework consists of two main parts: the 3-D attention soft data augmentation module combined with CutMix, the HC process for reclassification of similar classes.

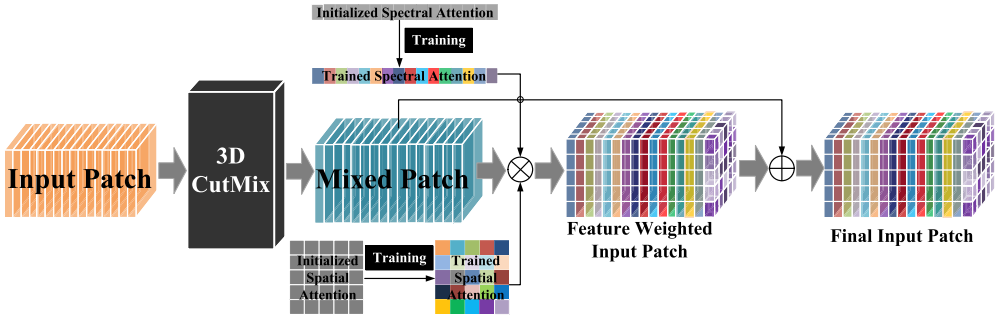

Fig. 2. Architecture of the 3DAA module. The proposed 3DAA module consists of two main parts: the 3-D CutMix part and the 3-D feature attention part. This module aims at generating augmented training samples, which can be effective for guiding the CNN classifier to attend on discriminative features of HSI.

the feature-weighted data are input into the CNN classifier for training and testing to get the initial classification results on validation and testing samples. Then, the merging matrix using validation samples is designed to measure the similarity among classes. Two classes with similar features will be merged into one big category. Every category will be reclassified after retraining and retesting. Finally, the initial coarse classification map will be corrected using the reclassification map of every category one by one.

B. 3DAA Module

The proposed 3DAA module consists of two main parts: the 3-D CutMix part and the 3-D feature attention part. This module aims at generating augmented training samples, which can be effective for guiding the CNN classifier to focus on discriminative features of HSI. Fig. 2 illustrates the architecture of the 3DAA module.

Original 2-D CutMix only used spatial information in RGB images. But for HSI, the spectral information should also be utilized. So in this article, the CutMix method is expanded from 2-D to 3-D to get a better fit for the subsequent 3-D attention part.

Let $\mathbf{X}_i \in \mathbb{R}^{B\times W\times W}$ denotes a 3-D training patch with B bands and sliding window size W. y represents the label of $\mathbf{X}_i$. Then, the training batch can be represented as $\mathbf{X} = [(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \ldots, (\mathbf{X}_n, y_n)]$, where n is the batch size. The purpose of 3-D CutMix is to generate a new training batch $\tilde{\mathbf{X}} = [(\tilde{\mathbf{X}}_1, \tilde{y}_1), (\tilde{\mathbf{X}}_2, \tilde{y}_2), \ldots, (\tilde{\mathbf{X}}_n, \tilde{y}_n)]$ by replacing a 3-D region $\mathbf{B}^{\bar{b}\times\bar{x}\times\bar{y}}$ in $\mathbf{X}_i (i = 1, 2, \ldots, n)$ with the same region of one randomly selected sample $\mathbf{X}_R$. The label of the new sample $\tilde{\mathbf{X}}_i (i = 1, 2, \ldots, n)$ is also the mixed label of y_i and y_R. We define the combining operation as

$$\begin{aligned} \tilde{\mathbf{X}}_i &= \mathbf{M} \odot \mathbf{X}_i + (1 - \mathbf{M}) \odot \mathbf{X}_R \\ \tilde{y}_i &= \lambda y_i + (1 - \lambda) y_R \end{aligned} \tag{1}$$

where $\mathbf{M} \in \{0, 1\}^{B\times W\times W}$ is a binary mask whose values are 0 in $\mathbf{B}^{\bar{b}\times\bar{x}\times\bar{y}}$ and 1 in the other regions. $\odot$ is elementwise multiplication. Like 2-D CutMix, the combination ratio λ is sampled from the uniform beta distribution Beta (1, 1).

The position of the 3-D cropped region $\mathbf{B}^{\bar{b}\times\bar{x}\times\bar{y}}$ is represented as $[(r_{b1}, r_{b2}), (r_{x1}, r_{x2}), (r_{y_1}, r_{y2})]$, in which $\bar{b} = b_2 - b_1$, $\bar{x} = x_2 - x_1$, and $\bar{y} = y_2 - y_1$. The position is uniformly sampled according to

$$\begin{aligned} r_{b1} &\sim \text{Unif}\,(0, B)\,, r_{b2} = B(1-\lambda)^{\frac{1}{3}} \\ r_{x1} &\sim \text{Unif}\,(0, W)\,, r_{x2} = W(1-\lambda)^{\frac{1}{3}} \\ r_{y1} &\sim \text{Unif}\,(0, W)\,, r_{y2} = W(1-\lambda)^{\frac{1}{3}} \end{aligned} \tag{2}$$

to make the cropped region meet $(\bar{b} \times \bar{x} \times \bar{y})/(B \times W \times W) = 1 - \lambda$.

The 3-D CutMix part can enhance the performance of the 3-D attention part in feature extraction by constructing new training samples $\tilde{\mathbf{X}}_i$ and labels $\tilde{y}_i$. It can also avoid the overfitting problem of CNN to some extent and make the model more robust.

The subsequent part after 3-D CutMix is the 3-D attention part based on the attention mechanism to capture discriminative spectral-spatial features among classes.

To further improve the perception of features, the 3-D feature attention part is applied to every new training patch.

Let $\boldsymbol{A}_{\text{spe}} \in \mathbb{R}^B$ and $\boldsymbol{A}_{\text{spa}} \in \mathbb{R}^{W\times W}$ denote spectral and spatial attention weights, respectively. Their sizes correspond to the spectral and spatial dimensions of the patch. Both $\boldsymbol{A}_{\text{spe}}$ and $\boldsymbol{A}_{\text{spa}}$ are trainable variables that can be optimized by CNN. The elements in $\boldsymbol{A}_{\text{spe}}$represent the different contributions of every band for HSI classification, while the elements in $\boldsymbol{A}_{\text{spa}}$ represent the different contributions of every spatial region. Both $\boldsymbol{A}_{\text{spe}}$ and $\boldsymbol{A}_{\text{spa}}$ are normalized to [0, 1] and a high value means a high contribution to the classification task.

Before the training of these attention weights, we need to initialize them properly. It is noted that the way of initialization is different between the classification and reclassification process. However, their purpose is the same, that is to increase the feature difference among classes. The weight initialization method of preliminary classification will be introduced below and that of reclassification will be introduced in the subsection of the HC process.

Assuming that $\mathbf{P}_{C_i} \in \mathbb{R}^{B\times W\times W}$ means the average training samples of class *i*, then the average spectral and spatial information can be defined as $\mathbf{Spe}_{C_i} \in \mathbb{R}^B$ and $\mathbf{Spa}_{C_i} \in \mathbb{R}^{W\times W}$, respectively.

$$\begin{aligned}\mathbf{Spe}_{C_i} &= \text{Avg}_{\text{spa}}(\mathbf{P}_{c_i}) = \left[\mathbf{E}_1^{C_i}, \mathbf{E}_2^{C_i}, \ldots, \mathbf{E}_B^{C_i}\right] \\ \mathbf{Spa}_{C_i} &= \text{Avg}_{\text{spe}}(\mathbf{P}_{c_i}) = \begin{bmatrix} \mathbf{a}_{11}^{C_i} & \ldots & \mathbf{a}_{1W}^{C_i} \\ \vdots & \ddots & \vdots \\ \mathbf{a}_{W1}^{C_i} & \ldots & \mathbf{a}_{WW}^{C_i} \end{bmatrix}\end{aligned} \tag{3}$$

where operators $\text{Avg}_{\text{spa}}(\bullet)$ and $\text{Avg}_{\text{spe}}(\bullet)$ mean to get the value average in spatial and spectral dimensions. $\mathbf{E}_j^{C_i}$ means the average spectral value of the *i*th class in band *j* and $\mathbf{a}_{pq}^{C_i}$ means the average spatial value of the *i*th class at the position (*p, q*). To measure the spectral and spatial differences among classes, we define the difference factors $\boldsymbol{\sigma}_{\text{spe}}$ and $\boldsymbol{\sigma}_{\text{spa}}$ as the following: (4) shown at the bottom of this page, where $\text{var}(\bullet)$ means to calculate the variance of the variables and *N* is the number of classes. So, if there is much spectral feature difference among classes in band *j*, the value $\text{var}(\{\mathbf{Spe}_j^{C_1}, \ldots, \mathbf{Spe}_j^{C_N}\})$ must be high. The same goes for the spatial feature difference.

A_{spe} and A_{spa} are initialized as the normalized $\boldsymbol{\sigma}_{\text{spe}}$ and $\boldsymbol{\sigma}_{\text{spa}}$, respectively, which can be expressed as

$$\begin{aligned}A_{\text{spe}} &= \mathcal{N}(\boldsymbol{\sigma}_{\text{spe}}) = \frac{\boldsymbol{\sigma}_{\text{spe}} - \min(\boldsymbol{\sigma}_{\text{spe}})}{\max(\boldsymbol{\sigma}_{\text{spe}}) - \min(\boldsymbol{\sigma}_{\text{spe}})} \\ A_{\text{spa}} &= \mathcal{N}(\boldsymbol{\sigma}_{\text{spa}}) = \frac{\boldsymbol{\sigma}_{\text{spa}} - \min(\boldsymbol{\sigma}_{\text{spa}})}{\max(\boldsymbol{\sigma}_{\text{spa}}) - \min(\boldsymbol{\sigma}_{\text{spa}})}\end{aligned} \tag{5}$$

where $\mathcal{N}(\bullet)$ is the linear normalization operator.

Then, both of the weights are expanded to the same size of $\tilde{\mathbf{X}}_i \in \mathbb{R}^{B\times W\times W}$:

$$\begin{aligned}\mathbf{A}_{\text{spe}} &= (\boldsymbol{A}_{\text{spe}})^{B\times W\times W} \\ \mathbf{A}_{\text{spa}} &= (\boldsymbol{A}_{\text{spa}})^{B\times W\times W}.\end{aligned} \tag{6}$$

The final 3-D attention weight $\mathbf{A}_w$ and the attention weighted patch $\widehat{\mathbf{X}}_i$ are represented as

$$\begin{aligned}\mathbf{A}_w &= \mathcal{N}(\mathbf{A}_{\text{spe}} \odot \mathbf{A}_{\text{spa}}) \\ \widehat{\mathbf{X}}_i &= (\mathbf{1} + \mathbf{A}_w) \odot \tilde{\mathbf{X}}_i.\end{aligned} \tag{7}$$

Both $\boldsymbol{A}_{\text{spe}}$ and $\boldsymbol{A}_{\text{spa}}$ are trainable parameters. They will be optimized with the training of the CNN.

C. HC Process

The HC process is a coarse-fine two steps method to help the classifier attend to the differentiation between similar classes. Fig. 3 illustrates the architecture of the HC process.

The purpose of the HC process is to measure the similarity between classes and then get more precise reclassification results of these categories. Because it is easier and more efficient for the CNN model to catch the discriminative features between similar classes without the interference of the other classes. It can effectively solve the problem of information inequality between classes by transforming the classification task from the "one against others" problem to the "one against one" problem.

The HC process mainly includes the following four parts.

1) Coarse classification on validation samples to get the merging matrix for similarity measure among classes.
2) The initial attention feature weights are obtained for each group of similar classes.
3) Coarse classification on testing samples to get the initial classification map.
4) Fine classification for each group of similar classes to correct the initial classification map.

First, we need to define the "similar" classes. The "similar" means the feature difference between the two classes is too small and is difficult for the classifier to perceive. Because characteristics of features extracted by different classifiers can be different, "similar classes" may not be all the same for these classifiers even for the same image. So a classifier needs to determine its own "similar classes". In this article, we use the validation samples to determine which classes are "similar" for a specific classifier.

$$\begin{aligned}\boldsymbol{\sigma}_{\text{spe}} &= \text{var}\left(\{\mathbf{Spe}_{C_1}, \ldots, \mathbf{Spe}_{C_N}\}\right) \\ &= [\text{var}(\{\mathbf{Spe}_1^{C_1}, \ldots, \mathbf{Spe}_1^{C_N}\}), \ldots, \text{var}(\{\mathbf{Spe}_B^{C_1}, \ldots, \mathbf{Spe}_B^{C_N}\})] \\ \boldsymbol{\sigma}_{\text{spa}} &= \text{var}\left(\{\mathbf{Spa}_{C_1}, \ldots, \mathbf{Spa}_{C_N}\}\right) \\ &= \begin{bmatrix} \text{var}(\{\mathbf{Spa}_{11}^{C_1}, \ldots, \mathbf{Spa}_{11}^{C_N}\}) & \ldots & \text{var}(\{\mathbf{Spa}_{1W}^{C_1}, \ldots, \mathbf{Spa}_{1W}^{C_N}\}) \\ \vdots & \ddots & \vdots \\ \text{var}(\{\mathbf{Spa}_{W1}^{C_1}, \ldots, \mathbf{Spa}_{W1}^{C_N}\}) & \ldots & \text{var}(\{\mathbf{Spa}_{WW}^{C_1}, \ldots, \mathbf{Spa}_{WW}^{C_N}\}) \end{bmatrix}\end{aligned} \tag{4}$$

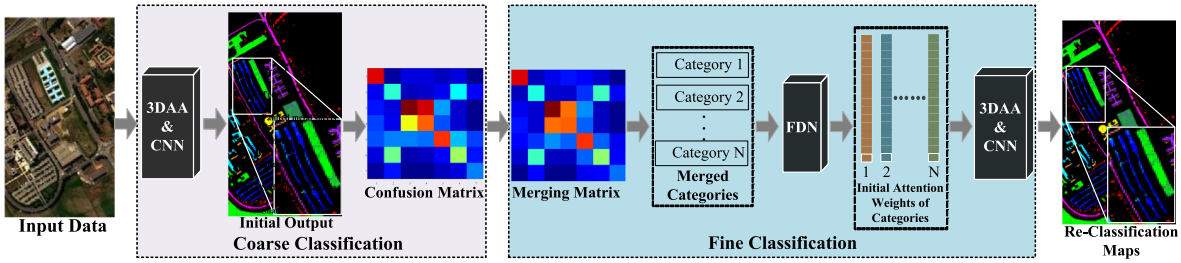

Fig. 3. Architecture of the HC process. The HC process measures the similarity between classes based on the merging matrix and merge similar classes into different categories. Then, every category will be reclassified to correct the initial testing map.

After testing on validation samples in the coarse classification process, the merging matrix $\mathbf{G}$ is defined based on the validation confusion matrix $\mathbf{M} \in \mathbb{R}^{N\times N}$ of the initial validation classification result as

$$\mathbf{G} = \mathbf{M} + \mathbf{M}^T \tag{8}$$

where N is the number of classes. It is obvious that $\mathbf{G}$ is a symmetric matrix. Its nondiagonal elements $g_{i,j}$ represent the confusion degree between class C_i and C_j. The element value represents the degree of similarity between the two classes. If there is a higher similarity between the two classes C_i and C_j, the corresponding element $g_{i,j}$ of the merging matrix will have a greater value. Then the elements will be selected based on their values from high to low and the corresponding classes will be merged into categories $R_{i,j}$ two by two.

$$R_{i,j}^k = \{C_i \cup C_j \,|g_{i,j} = g_{j,i} = k\text{th\_}\max(\mathbf{G}),\, i \neq j\} \tag{9}$$

where $R_{i,j}^k$ means the category containing class *i* and *j* in the *k*th mergence. $k\text{th\_}\max(G)$ means *k*th largest value in $\mathbf{G}$. It is noted that two categories $R_{i,p}^m$ and $R_{i,q}^n$ may contain one of the same class *i*. However, it is common and reasonable because they will meet the definition of similar classes from different views. For example, C_i and C_p may be made of similar materials, but C_i and C_q can also be spatial adjacency.

If the two similar classes are regarded as one same category, the original samples of misclassification between these two classes can be also regarded as correctly classified from the perspective of category. It is obvious that the overall accuracy (OA) is improved continuously with the continuous mergence. When the C_i and C_j are merged, the improved OA of the validation set $\Delta\text{OA}(R_{i,j})$ will be represented as follows:

$$\Delta\text{OA}(R_{i,j}) = \frac{g_{i,j}}{N_{\text{Val}}} \tag{10}$$

where N_{Val} is the number of samples in validation set.

In the extreme case, if all classes that may lead to misclassification have been merged into some categories, the overall classification accuracy will be 1, but this case is meaningless. To improve the classification accuracy more simply and efficiently and save computing resources, we expect to get a satisfactory improvement of overall classification accuracy (OA) on the premise of merging as few categories as possible. So, the merging times T is limited as

$$T = \arg\min_T \left\{ \Delta\text{OA}(T) < \frac{\Delta\text{OA}(1)}{2} \right\} \tag{11}$$

where $\Delta\text{OA}(i)$ means the improvement of OA after i times merging. Because classes are merged based on the values in $\mathbf{G}$ from high to low, it is obvious that $\Delta\text{OA}(1)$ is the biggest. The more times of mergence, the less obvious in the improvement of OA.

Second, these categories need to be reclassified for more precise accuracy. In the HC process, the initialization of the 3DAA module is different from that in the "coarse" classification mentioned in Section II-B. To make the features more discriminative, we use the feature difference of similar classes to initialize the attention weights in the 3DAA module.

The feature differences normalization module is used for getting the spectral $\mathbf{\Delta}_{\text{spe}}^{R_{i,j}}$ and spatial differences $\mathbf{\Delta}_{\text{spa}}^{R_{i,j}}$ in every category.

$$\begin{aligned}\mathbf{\Delta}_{\text{spe}}^{R_{i,j}} &= \mathcal{N}\left(\left|\text{Avg}_{\text{spa}}(\mathbf{P}_{C_i}) - \text{Avg}_{\text{spa}}(\mathbf{P}_{C_j})\right|\right)\\ \mathbf{\Delta}_{\text{spa}}^{R_{i,j}} &= \mathcal{N}\left(\left|\text{Avg}_{\text{spe}}(\mathbf{P}_{C_i}) - \text{Avg}_{\text{spe}}(\mathbf{P}_{C_j})\right|\right)\end{aligned} \tag{12}$$

where Avg_{spe} and Avg_{spa} denote the value average of patches in spectral and spatial dimensions, respectively.

The last step is to recalibrate the initial classification results.

After the coarse classification of the testing samples, we can get the coarse classification map Map^{Ini}, in which there are many misclassification samples in each category. It is assumed that $\text{Map}_{R_{i,j}}^{\text{Ini}}$ represent the test samples classified into C_i or C_j in the coarse classification process. In the HC process, these samples will be reclassified. For each category, this process is a binary classification and we just need to judge which of the two similar classes these samples belong to without the interference of the other classes. The reclassification results of these samples will correct the labels in the coarse classification map $\text{Map}_{R_{i,j}}^{\text{Ini}}$, which can be expressed as:

$$\text{Map}_{R_{i,j}}^{\text{Cor}} = \text{HC}\left(R_{i,j}\right) = \text{HC}\left(\text{Map}_{R_{i,j}}^{\text{Ini}}\right). \tag{13}$$

Every category will be reclassified one by one. In the reclassification of $R_{i,j}^k$ in the HC process, the attention weights $\boldsymbol{A}_{\text{spe}}$ and $\boldsymbol{A}_{\text{spa}}$ in the 3DAA module will be initialized as the normalized spectral and spatial feature differences of each category, $\mathbf{\Delta}_{\text{spe}}^{R_{i,j}}$ and $\mathbf{\Delta}_{\text{spa}}^{R_{i,j}}$, respectively.

After the correction on Map^{Ini} with the fine classification results of every category one by one, the final fine classification

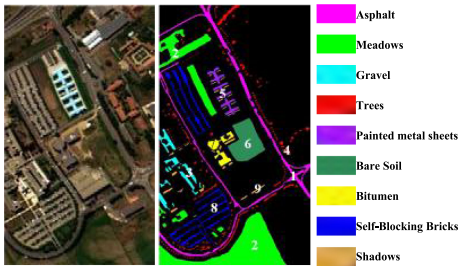

Fig. 4. False-color image of PU and its ground-truth map.

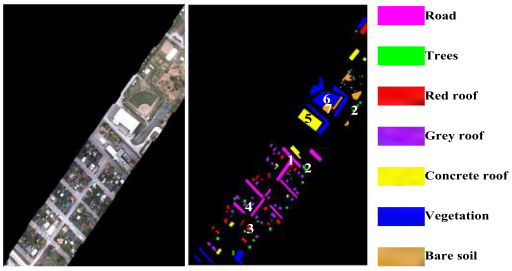

Fig. 5. RGB image of DFC data and its ground-truth map.

map can be expressed as

$$\mathrm{Map}_R^{\mathrm{Cor}} = \begin{cases} \mathrm{HC}\left(\mathrm{Map}_{R_{i,j}^k}^{\mathrm{Ini}}\right) & R \in R_{i,j}^k \ k = 1, 2, \ldots, T \\ \mathrm{Map}^{\mathrm{Ini}} & R \notin R_{i,j}^k \end{cases} \quad (14)$$

where T means the number of categories.

The reasons why the HC process can better distinguish similar classes can be summarized as follows.

1) From the perspective of data enhancement, the reclassification process of similar classes belongs to binary classification, and the samples between these classes can be mixed more fully.
2) From the perspective of feature attention, the trained feature attention weights are more targeted for distinguishing similar classes in each category.
3) From the perspective of classifier design, it is easier for the classifier to find the appropriate decision boundaries for distinguishing the two classes without the interference of other classes.

III. Experiments and Discussion

A. Data Description

Three publicly available hyperspectral datasets were used for experiments, i.e., the Pavia University (PU) dataset, the 2014 IEEE GRSS Data Fusion Contest (DFC) dataset, and the SV dataset. Figs. 4 –6 illustrate the false-color image, RGB image, and its corresponding ground-truth map for the PU, the DFC, and the SV datasets, respectively.

1) PU Hyperspectral Dataset: The PU dataset is acquired by the ROSIS sensor during a flight campaign over the University of Pavia, northern Italy. The size of PU is 610×304 pixels and the number of spectral bands is 115, of which 12 bands were removed before being processed because of the noise. The band ranges from 0.43 to 0.86 μm. The spatial resolution is 1.3 m/pixel. Besides unknown pixels, nine classes are manually annotated in the reference data.

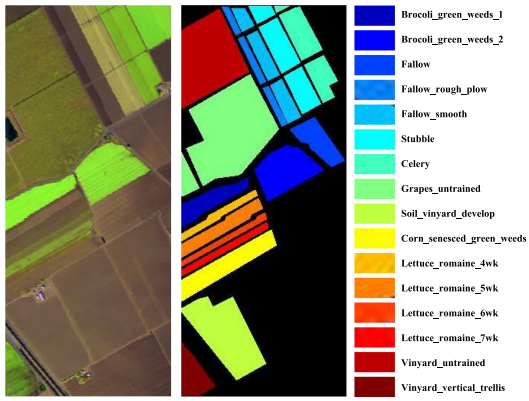

Fig. 6. False-color image of SV and its ground-truth map.

2) 2014 IEEE GRSS DFC Dataset: DFC involves two image sources acquired at different spectral ranges, namely the hyperspectral image in the long-wave infrared region (LWIR) (7.8–11.5 μm) with 84 bands and the RGB image in the visible (VIS) region. The LWIR image was acquired using the "Hyper-Cam," an airborne long-wave infrared hyperspectral imager based on a Fourier-transform spectrometer. The two data sources cover an urban area near Thetford Mines in Québec, Canada. The spatial resolution is 1 m/pixel in LWIR data and 0.2 m/pixel in RGB data. In this article, we combine the RGB data and the LWIR data of the DFC dataset as a complete HSI image. All data are resized to 795×564 pixels and 1 m/pixel. There are seven classes in the scene.

3) Salinas Valley Hyperspectral Dataset: This scene was collected by the 224-band AVIRIS sensor over Salinas Valley (SV), California, and is characterized by high spatial resolution (3.7-m pixels). The area covered comprises 512 lines by 217 samples. We discarded the 20 water absorption bands, in this case bands: [108-112], [154-167], 224. This image was available only as at-sensor radiance data. It includes vegetables, bare soils, and vineyard fields. Salinas ground-truth contains 16 classes.

B. Experimental Setup

To evaluate the performance of HC-3DAA, different kinds of CutMix modules with different feature dimensions and the HC module are discussed separately on two hyperspectral datasets mentioned above. All datasets are normalized in the range [0, 1]. It is noted that the RGB image and LWIR image are normalized separately in the DFC dataset.

In this article, we use the merging matrix $\mathbf{G}$ to determine which classes are "similar" for a specific classifier. The element value of $\mathbf{G}$ represents the degree of similarity between the two classes. So the number of each class in the validation set must be equal to ensure that the confidence among classes is balanced. If we select samples of the validation set in proportion, classes with a large sample base are more likely to be judged as "similar" classes. Therefore, we cannot judge the similarity among classes

TABLE I
AMOUNTS OF TRAINING, VERIFICATION, AND TEST DATA ON THE PU DATASET

Class No.	Class Name	Training	Validation	Testing
1	Asphalt	200	200	6631
2	Meadows	200	200	18649
3	Gravel	200	200	2099
4	Trees	200	200	3064
5	Printed Metal Sheets	200	200	1345
6	Bare Soil	200	200	5029
7	Bitumen	200	200	1330
8	Bricks	200	200	3682
9	Shadows	200	200	947
	TOTAL	1800	1800	42776

TABLE II
AMOUNTS OF TRAINING, VERIFICATION, AND TEST DATA ON THE DFC DATASET

Class No.	Class Name	Training	Validation	Testing
1	Road	200	200	4443
2	Trees	200	200	1093
3	Red Roof	200	200	1854
4	Grey Roof	200	200	2126
5	Concrete Roof	200	200	3888
6	Vegetation	200	200	7357
7	Bare Soil	200	200	1771
	TOTAL	1400	1400	22532

TABLE III
AMOUNTS OF TRAINING, VERIFICATION, AND TEST DATA ON THE SV DATASET

Class No.	Class Name	Training	Validation	Testing
1	Brocoli_green_weeds_1	200	200	2009
2	Brocoli_green_weeds_2	200	200	3726
3	Fallow	200	200	1976
4	Fallow_rough_plow	200	200	1394
5	Fallow_smooth	200	200	2678
6	Stubble	200	200	3959
7	Celery	200	200	3579
8	Grapes_untrained	200	200	11271
9	Soil_vinyard_develop	200	200	6203
10	Corn_senesced_green_weeds	200	200	3278
11	Lettuce_romaine_4wk	200	200	1068
12	Lettuce_romaine_5wk	200	200	1927
13	Lettuce_romaine_6wk	200	200	916
14	Lettuce_romaine_7wk	200	200	1070
15	Vinyard_untrained	200	200	7268
16	Vinyard_vertical_trellis	200	200	1807
	TOTAL	3200	3200	54129

by the element value of **G**. So the number of validation samples of each class should be a fixed value.

However, we have no requirements for the selection way of the number of samples in the training set. The samples of the training set can be selected in proportion or a fixed value. In this article, we want to keep the number of training samples consistent with that of the validation samples, so both the training and validation samples are fixed at 200. Of course, this is not a must. The number of training, validation, and testing samples for each class are detailed in Tables I–III. It is noted that when measuring the similarity among classes, only the samples and labels of the validation set are used. When evaluating the overall classification accuracy, all samples and labels in the data are used for testing. In other words, training and validation samples are included in the testing samples, which can also be reflected in Tables I–III.

We adopt four quantitative evaluation indices to assess the classification performance and they are listed as the following.

1) *OA*: This criterion is calculated as the fraction of test samples that are differentiated correctly.
2) *kappa coefficient (κ)*: To assess the performance concerning each class in a data set, we also compute per-class accuracy (PA). This measurement is particularly useful when class labels are not uniformly distributed.
3) *PA*: This criterion is computed as the classification accuracies of each class.
4) *Confusion Matrix*: The confusion matrix is used to analyze the misclassified classes for the assessment of intraclass classification performance.

In this article, we insert the 3DAA module and HC process into a 3D-CNN and a 3D-ResNet [47] to evaluate the improvement of our proposed method. We use the Pycharm framework to implement and train networks. The Adam optimization function [48] with an initial learning rate of 1×10^{-4} has been adopted in the "coarse" classification process and 1×10^{-3} in the HC process. The batch size is set to 5 because a small batch size is more conducive to the full mixing of training samples among different classes. The epochs are set to 2000 in the "coarse" classification process and 1000 in the HC process. For comparison, the window sizes (the spatial size of the input patch) are set to 5×5 (W-5) and15×15 (W-15) for experiments. Finally, we train networks on an NVIDIA GeForce RTX 2080 Ti 11 GB GPU.

In this article, the structure of 3D-CNN is detailed as follows. First, the data patch with a specific spatial size is input into the 3DAA module to get the feature-weighted data. Then the feature-weighted data is input into several 3-D convolution layers with no padding and the input patch will be finally convoluted into a 1-D vector. The convolution layer is followed by two full connection layers. For example, the 3D-CNN architecture with a spatial size of 15 is shown in Table IV. There are seven 3-D convolution layers with ReLU active functions. Two fully connected layers are after the convolutional layers. The stride value of the convolutional layer is set to 1.

To quantitatively compare different models for hyperspectral data classification tasks from various aspects, the following three experiments are designed.

1) Effectiveness of the Proposed 3DAA Module: To evaluate the performance of the proposed 3DAA module, we compare the experimental results of two different classifiers (3D-CNN and ResNet) combined with different types of modules (2-D CutMix with attention soft augmentation and 3-D CutMix with attention soft augmentation). Thus, the followings are methods included in this experiment: 3D-CNN, ResNet, 3D-CNN with 2D/3DAA, and ResNet with 2D/3DAA. Besides, two window sizes 5×5(W-5) and 15×15(W-15) are also considered in the experiment.

2) Effectiveness of the HC Process: To evaluate the performance of the HC process, the classification results of two classifiers before and after the HC process are compared. We choose the window size setting that can better reflects the advantages of

TABLE IV
CONFIGURATION OF A 3DAA-BASED 3-D CONVOLUTIONAL NETWORK FOR THE PU DATASET WITH W-15

Layer	Input Shape	Output Shape	Channels (in_channels, out_channels)	Upper connection	Configuration
3DAA Module	(103, 15, 15)	(103, 15, 15)	-	Input data	spectral parameters 103, spatial parameters 15×15
Conv 1	(103, 15, 15)	(101, 13, 13)	(1, 8)	3DAA Module	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 2	(101, 13, 13)	(99, 11, 11)	(8, 16)	Conv 1	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 3	(99, 11, 11)	(97, 9, 9)	(16, 32)	Conv 2	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 4	(97, 9, 9)	(95, 7, 7)	(32, 64)	Conv 3	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 5	(95, 7, 7)	(93, 5, 5)	(64, 128)	Conv 4	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 6	(93, 5, 5)	(91, 3, 3)	(128, 256)	Conv 5	3×3×3 kernel, strid 1, no padding, bn, relu
Conv 7	(91, 3, 3)	(89, 1, 1)	(256, 512)	Conv 6	3×3×3 kernel, strid 1, no padding, bn, relu
fc1	(89, 1, 1)	(50,)	(512×89 , 50)	Conv 7	50 units, relu
fc2	(50,)	(9,)	(50, 9)	fc1	9 units, softmax

TABLE V
ACCURACY COMPARISONS OF 3DAA MODULE FOR THE PU DATASET WITH W-5

Class No.	Class Name	3D-CNN	3D-CNN& 2DAA	3D-CNN& 3DAA	ResNet	ResNet& 2DAA	ResNet& 3DAA
1	Asphalt	88.84	90.92	**97.15**	93.37	96.46	**98.46**
2	Meadows	91.51	**96.35**	94.99	91.1	92.99	**95.80**
3	Gravel	90.36	92.66	**98.05**	92.37	**95.09**	92.81
4	Trees	97.87	97.42	**98.30**	93.04	99.31	**99.97**
5	Printed Metal Sheets	98.52	**100.00**	**100.00**	99.92	**100.00**	**100.00**
6	Bare Soil	92.01	91.73	**97.30**	94.08	**98.23**	94.13
7	Bitumen	93.08	99.32	**99.85**	94.36	**99.85**	99.70
8	Bricks	93.22	90.68	**93.70**	93.13	**96.99**	96.47
9	Shadows	97.84	99.26	**99.79**	99.85	**100.00**	**100.00**
OA (%)	-	92.11	94.64	**96.29**	92.75	95.63	**96.57**
κ (%)	-	91.07	92.94	**95.13**	91.52	94.29	**95.48**

Note: Bold numbers indicate the best performance.

TABLE VI
ACCURACY COMPARISONS OF 3DAA MODULE FOR THE DFC DATASET WITH W-15

Class No.	Class Name	3D-CNN	3D-CNN& 2DAA	3D-CNN& 3DAA	ResNet	ResNet& 2DAA	ResNet& 3DAA
1	Road	99.48	**100.00**	99.86	99.28	**100.00**	99.93
2	Trees	93.60	**96.16**	91.49	93.41	88.75	**96.16**
3	Red Roof	99.68	**99.73**	97.52	99.95	**99.73**	98.87
4	Grey Roof	97.74	99.06	**99.58**	97.41	98.82	**99.34**
5	Concrete Roof	99.23	**99.77**	99.33	98.56	99.00	**99.07**
6	Vegetation	92.10	92.39	**95.04**	92.27	92.84	**95.49**
7	Bare Soil	99.21	99.94	**100.00**	99.38	99.94	**100.00**
OA (%)	-	96.57	97.17	**97.58**	96.47	96.81	**98.01**
κ (%)	-	95.43	96.73	**97.00**	95.22	95.79	**97.43**

Note: Bold numbers indicate the best performance.

our proposed method. The confusion matrices with the window size of 5 (W-5) on the PU dataset and with the window size of 15 (W-15) on the DFC dataset and SV dataset are also used to analyze the improvement in the classification of similar classes. The results before and after the HC process are compared in confusion matrices.

3) Performance Comparison of HC-3DAA With Other State-of-the-Art Methods: In this experiment, we compare the performance of our proposed HC-3DAA method using 3D-CNN classifier with some state-of-the-art attention-based methods: deep feature fusion network (DFFN) [30], spectral-spatial residual network (SSRN) [27], spectral attention network (SpecAttenNet) [37], and SSAN [38].

For a fair comparison, the proposed method and compared methods adopt the same experimental settings. The spatial window size of HSI cubes for all methods is set to 15×15. The batch size is set to 5. The number of training epochs is set to 2000 and weight parameters of all methods are optimized with the Adam. The learning rate is set to 1×10^{-4}.

C. Results and Discussion

1) Effectiveness of the Proposed 3DAA Module: Tables V–VII give information about PA, OAs, and kappa coefficients with W-5 on the PU dataset and with W-15 on the DFC dataset and SV dataset. In the experiments, the accuracy of classifiers with no attention module, 2DAA module, and 3DAA module are compared. Tables V–VII list the results with W-5 on the PU dataset and with W-15 on the DFC, and SV datasets, respectively. Table VIII shows the OA and kappa coefficients in all cases.

TABLE VII
ACCURACY COMPARISONS OF 3DAA MODULE FOR THE SV DATASET WITH W-15

Class No.	Class Name	3D-CNN	3D-CNN& 2DAA	3D-CNN& 3DAA	ResNet	ResNet& 2DAA	ResNet& 3DAA
1	Brocoli_green_weeds_1	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
2	Brocoli_green_weeds_2	98.95	99.84	**100.00**	99.92	**100.00**	99.92
3	Fallow	**100.00**	**100.00**	**100.00**	99.85	**99.95**	99.49
4	Fallow_rough_plow	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
5	Fallow_smooth	**100.00**	**100.00**	**100.00**	99.22	99.78	**99.93**
6	Stubble	**100.00**	**100.00**	**100.00**	99.67	**100.00**	**100.00**
7	Celery	99.92	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
8	Grapes_untrained	85.53	81.31	**90.35**	89.15	**95.57**	95.19
9	Soil_vinyard_develop	98.87	**100.00**	**100.00**	99.97	**100.00**	**100.00**
10	Corn_senesced_green_weeds	97.07	99.60	**99.88**	99.54	99.85	**99.97**
11	Lettuce_romaine_4wk	99.63	99.91	**100.00**	99.91	**100.00**	**100.00**
12	Lettuce_romaine_5wk	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
13	Lettuce_romaine_6wk	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
14	Lettuce_romaine_7wk	99.72	99.72	**99.91**	**100.00**	**100.00**	**100.00**
15	Vinyard_untrained	85.53	90.62	**92.52**	92.71	92.60	**98.05**
16	Vinyard_vertical_trellis	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA (%)	-	94.65	95.54	**96.98**	96.65	98.06	**98.71**
κ (%)	-	94.05	94.93	**96.64**	96.28	97.84	**98.56**

Note: Bold numbers indicate the best performance.

TABLE VIII
CLASSIFICATION RESULTS OF DIFFERENT TYPES OF MODULE ON THREE DATASETS

Dateset	Window Size	Evaluation Indice	3D-CNN	3D-CNN& 2DAA	3D-CNN& 3DAA	ResNet	ResNet& 2DAA	ResNet& 3DAA
PU	W-5	OA (%)	92.11	94.64	**96.29**	92.75	95.63	**96.57**
		κ (%)	91.07	92.94	**95.13**	91.52	94.29	**95.48**
	W-15	OA (%)	98.27	99.02	**99.51**	98.82	99.47	**99.62**
		κ (%)	97.72	98.70	**99.36**	98.44	99.30	**99.49**
DFC	W-5	OA (%)	89.93	91.75	**94.10**	90.57	92.25	**94.14**
		κ (%)	88.31	89.66	**92.57**	89.11	91.91	**93.43**
	W-15	OA (%)	96.57	97.17	**97.58**	96.47	96.81	**98.01**
		κ (%)	95.43	96.73	**97.00**	95.22	95.79	**97.43**
SV	W-5	OA (%)	91.27	93.33	**94.63**	92.02	94.56	**95.98**
		κ (%)	90.82	92.71	**94.01**	91.13	93.62	**95.21**
	W-15	OA (%)	94.65	95.54	**96.98**	96.65	98.06	**98.71**
		κ (%)	94.05	94.93	**96.64**	96.28	97.84	**98.56**

Note: Bold numbers indicate the best performance.

On the whole, on the PU dataset, using 3D-CNN and ResNet classifiers, the 3DAA module is capable of achieving OA increments of 4.18% and 3.82% with W-5 for 3D-CNN and ResNet, of 1.24% and 0.8% with W-15 for 3D-CNN and ResNet. Regarding the DFC scene, OA increments are 4.17% and 3.57% with W-5 and 1.01% and 1.54% with W-15. As for the SV dataset, OA increments are 3.36% and 3.96% with W-5 and 2.33% and 2.06% with W-15. Similarly, the improvements in kappa coefficients in all cases are also very obvious.

Concerning the obtained classification results, attention-based modules, including the 2DAA module and 3DAA module, can improve the classification performance of classifiers regarding OA and the kappa coefficient. It is because the attention-based module can help the classifier attend to the discriminative features of HSI. So the classifier can be more robust in finding appropriate decision boundaries to separable different classes. The significant accuracy improvements of both 3D-CNN and ResNet show that the proposed attention module has good portability. Besides, the 3DAA module is more effective than the 2DAA module because of the use of information in spectral dimensions of HSI.

Furthermore, to more intuitively show the effectiveness of the 3DAA module in feature extraction, we use t-SNE [49], [50] to visualize features of training and validation samples before and after this module on the PU dataset, the DFC dataset, and the SV dataset in Figs. 7 –9, respectively. As is shown in Fig. 7, in the original samples, some classes are mixed together (e.g., class 2 and class 6), while they are separated in recalibrated samples. This means that recalibrated samples can be easier to classify. At the same time, it is known that class 2 (“Meadows”) and class 6 (“Bare Soil”) are similar in characteristics because they are spatial adjacency. So they are similar classes. Similarly, we can get the same conclusion in Fig. 8. The mixed classes (e.g., class 2 and class 6) are separated after the processing of the 3DAA module. In the DFC scene, class 2 (Trees) and class 6 (Vegetation) can belong to the same big category (Vegetation). So, they are similar classes too. Fig. 9 shows that class 8 (Grapes_untrained) and class 15 (Vinyard_untrained) are features mixed in original

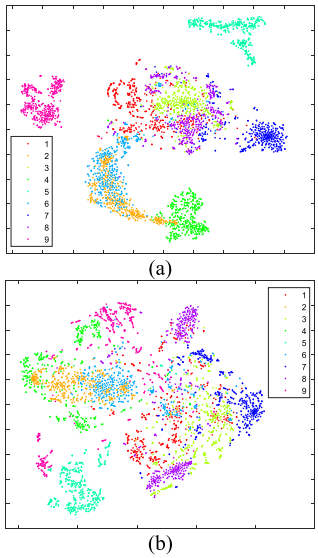

Fig. 7. Visualization of (a) original training and validation samples and (b) recalibrated ones by the 3DAA attention module with W-5 of the PU dataset by t-SNE. Different colors represent nine different classes. It is shown that, in the original samples, some classes are mixed together (e.g., class 2 and class 6), whereas they are separated in recalibrated samples. This means that recalibrated samples can be easier to classify.

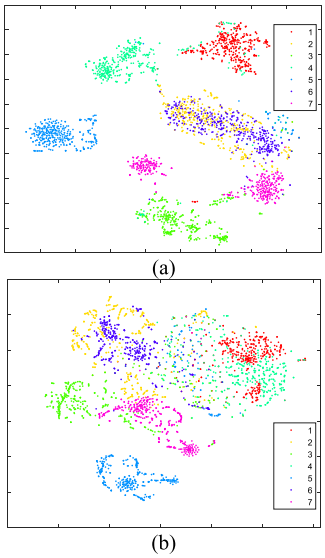

Fig. 8. Visualization of (a) original training and validation samples and (b) recalibrated ones by the 3DAA attention module with W-15 of the DFC dataset by t-SNE. Different colors represent seven different classes. It is shown that, in the original samples, some classes are mixed together (e.g., class 2 and class 6), whereas they are separated in recalibrated samples. This means that recalibrated samples can be easier to classify.

SV data. Just judging by common sense, these two classes are very similar. After the feature attention of the 3DAA module, the two classes are obviously separated.

2) Effectiveness of the HC Process: Tables IX–XI show the confusion matrices of validation samples with W-5 on the PU dataset, with W-15 on the DFC dataset and with W-15 on the SV dataset using a 3DAA classifier after the coarse classification. The nondiagonal elements in the confusion matrix represent

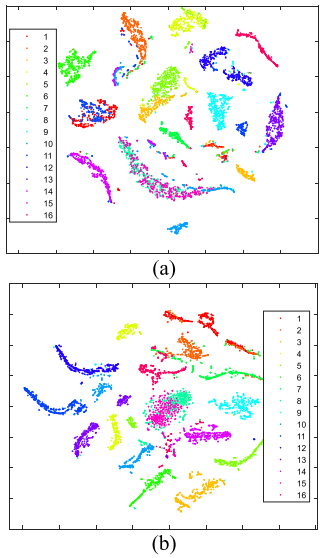

Fig. 9. Visualization of (a) original training and validation samples and (b) recalibrated ones by the 3DAA attention module with W-15 of the SV dataset by t-SNE. Different colors represent 16 different classes. It is shown that, in the original samples, some classes are mixed together (e.g., class 8 and class 15), whereas they are separated in recalibrated samples. This means that recalibrated samples can be easier to classify.

TABLE IX
CONFUSION MATRIX OF VALIDATION SAMPLES WITH W-5 ON THE PU DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7	8	9
1	194	0	0	0	0	0	0	2	0
2	0	189	0	0	0	5	0	0	0
3	0	0	196	2	0	0	0	11	0
4	0	1	0	196	0	0	0	0	0
5	0	0	0	0	200	0	0	0	0
6	0	10	0	1	0	194	0	0	0
7	2	0	0	1	0	1	200	0	0
8	4	0	4	0	0	0	0	187	0
9	0	0	0	0	0	0	0	0	200

TABLE X
CONFUSION MATRIX OF VALIDATION SAMPLES WITH W-15 ON THE DFC DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7	8	9
1	-	0	0	0	0	0	2	6	0
2	0	-	0	1	0	**15**	0	0	0
3	0	0	-	2	0	0	0	**15**	0
4	0	1	2	-	0	1	1	0	0
5	0	0	0	0	-	0	0	0	0
6	0	**15**	0	1	0	-	1	0	0
7	2	0	0	1	0	1	-	0	0
8	6	0	**15**	0	0	0	0	-	0
9	0	0	0	0	0	0	0	0	-

TABLE XI
CONFUSION MATRIX OF VALIDATION SAMPLES WITH W-15 ON THE SV DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7
1	199	1	1	0	0	3	0
2	0	182	1	0	0	7	0
3	0	0	195	0	0	0	0
4	0	2	3	200	0	0	0
5	0	0	0	0	198	0	0
6	1	15	0	0	0	190	0
7	0	0	0	0	2	0	200

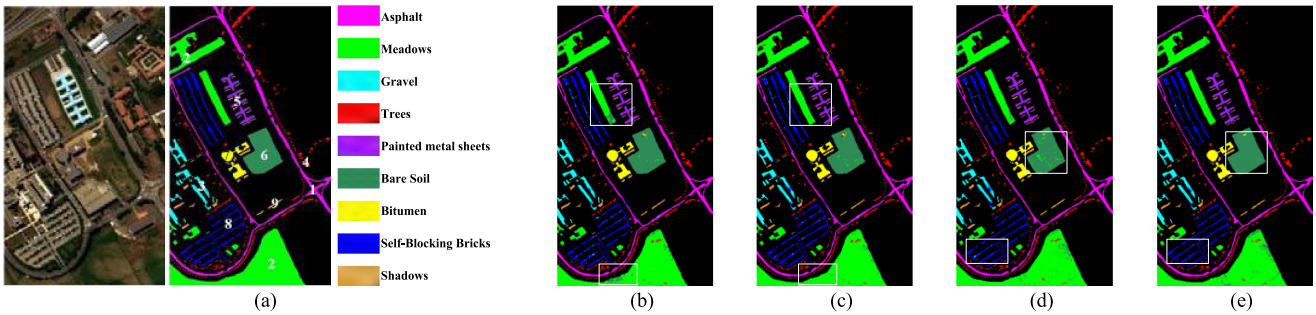

Fig. 10. Visual classification results of different approaches with W-5 on the PU dataset before and after the HC process. (a) False-color image of PU and its ground-truth map. (b) 3DAA with 3D-CNN classifier. (c) HC-3DAA with 3D-CNN classifier. (d) 3DAA with ResNet classifier. (e) HC-3DAA with ResNet classifier. The regions in white boxes show the main accuracy improvement of the HC process between similar classes.

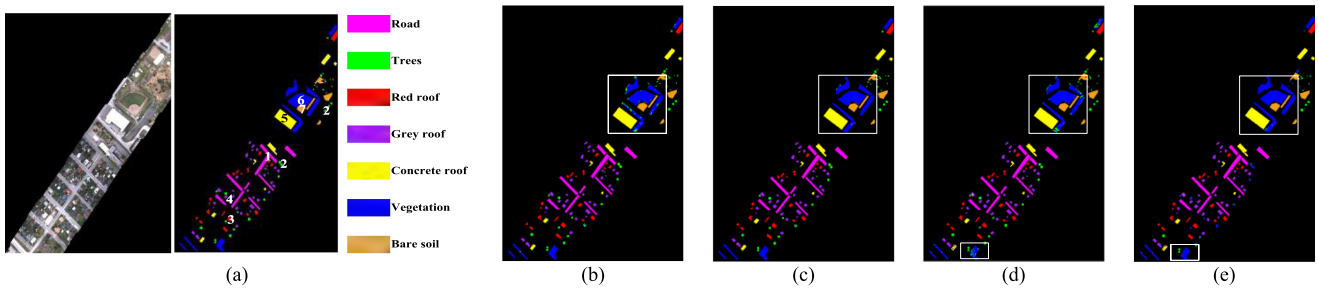

Fig. 11. Visual classification results of different approaches with W-15 on the DFC dataset before and after the HC process. (a) False-color image of DFC and its ground-truth map. (b) 3DAA with 3D-CNN classifier. (c) HC-3DAA with 3D-CNN classifier. (d) 3DAA with ResNet classifier. (e) HC-3DAA with ResNet classifier. The regions in white boxes show the main accuracy improvement of the HC process between similar classes.

TABLE XII
MERGING MATRIX OF VALIDATION SAMPLES WITH W-5 ON THE PU DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7
1	-	1	1	0	0	**4**	0
2	1	-	1	2	0	**22**	0
3	1	1	-	3	0	0	0
4	0	2	3	-	0	0	0
5	0	0	0	0	-	0	2
6	**4**	**22**	0	0	0	-	0
7	0	0	0	0	2	0	-

Note: Bold numbers indicate the merged classes with top confusion errors.

the number of misclassified samples. Besides, the merging matrices based on the confusion matrices are shown in Tables XII–XIV, respectively. As mentioned in Section II, the merging matrix is used to determine the similarity between classes. It is the symmetric matrix and classes with top values are defined as similar classes. Similar classes will be merged into a big category $R_{i,j}$.

According to Tables XII–XIV, selected categories are $R_{2,6}$ and $R_{3,8}$ in the PU dataset, $R_{1,6}$ and $R_{2,6}$ in the DFC dataset, and $R_{8,15}$ and $R_{8,10}$ in the SV dataset. These categories are consistent with the conclusion obtained in Figs. 7–9. They all meet the definition of similar classes in Section I. As we can see in Tables IX–XI, the classification performance of a classifier is mainly limited by the accuracy among these similar classes. The goal of the HC process is to better distinguish these categories through HC.

For every category, the improvements in PA are shown in Tables XV–XVII. It is obvious that the HC process can guide the classifiers to distinguish similar classes more efficiently. The PAs of C_2 and C_8 have increased by 3% and 1.8% in the reclassification of $R_{2,6}$ on the PU dataset. The PAs of C_6 and C_2 have increased by 3.00% and 1.10% in the reclassification of $R_{1,6}$ and $R_{2,6}$ on the DFC dataset. The PAs of C_8 and C_{15} have increased by 6.43% and 1.37% in the reclassification of $R_{8,15}$ and $R_{8,10}$ on the SV dataset. It is noted that although the reclassification of $R_{3,8}$ on the PU dataset doesn't achieve ideal results, the OA has also been improved and the results in other cases are all in line with expectations.

Figs. 10–12 demonstrate the visual classification maps of different approaches with W-5 on the PU dataset, with W-15 on the DFC dataset, and with W-15 on the SV dataset before and after the HC process, respectively. The regions in white boxes show the main accuracy improvement of the HC process between similar classes.

Table XVIII lists the OA and kappa coefficients before and after the HC process in all cases. Table XVIII also shows the categories $R_{i,j}$ in these cases. It is shown in Table XVIII that, even on the same image, the merged classes in different cases are sometimes inconsistent because of the randomness of the training process. But they are all reasonable, such as meadows and trees $R_{4,2}$ in PU belonging to the vegetation category, meadows and bare soil $R_{6,2}$ in PU being spatial adjacency, self-blocking bricks and gravel $R_{8,1}$ being made of similar materials, and so on. The effectiveness of the HC process on all datasets is

TABLE XIII
MERGING MATRIX OF VALIDATION SAMPLES WITH W-15 ON THE DFC DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	200	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	200	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	200	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	200	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	200	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	200	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	180	0	0	0	0	0	0	15	0
9	0	0	0	0	0	0	0	0	200	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	2	0	200	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	200	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	200	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	200	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	200	0	0
15	0	0	0	0	0	0	0	18	0	0	0	0	0	0	185	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	200

Note: Bold numbers indicate the merged classes with top confusion errors

TABLE XIV
MERGING MATRIX OF VALIDATION SAMPLES WITH W-15 ON THE SV DATASET AFTER COARSE CLASSIFICATION

C_i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	-	0	**2**	0	0	0	0	**33**	0
9	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	**2**	0	-	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0
15	0	0	0	0	0	0	0	**33**	0	0	0	0	0	0	-	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-

Note: Bold numbers indicate the merged classes with top confusion errors

TABLE XV
CONFUSION MATRIX OF HC PROCESS WITH W-5 ON PU DATASET

C_i	1	2	3	4	5	6	7	8	9
1	6442	0	0	11	0	2	1	40	0
2	7 (-1)	17715 **(+560)**	0	14 (+9)	0	72 (+54)	0	3 (+1)	0
3	33 (-12)	0	2058 (-89)	0	0	0	0	171 (-59)	0
4	0	175	0	3012	0	20	0	0	0
5	0	0	0	0	1345	0	0	0	1
6	2 (+1)	759 (-560)	0	19 (-9)	0	4893 (-54)	0	5 (-1)	0
7	69	0	1	1	0	28	1328	10	1
8	77 (+12)	0	40 (+89)	4	0	14	1	3450 **(+59)**	0
9	1	0	0	3	0	0	0	3	945
PA (%)	97.15	94.99 **(+3)**	98.05 (-4.24)	98.30	100	97.30 (-1.08)	99.85	93.70 **(+1.6)**	99.79

Note: Bold numbers indicate the improvements of HC process

TABLE XVI
CONFUSION MATRIX OF HC PROCESS WITH W-15 ON DFC DATASET

C_i	1	2	3	4	5	6	7
1	4437	7	6	3	0	45 (-44)	0
2	0	1000 **(+12)**	4 (-2)	1	0	285 (-176)	0
3	5	2	1808	4	4	0	0
4	1	16	29	2117	10	0	0
5	0	0	0	1	3862	5	0
6	0	66 (-12)	4 (+2)	0	0	6992 **(+220)**	0
7	0	2	3	0	12	30	1771
PA (%)	99.86	91.49 **(+1.10)**	97.52	99.58	99.33	95.04 **(+3.00)**	100

Note: Bold numbers indicate the improvements of HC process

very encouraging. On the PU dataset, the OAs increase by 0.75% and 0.66% for 3D-CNN and ResNet with W-5 and increase by 0.27% and 0.04% for 3D-CNN and ResNet with W-15. The OA improvement is more significant on DFC data that increased by 0.61% and 0.90% for 3D-CNN and ResNet with W-5, increased by 1.03% and 1.51% for 3D-CNN and ResNet with W-15. As for the SV dataset, the OAs have also been significantly improved. OAs increase by 2.14% and 1.44% for 3D-CNN and ResNet with W-5 and increase by 1.52% and 0.48% for 3D-CNN and ResNet with W-15. The main reason why the HC process can be effective

TABLE XVII
CONFUSION MATRIX OF HC PROCESS WITH W-15 ON SV DATASET

C_i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2009	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	3726	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1976	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	1394	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	2678	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	3959	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	3579	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	10183 **(+725)**	0	0	0	0	0	0	544 (-100)	0
9	0	0	0	0	0	0	0	0	6203	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	18 (-18)	0	3274	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	1068	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	2	0	1927	0	1	0	0
13	0	0	0	0	0	0	0	0	0	2	0	0	916	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1069	0	0
15	0	0	0	0	0	0	0	1065 (-707)	0	0	0	0	0	0	6724 **(+100)**	0
16	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	1807
PA(%)	100	100	100	100	100	100	100	90.35 **(+6.43)**	100	99.88	100	100	100	99.91	92.52 **(+1.37)**	100

Note: Bold numbers indicate the improvements of HC process

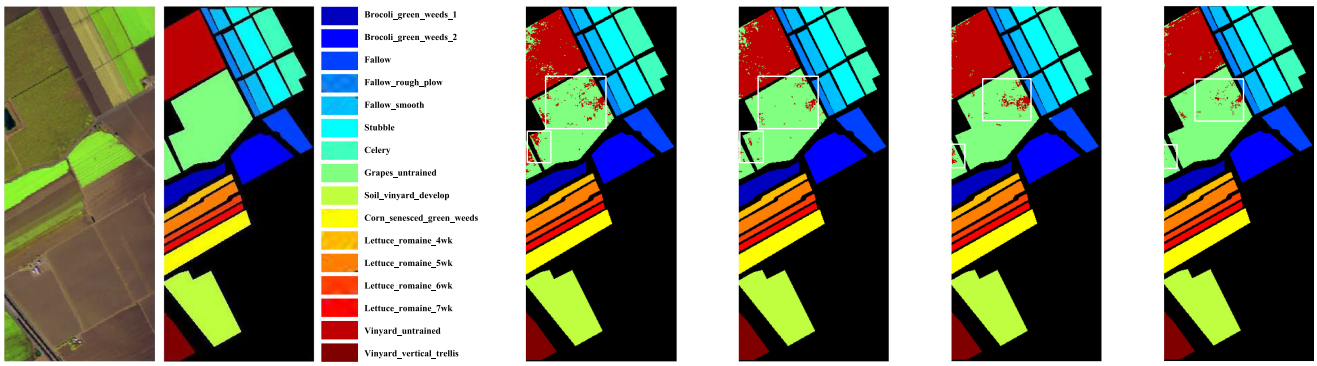

Fig. 12. Visual classification results of different approaches with W-15 on the SV dataset before and after the HC process. (a) False-color image of SV and its ground-truth map. (b) 3DAA with 3D-CNN classifier. (c) HC-3DAA with 3D-CNN classifier. (d) 3DAA with ResNet classifier. (e) HC-3DAA with ResNet classifier. The regions in white boxes show the main accuracy improvement of the HC process between similar classes.

is that discriminative features between similar classes are easier to be caught by the attention model without the interference of the other classes.

Throughout all the experiments, the proposed HC-3DAA module improves the OA of the 3D-CNN classifier from 92.11% to 97.04% with W-5 and from 98.27% to 99.78% with W-15 on the PU dataset, whereas from 92.75% to 97.23% with W-5 and from 98.82% to 99.66% with W-15 for the ResNet classifier. The improvements on the DFC and SV dataset are also very considerable and the classification accuracy has been improved by more than 3% in all cases.

3) Performance Comparison of HC-3DAA With Other State-of-the-Art Methods: In this experiment, some state-of-the-art attention-based methods, namely DFFN, SSRN, SpecAttenNet, and SSAN, are used as compared methods to verify the effectiveness of HC-3DAA.

Table XIX and Fig. 13 report the classification results conducted on the PU dataset. The proposed HC-3DAA method with the 3D-CNN classifier outperforms several state-of-the-art methods. For most classes, our proposed method has the superior PA. Especially for some classes with similar characteristics, such as "Meadows," "Gravel," "Bare Soil," and "Bitumen," the superiority of this algorithm is particularly encouraging. HC-3DAA achieves an OA of 99.78% and a kappa coefficient of 99.70%, which are the highest among compared methods.

As for the DFC dataset, Table XX and Fig. 14 show the quantitative evaluation results and visual results, respectively. The HC-3DAA method also achieves superior performance. In the experiment, all methods achieve very high accuracy in the "Road" and "Bare Soil" because these classes have specific radiation characteristics in LWIR and are easy to be distinguished from the other classes. The HC-3DAA outperforms other methods in the classification of similar classes. It achieves the highest OA of 98.61% and the highest kappa coefficient of 98.27%.

TABLE XVIII
CLASSIFICATION RESULTS BEFORE AND AFTER THE HC PROCESS

Dateset	Window Size	Evaluation Indice	3D-CNN & 3DAA	3D-CNN & HC-3DAA	ResNet & 3DAA	ResNet & HC-3DAA
PU	W-5	OA (%)	96.29	**97.04**	96.57	**97.23**
		κ (%)	95.13	**96.57**	95.48	**96.35**
		$R_{i,j}$	$R_{6,2}$, $R_{8,3}$		$R_{6,2}$, $R_{8,3}$	
	W-15	OA (%)	99.51	**99.78**	99.62	**99.66**
		κ (%)	99.36	**99.70**	99.49	**99.55**
		$R_{i,j}$	$R_{6,2}$, $R_{4,2}$		$R_{6,2}$, $R_{8,1}$	
DFC	W-5	OA (%)	94.10	**94.71**	94.14	**95.04**
		κ (%)	92.57	**93.18**	93.43	**93.67**
		$R_{i,j}$	$R_{6,2}$, $R_{5,4}$		$R_{6,2}$, $R_{5,4}$	
	W-15	OA (%)	97.58	**98.61**	98.01	**99.52**
		κ (%)	97.00	**98.27**	97.43	**99.31**
		$R_{i,j}$	$R_{6,2}$, $R_{6,1}$		$R_{6,2}$, $R_{6,1}$	
SV	W-5	OA (%)	94.63	**96.77**	95.98	**97.42**
		κ (%)	94.01	**96.21**	95.21	**96.88**
		$R_{i,j}$	$R_{8,15}$, $R_{8,10}$		$R_{8,15}$, $R_{8,10}$	
	W-15	OA (%)	96.98	**98.50**	98.71	**99.19**
		κ (%)	96.64	**98.33**	98.56	**99.09**
		$R_{i,j}$	$R_{8,15}$, $R_{8,10}$		$R_{8,15}$, $R_{5,3}$	

Note: $R_{i,j}$ is the merged category containing similar classes i and j

TABLE XIX
ACCURACY COMPARISONS FOR THE PU DATASET WITH W-15

Class No.	Class Name	DFFN	SSRN	SpecAttenNet	SSAN	HC-3DAA
1	Asphalt	99.31	**99.44**	98.25	99.59	**99.44**
2	Meadows	99.44	99.22	98.75	99.65	**99.95**
3	Gravel	99.19	99.90	99.19	99.86	**100.00**
4	Trees	98.66	98.76	**98.99**	98.76	98.62
5	Printed Metal Sheets	**100.00**	**100.00**	**100.00**	99.93	99.92
6	Bare Soil	97.20	**100.00**	99.32	98.35	99.88
7	Bitumen	**100.00**	99.85	**100.00**	**100.00**	**100.00**
8	Bricks	99.08	99.65	99.78	**99.97**	**99.97**
9	Shadows	99.89	99.47	99.89	**100.00**	**100.00**
OA (%)	-	99.10	99.43	98.97	99.49	**99.78**
κ (%)	-	98.77	99.20	98.60	99.26	**99.70**

Note: Bold numbers indicate the best performance

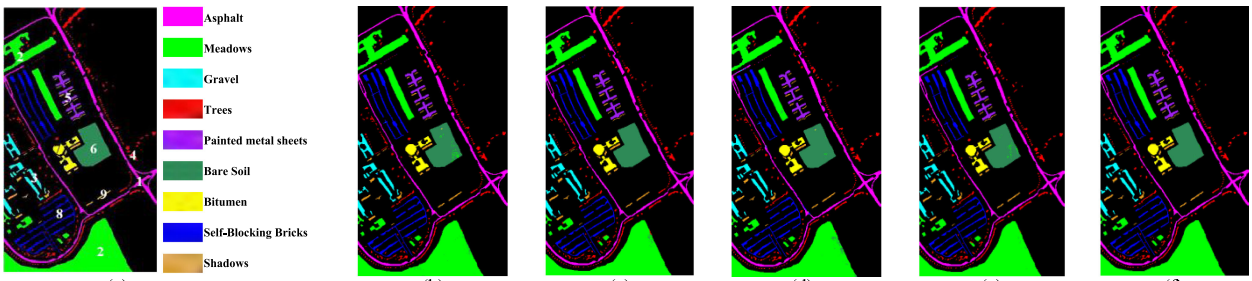

Fig. 13. Visual classification results of different approaches with W-15 on the PU dataset. (a) Ground-truth map of PU. (b) DFFN. (c) SSRN. (d) SpecAttenNet. (e) SSAN. (f) HC-3DAA with 3D-CNN classifier.

Table XXI and Fig. 15 show the accuracy comparison for the SV dataset with W-15. It is shown that all compared methods have good performance in classification for most classes. Their difference is mainly reflected in the distinction of similar classes, such as class 8 and class 15. The classification accuracy of these similar classes seriously restricts the OA. Our proposed HC-3DAA method has the best performance in the distinction between class 8 and class 15. It achieves the highest PAs of 96.78% and 93.89% for class 8 and class 15, respectively. Besides, the OA and kappa coefficient of HC-3DAA are also the best among compared methods.

TABLE XX
ACCURACY COMPARISONS FOR THE DFC DATASET WITH W-15

Class No.	Class Name	DFFN	SSRN	SpecAttenNet	SSAN	HC-3DAA
1	Road	99.71	**100.00**	99.91	**100.00**	99.86
2	Trees	86.83	91.86	**93.60**	90.12	92.59
3	Red Roof	**99.24**	97.52	98.76	98.00	97.52
4	Grey Roof	97.70	98.58	98.26	99.72	**99.58**
5	Concrete Roof	95.91	99.07	95.83	**99.33**	**99.33**
6	Vegetation	97.44	96.54	94.89	95.27	**98.04**
7	Bare Soil	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA (%)	-	97.48	97.98	97.02	97.67	**98.61**
κ (%)	-	97.02	97.36	96.88	97.42	**98.27**

Note: Bold numbers indicate the best performance

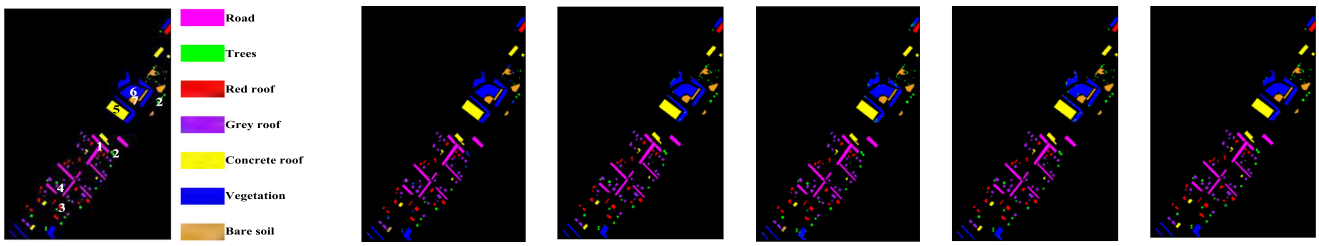

Fig. 14. Visual classification results of different approaches with W-15 on the DFC dataset. (a) Ground-truth map of DFC. (b) DFFN. (c) SSRN. (d) SpecAttenNet. (e) SSAN. (f) HC-3DAA with 3D-CNN classifier.

TABLE XXI
ACCURACY COMPARISONS FOR THE SV DATASET WITH W-15

Class No.	Class Name	DFFN	SSRN	SpecAttenNet	SSAN	HC-3DAA
1	Brocoli_green_weeds_1	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
2	Brocoli_green_weeds_2	99.76	99.89	99.81	**100.00**	**100.00**
3	Fallow	99.85	99.54	**100.00**	99.04	**100.00**
4	Fallow_rough_plow	99.64	99.50	**100.00**	**100.00**	**100.00**
5	Fallow_smooth	**100.00**	**100.00**	98.32	100.00	**100.00**
6	Stubble	**100.00**	**100.00**	99.92	99.90	**100.00**
7	Celery	99.58	**100.00**	**100.00**	**100.00**	**100.00**
8	Grapes_untrained	94.30	92.71	91.46	92.90	**96.78**
9	Soil_vinyard_develop	99.95	**100.00**	**100.00**	**100.00**	**100.00**
10	Corn_senesced_green_weeds	96.98	98.05	97.90	98.90	**99.88**
11	Lettuce_romaine_4wk	98.31	99.72	98.88	**100.00**	**100.00**
12	Lettuce_romaine_5wk	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
13	Lettuce_romaine_6wk	99.45	**100.00**	**100.00**	**100.00**	**100.00**
14	Lettuce_romaine_7wk	99.35	99.16	**100.00**	99.72	99.91
15	Vinyard_untrained	88.46	93.59	89.71	91.03	**93.89**
16	Vinyard_vertical_trellis	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
OA (%)	-	96.96	97.44	96.59	97.27	**98.50**
κ (%)	-	96.43	96.92	96.07	96.96	**98.33**

Note: Bold numbers indicate the best performance

All the experiments on the PU, the DFC, and the SV datasets illustrate the advantages of our proposed method in similar classes' differentiation.

To analyze the computational complexity of the proposed network and other methods, we calculate the average training and testing time (on GPU) on the PU dataset with W-15. The calculation results are shown in Table XXII. Because the parameters of each model are different and the average time is positively correlated with the number of parameters, the time consumption of the proposed network is more than that of other methods, which means that the improvements in classification performance increase the computation complexity.

IV. CONCLUSION

In this article, a new classification module named HC-3DAA is proposed to improve classification accuracy, especially for similar classes. Our HC-3DAA method is mainly divided into two parts, namely the 3DAA part and the HC process part. The 3DAA part aims to guide the classifiers to attend to discriminative features between classes and the HC process part can help

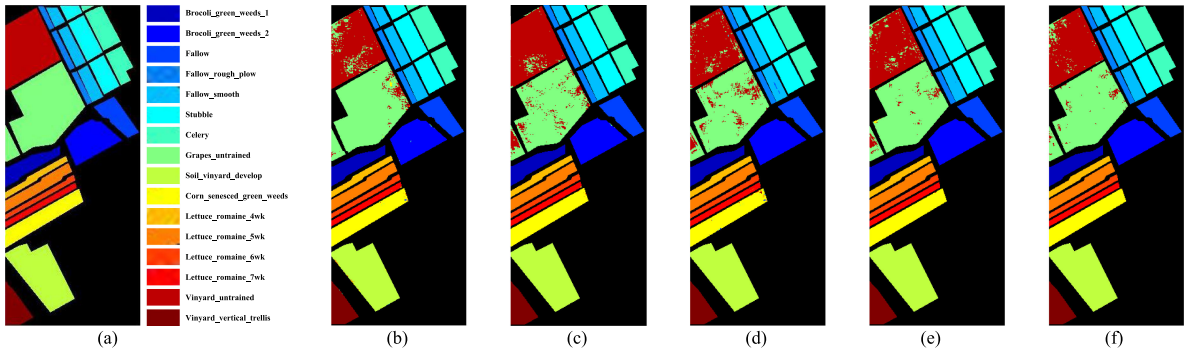

Fig. 15. Visual classification results of different approaches with W-15 on the SV dataset. (a) Ground-truth map of SV. (b) DFFN. (c) SSRN. (d) SpecAttenNet. (e) SSAN. (f) HC-3DAA with 3D-CNN classifier.

TABLE XXII
AVERAGE TRAINING AND TESTING TIME (ON GPU) ON THE PU DATASET WITH W-15

Methods	DFFN	SSRN	SpecAttenNet	SSAN	HC-3DAA	
					Coarse Classification	Fine Classification
Training Time (m)	138.4	463.6	170.8	485.2	766.8	191.6
Testing Times (s)	16.9	66.4	21.6	79.7	181.1	45.3

the classifiers distinguish similar classes more efficiently. The obtained results on the PU, DFC, and SV datasets demonstrate that the proposed method can significantly improve the performance of 3D-CNN and ResNet classifiers in HSI classification. It is also shown that our HC-3DAA method outperforms some state-of-the-art attention-based methods in similar classes' differentiation.

REFERENCES

[1] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. M. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 2, pp. 6–36, Jun. 2013.

[2] N. He *et al.*, "Feature extraction with multiscale covariance maps for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 755–769, Feb. 2019.

[3] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral-spatial hyperspectral image classification: An overview and new guidelines," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 3, pp. 1579–1597, Mar. 2018.

[4] 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.

[5] N. Bali and A. Mohammad-Djafari, "Bayesian approach with hidden Markov modeling and mean field approximation for hyperspectral data analysis," *IEEE Trans. Image Process.*, vol. 17, no. 2, pp. 217–225, Feb. 2008.

[6] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[7] P. O. Gislason, J. A. Benediktsson, and J. R. Sveinsson, "Random forests for land cover classification," *Pattern Recognit. Lett.*, vol. 27, no. 4, pp. 294–300, Mar. 2006.

[8] S. R. Joelsson, J. A. Benediktsson, and J. R. Sveinsson, "Random forest classifiers for hyperspectral data," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2005, pp. 4.

[9] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[10] 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.

[11] B. Waske, S. van der Linden, J. Benediktsson, A. Rabe, and P. Hostert, "Sensitivity of support vector machines to random feature selection in classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 7, pp. 2880–2889, Jul. 2010.

[12] 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.

[13] G. F. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large margin deep networks for classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 850–860.

[14] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[15] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1746–1751.

[16] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[17] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2261–2269.

[18] Q. Li, L. Mou, Q. Liu, Y. Wang, and X. X. Zhu, "HSF-Net: Multiscale deep feature embedding for ship detection in optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 12, pp. 7147–7161, Dec. 2018.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1–9.

[20] Y. Chen, X. Zhao, and X. Jia, "Spectral–spatial classification of hyperspectral data based on deep belief network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2381–2392, Jun. 2015.

[21] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 6, pp. 2094–2107, Jun. 2014.

[22] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3639–3655, Jul. 2017.

[23] H. Lyu, H. Lu, and L. Mou, "Learning a transferable change rule from a recurrent neural network for land cover change detection," *Remote Sens.*, vol. 8, no. 6, 2016, Art. no. 506.

[24] H. Lyu *et al.*, "Long-term annual mapping of four cities on different continents by applying a deep information learning method to Landsat data," *Remote Sens,* vol. 10, no. 3, 2018, Art. no. 471.

[25] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2020, doi: 10.1109/LGRS.2019.2918719.
[26] M. E. Paoletti, J. M. Haut, R. Fernandez-Beltran, J. Plaza, A. J. Plaza, and F. Pla, "Deep pyramidal residual networks for spectral–spatial hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 740–754, Feb. 2019, doi: 10.1109/TGRS.2018.2860125.
[27] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.
[28] Q. Wang, S. Liu, J. Chanussot, and X. Li, "Scene classification with recurrent attention of VHR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1155–1167, Feb. 2019.
[29] J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, and J. Li, "Visual attention-driven hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 8065–8080, Oct. 2019.
[30] W. Song, S. Li, L. Fang, and T. Lu, "Hyperspectral image classification with deep feature fusion network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, Jun. 2018.
[31] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.
[32] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 137–141, Feb. 2016.
[33] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020, doi: 10.1109/TPAMI.2019.2913372.
[34] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Computer Vision – ECCV 2018. ECCV 2018 (Lecture Notes in Computer Science including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. New York, NY, USA: Springer, 2018, pp. 3–19.
[35] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 510–519, doi: 10.1109/CVPR.2019.00060.
[36] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop*, 2019, pp. 1971–1980, doi: 10.1109/ICCVW.2019.00246.
[37] L. Mou and X. X. Zhu, "Learning to pay attention on spectral domain: A spectral attention module-based convolutional network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 110–122, Jan. 2020.
[38] H. Sun, X. Zheng, X. Lu, and S. Wu, "Spectral-Spatial attention network for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3232–3245, May 2020, doi: 10.1109/TGRS.2019.2951160.
[39] J. Choe and H. Shim, "Attention-based dropout layer for weakly supervised object localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2214–2223, doi: 10.1109/CVPR.2019.00232.
[40] J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza, and L. Plaza, "Hyperspectral image classification using random occlusion data augmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 11, pp. 1751–1755, Nov. 2019, doi: 10.1109/LGRS.2019.2909495.
[41] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 13001–13008, doi: 10.1609/aaai.v34i07.7000.
[42] K. K. Singh and Y. J. Lee, "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3544–3553, doi: 10.1109/ICCV.2017.381.
[43] G. Ghiasi, T. Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 10750–10760.
[44] J. Choe and H. Shim, "Attention-based dropout layer for weakly supervised object localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2214–2223, doi: 10.1109/CVPR.2019.00232.
[45] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "*mixup*: Beyond empirical risk minimization," in *Proc. Int. Conf. Learn. Representations Conf. Acceptance Decis.*, 2018.
[46] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6022–6031.
[47] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6546–6555, doi: 10.1109/CVPR.2018.00685.
[48] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, May 2015.
[49] L. Van Der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.
[50] L. van der Maaten, "Accelerating t-SNE using tree-based algorithms," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 3221–3245, Oct. 2014.

Xinyuan Miao (Student Member, IEEE) received the B.S. and M.S. degrees in information and communication engineering, in 2015 and 2017, respectively, from the Harbin Institute of Technology, Harbin, China, where he is currently working toward the Ph.D. degree in information and communication engineering.

His current research interests include hyperspectral image classification, thermal infrared remote sensing image processing, and artificial intelligence application.

Ye Zhang (Member, IEEE) received the B.S. degree in communication engineering and the M.S. and Ph.D. degrees in communication and electronic system from the Harbin Institute of Technology (HIT), Harbin, China, in 1982, 1985, and 1996, respectively.

In 1985, he joined HIT as a teacher. Between 1998 and 1999, he was a Visiting Scholar with the University of Texas at San Antonio. He is currently a Professor and Doctoral Supervisor in information and communication engineering. He is the Director of the Institute of Image and Information Technology with the School of Electronic and Information Engineering, HIT. His research interests include remote sensing hyperspectral image analysis and processing and image video compression and transmission as well as multisource information collaboration processing and applications.

Junping Zhang (Senior Member, IEEE) received the B.S. degree in biomedical engineering and instrument from the Harbin Engineering University and Harbin Medical University, Harbin, China, in 1993, and the M.S. and Ph.D. degrees in signal and information processing from the Harbin Institute of Technology (HIT), Harbin, China, in 1998 and 2002, respectively.

She is currently a Professor with the Department of Information Engineering, School of Electronics and Information Engineering, HIT. Her research interests include hyperspectral data analysis and image processing, multisource information fusion, pattern recognition, and classification.

Xuejian Liang (Graduate Student Member, IEEE) received the B.S. and M.S. degrees in software engineering from Liaoning Technical University, Huludao, China, in 2015 and 2018, respectively. He is currently working toward the Ph.D. degree in information and communication engineering with the Harbin Institute of Technology.

His current research interests include hyperspectral image classification, biophysical/biochemical retrieval, and artificial intelligence application.